



20212/056/
2024

uciaprcn0141
IP外面仕様書

Ver1.012

個別IP要求仕様書

文書番号: LLWEB-10016241

文書名: RS-CANFDv4.1 要求仕様書

ルネサスエレクトロニクス株式会社

Approved	Checked	Prepared
2022.06.29 2021.05.27 DPDS1	2022.06.29 2021.05.27 DPDS1	202 <u>12</u> .0 <u>56</u> . 224 DPDS1



PROJECT TITLE RS-CAN-FD

DOCUMENT TITLE IP SPECIFICATION

IP SPECIFICATION

Version: 1.021-04 Status: Issued

PROJECT TITLE: RS-CAN-FD

PRODUCT: NA

PRODUCT VARIANT 8C

DOCUMENT TITLE: IP SPECIFICATION

DOCUMENT NUMBER: D012250_111_S01

AUTHOR H.NAKAMURA

DOCUMENT FILE LOCATION RS-CAN-FD REPOSITORY

DOCUMENT HISTORY

	Name	Responsibility	Initials	Version	Date
Prepared	Hideo Nakamura	Engineer	H.N.	1.042	May Jun 224 20242
Prepared	Toshiyuki <u>Uemura</u>	PM	T. U	1.042	Jun 29 2022 May 27 2024

1 Abstract

1.1 Purpose

This document provides details of the registers and functionality of the RS-CAN-FD module.

Note: This is an exported document from DOORS. It is not under revision control. Maintenance is done in DOORS. Please make sure that the latest export version from the DOORS database is used.

1.2 Table of Contents

1	Abstract	3
1.1	Purpose.....	3
1.2	Table of Contents	3
2	RS-CAN-FD Outline	9
2.1	Overview	9
2.2	RS-CAN-FD module Overview	10
2.3	Notes for the product design.....	11
2.3.1	Module connection to RAM and ECC	11
2.3.2	APB Interface	12
2.4	Notes for the User Manual (UM) creation.....	12
2.5	RS-CAN-FD usage limitation	13
2.5.1	Requirement notes.....	13
2.5.2	Standards.....	13
2.5.3	Known bugs	13
2.5.4	Legal information (patent and export control license).....	13
2.5.5	Parameterization	13
3	Interface Specification	18
3.1	Interconnection	18
3.2	Interface signals.....	19
3.2.1	Reset and Clock control	19
3.2.2	APB bus control	23
3.2.3	Interrupt Interface.....	25
3.2.4	DMA Interface	26
3.2.5	Port Interface and mode select signal	28
3.2.6	RAM Interface	36
3.2.7	R-ACE Interface.....	45
4	Register Overview	49
4.1	Register Table	49
4.2	Legend	49
4.3	Register description	54
4.3.1	CFDCnNCFG	54
4.3.2	CFDCnCTR	56
4.3.3	CFDCnSTS	63
4.3.4	CFDCnERFL	66
4.3.5	CFDCnDCFG	74
4.3.6	CFDCnFDCFG	76
4.3.7	CFDCnFDCTR	81
4.3.8	CFDCnFDSTS	83
4.3.9	CFDCnFDCRC	86
4.3.10	CFDGIPV	87
4.3.11	CFDGCFG	90
4.3.12	CFDGCTR	93
4.3.13	CFDGSTS	96
4.3.14	CFDGERFL	98
4.3.15	CFDGTINTSTSv	102
4.3.16	CFDGTSC	106

4.3.17	CFDGAFLCTR	107
4.3.18	CFDGAFLCFGw	109
4.3.19	CFDGAFLIDr.....	111
4.3.20	CFDGAFLMr	113
4.3.21	CFDGAFLP0r.....	115
4.3.22	CFDGAFLP1r.....	118
4.3.23	CFDRMNB	122
4.3.24	CFDRMNDt.....	123
4.3.25	CFDRFCCa.....	125
4.3.26	CFDRFSTSa	128
4.3.27	CFDRFPCTRa	131
4.3.28	CFDCFCCd.....	132
4.3.29	CFDCFCCEd	136
4.3.30	CFDCFSTSD	139
4.3.31	CFDCFPCTRd	143
4.3.32	CFDFESTS	144
4.3.33	CFDFFSTS	145
4.3.34	CFDFMSTS.....	146
4.3.35	CFDRFISTS	147
4.3.36	CFDCFRISTS	148
4.3.37	CFDCFTISTS	149
4.3.38	CFDFFFSTS	150
4.3.39	CFDCFMOWSTS.....	151
4.3.40	CFDCFOFRISTS	152
4.3.41	CFDCFOFTISTS	153
4.3.42	CFDCDTCT	154
4.3.43	CFDCDTSTS	156
4.3.44	CFDCDTTCT	158
4.3.45	CFDCDTTSTS	160
4.3.46	CFDGRINTSTS _n	163
4.3.47	CFDTMCi	166
4.3.48	CFDTMSTSj.....	168
4.3.49	CFDTMTRSTSf.....	170
4.3.50	CFDTMTARSTSf	172
4.3.51	CFDTMTCSTSf.....	174
4.3.52	CFDTMTASTSf	176
4.3.53	CFDTMIEcf	178
4.3.54	CFDTXQCC0n	180
4.3.55	CFDTXQCC1n	184
4.3.56	CFDTXQCC2n	188
4.3.57	CFDTXQCC3n	192
4.3.58	CFDTXQSTS0n	195
4.3.59	CFDTXQSTS1n	199
4.3.60	CFDTXQSTS2n	202
4.3.61	CFDTXQSTS3n	205
4.3.62	CFDTXQPCTR0n	208
4.3.63	CFDTXQPCTR1n	209
4.3.64	CFDTXQPCTR2n	210
4.3.65	CFDTXQPCTR3n	211
4.3.66	CFDTXQESTS	212
4.3.67	CFDTXQFISTS	213
4.3.68	CFDTXQMSTS	215
4.3.69	CFDTXQOWSTS	217
4.3.70	CFDTXQISTS	218
4.3.71	CFDTXQOFTISTS	219
4.3.72	CFDTXQOFRISTS	220
4.3.73	CFDTXQFSTS	222
4.3.74	CFDTHLCCn	223

4.3.75	CFDTHLSTS _n	225
4.3.76	CFDTHLACC0 _n	227
4.3.77	CFDTHLACC1 _n	229
4.3.78	CFDTHLPCTR _n	230
4.3.79	CFDGRSTC	231
4.3.80	CFDGFCMC.....	232
4.3.81	CFDGBISC.....	234
4.3.82	CFDGFTBAC	235
4.3.83	CFDGTSTCFG.....	237
4.3.84	CFDGTSTCTR.....	239
4.3.85	CFDGFDCFG.....	240
4.3.86	CFDGCRCFCG	241
4.3.87	CFDGLOCKK.....	242
4.3.88	CFDGLOTB.....	243
4.3.89	CFDRPGACCK	245
4.3.90	CFDCnBLCT	246
4.3.91	CFDCnBLSTS.....	247
4.3.92	CFDGFFIMC	248
4.3.93	CFDGVMEIS	249
4.3.94	CFDVMCFGn.....	251
4.3.95	CFDVMRFCFG	255
4.3.96	CFDVMISTSn	257
4.3.97	CFDGPFLCTR	262
4.3.98	CFDGPFLCFGu.....	263
4.3.99	CFDGPFLIDs	265
4.3.100	CFDGPFLMs	267
4.3.101	CFDGPFLP0s	269
4.3.102	CFDGPFLP1s	272
4.3.103	CFDGPFLPTs	275
4.3.104	CFDGPFLPD0s	277
4.3.105	CFDGPFLPD1s	278
4.3.106	CFDGPFLPM0s	279
4.3.107	CFDGPFLPM1s	280
4.3.108	CFDGAALIGNENT	281
4.3.109	CFDGAALIGNCTR	282
4.4	Identifier Bits Alignment	283
4.5	Message Buffer Component Structure	284
4.5.1	Start Addresses	284
4.5.2	CFDRMBCPb[i]	285
4.5.3	CFDRMID	287
4.5.4	CFDRMPTR	288
4.5.5	CFDRMFSTS	289
4.5.6	CFDRMDFp	291
4.5.7	CFDRFMBCPb[i]	293
4.5.8	CFDRFID	295
4.5.9	CFDRFPTR	296
4.5.10	CFDRFFDSTS	297
4.5.11	CFDRFDFp	299
4.5.12	CFDCFMBCPb[i]	301
4.5.13	CFDCFID	303
4.5.14	CFDCFPTR	305
4.5.15	CFDCFFDCSTS	306
4.5.16	CFDCFDFp	308
4.5.17	CFDTMBCPb[i]	310
4.5.18	CFDTMID	312
4.5.19	CFDTMPTR	314
4.5.20	CFDTMFDCTR	315
4.5.21	CFDTMDFp	317

5	Modes of operation	319
5.1	Overview	319
5.2	Global Modes	319
5.2.1	Global Sleep Mode	320
5.2.2	Global Reset Mode	321
5.2.3	Global Halt Mode	323
5.2.4	Global Operation Mode	325
5.3	Channel Modes	326
5.3.1	CAN channel Sleep Mode	326
5.3.2	CAN channel Reset Mode	326
5.3.3	CAN channel Halt Mode	327
5.3.4	CAN channel Operation Mode	328
5.3.5	CAN channel Bus-Off State	330
5.4	Global Mode – Channel Mode transition interactions	332
5.4.1	Global Mode change timing	333
5.4.2	Channel Mode change timing	333
5.4.3	Register behaviour in global/channel Modes	334
5.5	IP operation modes	369
5.6	Influence of the register by Parameterization	370
5.7	Influence of the bit by Parameterization	392
5.8	software constraint	439
5.9	Influence of the operation by Parameterization	446
5.10	Influence of the bit by Classical CAN only mode	448
6	Initialisation	530
6.1	Initialization of CAN Clock, Bit Timing and Baud Rate	530
6.1.1	Bit Timing Conditions	530
6.1.2	CAN Bit Timing	531
6.1.3	Baud Rate	532
6.1.4	Setting of CAN Clock, Bit Timing and Baud Rate	533
6.1.5	Transmitter delay compensation	534
6.2	CAN Module Configuration after H/W Reset	536
7	Acceptance Filtering Function using Global Acceptance Filter List (AFL)	539
7.1	Overview	539
7.2	Allocation of AFL entries to each CAN channel	540
7.3	AFL entry description	541
7.4	Entering entries in the AFL	543
7.5	Loopback modes	547
7.6	IDE Masking	548
7.7	updating AFL entry during communication	549
8	FIFO Buffers & Normal MB Configuration	553
8.1	Normal RX Message Buffers	555
8.1.1	Normal RX Message Buffer configuration	555
8.2	FIFO Buffers	555
8.2.1	FIFO Buffers configuration	555
8.2.2	FIFO Buffers control	560
9	Interrupts and DMA	561
9.1	Interrupts	561
9.2	DMA Transfer	568
10	Reception and Transmission	572
10.1	Reception	572
10.1.1	Message storage in RX Message Buffers	572
10.1.2	Message storage in FIFO Buffers	573
10.1.3	Timestamp	575
10.2	Transmission	576
10.2.1	Transmission Priority	578
10.2.2	Normal Transmission	579
10.2.3	TX FIFO or GW FIFO Transmission	584
10.2.4	TX Queue	588

10.2.5	TX History List.....	593
10.2.6	TX data padding.....	597
11	ECC Check	598
12	Test Mode.....	603
12.1	Channel specific test modes.....	603
12.1.1	Basic test mode	603
12.1.2	Listen-only mode.....	603
12.1.3	Self test mode 0 (External Loop back mode).....	604
12.1.4	Self test mode 1 (Internal Loop back mode).....	604
12.1.5	Restricted Operation Mode	605
12.2	Global test modes	605
12.2.1	RAM Test Mode	606
12.2.2	Internal CAN Bus Communication Test Mode	609
12.2.3	OTB Test Mode	610
13	RAM area configuration.....	612
13.1	Examples	619
13.2	OTB Area	620
13.3	num_ram_chan dependencies	620
14	Bus traffic measurement	622
14.1	How to count the CAN bus idle time	622
14.2	Operations and measurement procedure	622
15	Flexible CAN mode.....	624
16	Flexible transmission buffer assignment	625
17	FFI mode	629
17.1	Outline in FFI mode (channel independence)	629
17.2	Outline in FFI mode (buffer independence)	629
17.3	Prohibition function in FFI mode	630
17.4	Restrictions function in FFI mode	630
17.5	Interruption output in FFI mode	630
17.6	Unavailable registers in FFI mode	634
17.7	Transition to FFI mode.....	635
17.8	<i>Setup of an external terminal (rs_vnnum_fix)</i>	637
17.9	Measures of fail safe.....	637
18	PNF (Pretended Network Filter)	639
18.1	Explanation in each mode	639
18.1.1	Pretended Network Filter Mode	640
18.1.2	Pretended Network Filter ID only and Acceptance Filter Mode	640
18.1.3	Acceptance Filter Mode	640
18.1.4	Pretended Network Filter and Acceptance Filter Mode	641
18.2	Details of a filtering function.....	641
18.2.1	PN ID filter.....	641
18.2.2	PN payload filter.....	641
18.2.3	Setup of the number of rules of PNF	642
18.2.4	Software flow of a payload filter	643
18.2.5	Byte alignment of payload filter.....	645

Notes

In the following sections of the RS-CAN-FD module specification, the use of following words express software restriction or hardware protection:

The words, “... should not ...” or “... should ... only ...” or "... should only ..." express a software restriction. A software restriction is mandatory for the users and the behaviour of the module is not guaranteed if these restrictions are violated.

The words “... cannot ...” or “... can ... only ...” or "... can only ..." express existence of hardware protection. The behaviour of the module is not affected if these conditions are violated.

Abbreviation

REE	Renesas Electronics Europe
ETC	European Technology Centre
CAN	Controller Area Network
CH	CHannel
DB	Data Byte
DLC	Data Length Code
DLL	Data Link Layer
DLLC	Data Link Layer Clock
FD	Flexible Data
FIFO	First In First Out
GW	GateWay
HTH	Hardware Transmission Handler
HW	HardWare
ID	IDentifier
INTF	Interrupt Flag
IP	Intellectual Property
MB	Message Buffer
MBCP	Message Buffer Component
MBCPR	Message Buffer Component Register
MBCGR	Message Buffer Configuration Register
MBCTR	Message Buffer Control Register
MBSTR	Message Buffer Status Register
PLL	Phase Locked Loop
R	Read-only access
R/W	Read or Write access
RAM	Random Access Memory
RN	Rule Number
RX	Receive
SW	SoftWare
TX	Transmit
RX FIFO	FIFO only for reception of CAN messages or Common FIFO in reception mode
TX FIFO	Common FIFO in transmission mode
GW FIFO	Common FIFO in CAN-CAN gateway mode
THL	TX History List
STD-ID	Standard ID
EXT-ID	Extended ID
OTB	One Time Buffer
SSP	Secondary Sample Point

TDC	Transceiver Delay Compensation
VM	Virtual Machine

2 RS-CAN-FD Outline

This Target Specification describes the functionality of the RS-CAN-FD IP 8 channel variant. The general RS-CAN-FD IP is the successor IP of the RS-CAN developed in REE.

The RS-CAN-FD module has been developed to meet the following objectives:

- improve system performance (to run CPU operations based on AUTOSAR system software);
- support multiple channel operation;
- support the gateway function;
- support CAN with Flexible Data-Rate;

It has a flexible Message Buffer and FIFO structure to meet requirements of various applications. It also provides test modes to achieve high testability of the module that can be useful for “power-on” testing.

This specification describes a 8 channel implementation of the RS-CAN-FD module.

2.1 Overview

Macro name	uciaprcn0141
Macro type	soft macro
Macro function	CAN-Controller
Macro Version	4.1a
Function overview	CAN-FD ISO 11898-1 (2015)
Protocol engine Version	RS-CANFD_PE V3.0
Number of channels	Parameter set(2, 4, 6, 8ch)

Each CAN channel in the RS-CAN-FD module complies with the CAN-FD ISO 11898-1 (2015) Specifications. Each CAN channel can transmit and receive messages in the 11-bits Standard ID format as well as the 29-bit Extended ID format.

2.2 RS-CAN-FD module Overview

Selectable number of CAN channels CH0 - CHn (n = 7for this specification)

Item	Specification
Communication	CAN functionality conform to CAN-FD ISO 11898-1 (2015)
Protocol engine Version	RS-CANFD_PE V3.0
Gateway Function	CAN 2.0 <> CAN 2.0 CAN 2.0 <> CAN-FD Gateway (Only 8 Byte Payload) CAN-FD <> CAN-FD
Data transfer rate	up to 1Mbps for arbitration phase and up to 8Mbps for data phase , individually for each CAN channel
Proposed min. operation frequency peripheral clock/APB clock	80MHz
Data Link Layer clock (DLLC)	max ≤ CHI operation Frequency
Input/Output pins	TX/RX
CAN channels	n+1
Selectable ID type	11-bit Standard ID 11-bit Standard ID + 18-bit Extended ID
Selectable Frame Type	Data Frame (RTR = 0) (CAN and CAN-FD frames) Remote Frame (RTR = 1) (only CAN frames)
Variable Data Byte Count for Data Frames	DLC range: 0 to F
Message Buffer	64 transmit message buffers per channel **1 4 transmission queue per channel Automatic message transfer into transmission queues supported 256 shared buffers for RXMB and FIFO buffers per channel **1 - RXMB: - Up to 16*(n+1) reception message buffers, shared among all the CAN channels - FIFO buffers: - 8 Reception FIFO Buffers - Up to 3*(n+1) FIFOs individually configurable as Reception FIFO / Transmission FIFO / CAN to CAN Gateway FIFO
Automatic delay interval timer for transmission	The delay timer can be applied to: -Transmission FIFO -CAN to CAN Gateway FIFO
Enhanced reception filtering	support of 11bit and 29bit CAN identifier programmable 29 bit CAN identifier acceptance filter mask for each entry
	programmable GW routing capability for each channel (up to 8 routing destinations)
	RTR and IDE masking
	DLC filter
	Message buffer payload overload protection
	Payload filter
	Updating AFL entry during communication
General SW Support	Automatic label information added to receive message (for upper SW layer support)
Timer	TX and RX Time Stamp function
Power down function	Module start stop function for each CAN node (Channel & Global Sleep Mode)
RAM	RAM ECC protected
Bus traffic measurement	CAN bus traffic measurement of each Channel is possible
<i>Bus Interface</i>	<i>support of APB and R-ACE Interface</i>
FFI mode	Channel level independency
	Buffer level independency for RX/TX handler

**1 : Refer to 2.5.5 Parameterization

Table 2.1 Overview of RS-CAN-FD items

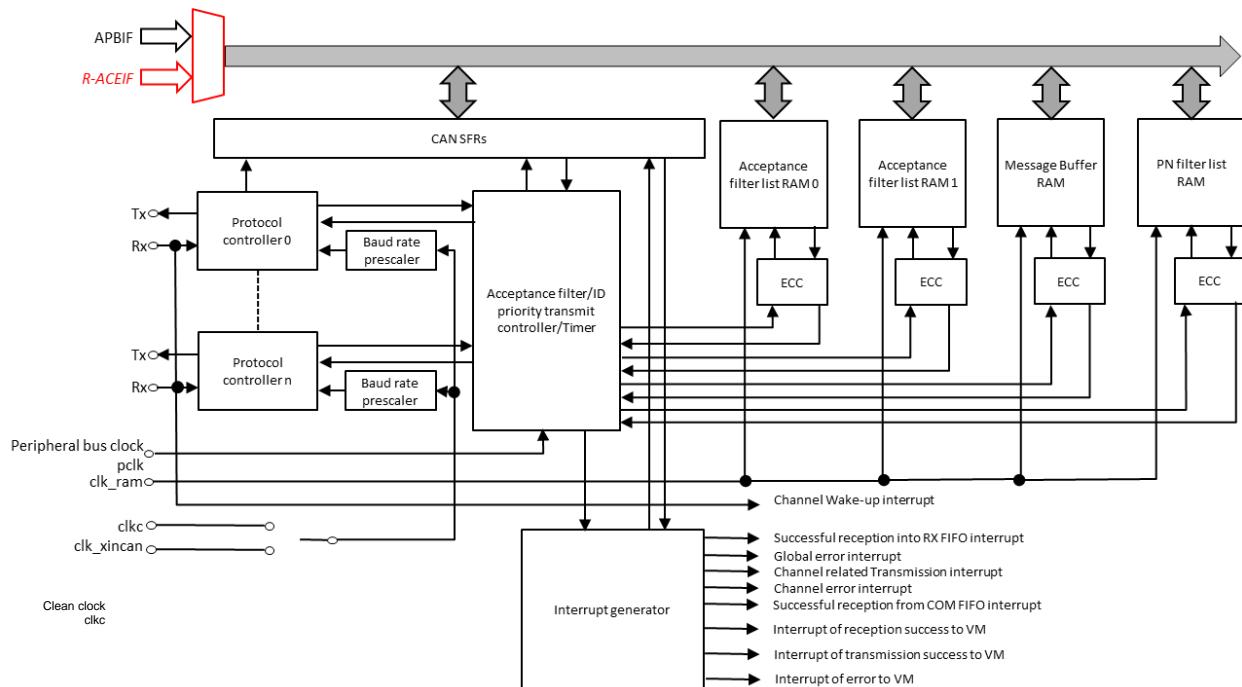


Figure 2.1 Overview of the RS-CAN-FD Module

Tx/Rx:

Input/Output pins of the CAN module

Protocol controller:

Handles CAN protocol processing such as bus arbitration, bit timing at transmission and reception, stuffing, error handling, etc.

Acceptance filter list RAM:

This RAM is used to store the message acceptance filtering entries for all channels. Each acceptance filter entry has an individual ID, data length code, data field, message pointer for upper layer application usage and message direction pointer. The AFLRAM is divided in two parts to accelerate the AFL access process.

Message Buffer RAM:

This RAM is used to store messages after reception or for transmission using a normal Message Buffer or a FIFO. Each message entry has an individual ID, data length code, data field, message pointer for upper layer application usage and a time stamp.

Acceptance filter:

Performs filtering of received messages. The entries in the Acceptance filter list RAM are used for the filtering process.

Timer:

Two timers

-Reception Timestamp function

-Transmission separation time for FIFO Buffers

Interrupt generator:

Generates several types of global and channel interrupts

CAN SFRs:

Registers associated with CAN. Refer to Section 4.3.

2.3 Notes for the product design

2.3.1 Module connection to RAM and ECC

This macro needs 4 RAM-modules. Maximum two 12.3K RAM (Total 24.6K RAM) for all AFL Rule entries, maximum one 197.9K RAM for all the Messages, and maximum one 5.8K RAM for all PFL Rule entries. The

RAM modules are connected via error connection code (ECC) soft macro.

During the product design, the timing requirements should be carefully considered.

(RAM size changes according to parameter conditions.)

2.3.2 APB Interface

The RS-CAN-FD IP is connected via the APB bus to the CPU. The implemented APB behaviour follows the APB-I/F Rule for M40 Platform Ver.4.0. with the exception:

Rule	Exception	comment
Address mapping rule (3)	flags are cleared by writing 1'b1	for RS-CAN-FD IP it is not allowed to use the bit clear instruction, instead of that the MOV instruction should be used

Table 2.2 Exception from APB-I/F Rule

2.4 Notes for the User Manual (UM) creation

The basic functionality of the RS-CAN-FD has not changed from the RS-CAN IP.

Items, which are not open to the customer are formatted in red colour and Italic style, and need to be deleted, when the UM is generated.

A blue colour shows the Change part from RS-CANFDv4.1 to RS-CANFDv4.1a.

There is the following explanation by the bit clearing of a status bit.

"Do not use bit clear instruction for clearing this bit.

Use the MOV instruction to ensure that only the specified bit is cleared.

Then the other bits remain set to 1'b1."

The MOV instruction is an instruction of SH microcomputer.

Please change this MOV instruction to the instruction of the microcomputer used.

2.5 RS-CAN-FD usage limitation

2.5.1 Requirement notes

2.5.1.1 Virtual cells

The RS-CAN-FD IP is using the following virtual cells:

ux099smcclmux20	clock multiplexer cell
ux099smc01gck0	clock gating cell
ux099smcmux20	multiplexer cell

Table 2.3 Virtual cells

2.5.1.2 Module area

The module area is the increase of 60% of RSCAN-FD Ver3 module area.

2.5.1.3 Operation frequency

The proposed minimum operation frequency pclk is 80 MHz (refer to Section. 3.2.1.5).

The maximum operation frequency pclk is about 80 MHz.

The maximum operation frequency clk_ram is about 160 MHz.

2.5.2 Standards

The RS-CAN-FD IP complies with CAN communication standard defined by the CAN-FD ISO 11898-1 (2015).

2.5.3 Known bugs

None

2.5.4 Legal information (patent and export control license)

License condition are still not finally concluded with BOSCH. But draft royalty contract between Renesas and BOSCH is in place. By this royalty must be paid only for the CAN FD protocol. But no royalty payment for CAN-FD disabled products.

2.5.5 Parameterization

RTL of RS-CAN-FDVer4 parameterized the following item.

- 1) Number of channel : 2channel, 4channel, 6channel, 8channel
- 2) Number of TXMB : 16buffer/ch, 32buffer/ch, 64buffer/ch
- 3) Number of AFL entry : 64rule/ch, 128rule/ch, 192rule/ch
- 4) Number of Pool Buffer (1buffer : 76Byte) : 32buffer/ch, 48buffer/ch, 64buffer/ch, 128buffer/ch, 256buffer/ch
- 5) Same ID overwrite function of TXQ : ON (implemented) / OFF (unimplemented)
- 6) *FCC parameter function ON (implemented) / OFF (unimplemented)*
(When the maximum parameter specification, an FCC parametric function can be set.)

The specification restrictions by parameter of a TXMB number are as follows.

Item	Number of TXMB		
	64TXMB	32TXMB	16TXMB
Number of TXMB	TXMB0-63	TXMB0-15 TXMB32-47	TXMB0-7 TXMB32-39
Flexible TXMB mode	Implementation		

FlexibleTXMB assignment range	0,4,8,12,16,20,24,28,32	0,4,8,12,16	0,4,8
Link destination of CFIFO	TXMB32-63	TXMB32-47	TXMB32-39
Number of TX Queue	TXQ0/1/2/3		
TX Queue Depth Configuration	from 3 to 32 by each TXQ	from 3 to 16 by each TXQ	from 3 to 8 by each TXQ
TX Queue access window	TXQ0 : TXMB0 / TXQ1 : TXMB31 / TXQ2 : TXMB32 / TXQ3 : TXMB63		
DMA IF	TXQ0, TXQ3		

The register of parameter specification

Handling of the register of parameter specification is performed as follows.

1. The address of a register is the same between parameter specifications.
The address of the register deleted is empty space.
2. The bit of a register does not modify a bit position.
3. When accessing the empty address of the register deleted,
register writing is invalid and read is set to 0.
4. RAM data will be destroyed if it writes in addition to the register part of RAM area.
Moreover, read-out is don't care.

Modification of the software between parameter specifications.

When it is common and uses software between parameter specifications, consideration is needed for software in the following parts. For example, when the software of the parameter of two channels and 16TXMB buffer is applied to the parameter of eight channels and 64TXMB buffer

1. RAM capacity changes with parameter specification.

- 1) RAM initialization time changes.

When initialization time is measured, change of software is needed.

The RAM initialization cycle time in each parameter is indicated to Table 13.2.

Refer to the CFDGIPV register for a parameter setup value.

b[29:27]: Number of channels

b[26:24]: Number of TXMB's

b[20:17] :Number of Pool buffer

Software needs to take into consideration that the number of initialization cycles of RAM is modified.

- 2) RAM test space changes.

Since RAM capacity changes, a RAM test page is modified.

The RAM area in each parameter is indicated to Table 13.1.

Refer to the CFDGIPV register for a parameter setup value.

b[29:27]: Number of channels

b[26:24]: Number of TXMB's

b[23:21] :Number of AFL entries

b[20:17] :Number of Pool buffers

RAM area is calculated from information and it asks for a RAM test page.

Software needs to take the calculated RAM test page into consideration.

3) RAM address of a storage location changes.

When confirming by a RAM test mode, since addresses differ,
the setting value of a RAM test page is modified.

The RAM area in each parameter is indicated to Table 13.1.

Refer to the **CFDGIPV** register for a parameter setup value.

b[29:27]: Number of channels

b[26:24]: Number of TXMB's

b[23:21] :Number of AFL entries

b[20:17] :Number of Pool buffers

RAM area is calculated from information and it asks for a RAM test page.

Software needs to take the calculated RAM test page into consideration.

2. Number of TXMBs changes with a parameter

- 1) When TXQ1 and 3 are used, since the number of TXMB(s) changes,
the buffer numbers stored in THL differ.

When the CAN frame is transmitted by TXQ, the buffer number range of a TXMB parameter is shown below.

TXMB parameter	TXQ1			TXQ3			TXQ1 (Flexible TXMB assignment)		
	from	to (min)	to (max)	from	to (min)	to (max)	from	to (min)	to (max)
64TXMB	TXMB31	TXMB29	TXMB0	TXMB63	TXMB61	TXMB32	TXMB95	TXMB93	TXMB64
32TXMB	TXMB15	TXMB13	TXMB0	TXMB47	TXMB45	TXMB32	TXMB79	TXMB77	TXMB64
16TXMB	TXMB7	TXMB5	TXMB0	TXMB39	TXMB37	TXMB32	TXMB71	TXMB69	TXMB64

Software needs to take into consideration that a buffer number is modified into the buffer range of a table.

- 2) When borrowing and using TXQ1 by a "Flexible transmission buffer assignment function", buffer quantity to borrow is made into the maximum.

Therefore, when borrowing and using TXQ1, it is necessary to modify the buffer quantity borrowed according to a TXMB parameter.

3. Number of channels changes with a parameter

- 1) When transmitting from FIFO using Interval timer, a worst time changes with number of channels.
Number of channels and the relation of a worst time are as follows.

Channel parameter	Worst time
2 channel	432 pclk cycle
4 channel	672 pclk cycle

6 channel	912 pclk cycle
8 channel	1152 pclk cycle

2) Moreover, the entry time of THL also changes with number of channels.

Number of channels and the relation of entry time are as follows.

Channel parameter	Entry time
2 channel	200 pclk cycle
4 channel	208 pclk cycle
6 channel	216 pclk cycle
8 channel	224 pclk cycle

The software needs to take into consideration that processing time is modified as if a channel changes.

The expanding specification of a parameter is shown below.

Parameter Item	Unit	IP A	IP B	IP C	IP D	IP E	IP E FCC	IP F	IP G
Number of channel	ch/macro	8	8	4	6	8	8	2	2
Number of TXMB	msg/ch	64	32	32	32	64	64	32	16
Number of AFL entry	ID/ch	128	128	128	128	192	192	128	64
Number of Pool Buffer	msg/ch	128	64	64	64	256	256	64	32
Same ID overwrite function of TXQ		ON	ON	ON	ON	ON	ON	ON	OFF
FCC parameter function		OFF	OFF	OFF	OFF	OFF	ON	OFF	OFF

FCC parameter function becomes a function only for U2B-FCC.

U2B-FCC is a chip for evaluation.

RS-CANFD mounted in U2B-FCC chip needs to realize the same function as each parameter.

So, the "FCC parameter function" parameter for exclusive use was prepared.

A setup of this parameter will add the following terminals.

If parameter values are inputted into these terminals, it will become a function equivalent to the parameter specification.

Port Name	bit width	set value	Function
num_chan	[2:0]	3'b000	Number of channels is 8.
		3'b010	Number of channels is 6.
		3'b100	Number of channels is 4.
		3'b110	Number of channels is 2.
		other	Reserved (don't set)

<i>num_pool</i>	[3:0]	<i>4'b0000</i>	<i>Number of pool buffers is 256 per channel.</i>
		<i>4'b0001</i>	<i>Number of pool buffers is 128 per channel.</i>
		<i>4'b0010</i>	<i>Number of pool buffers is 64 per channel.</i>
		<i>4'b0011</i>	<i>Number of pool buffers is 48 per channel.</i>
		<i>4'b0100</i>	<i>Number of pool buffers is 32 per channel.</i>
		<i>other</i>	<i>Reserved (don't set)</i>
<i>num_afl</i>	[2:0]	<i>3'b000</i>	<i>Number of AFL entries is 192 per channel.</i>
		<i>3'b001</i>	<i>Number of AFL entries is 128 per channel.</i>
		<i>3'b010</i>	<i>Number of AFL entries is 64 per channel.</i>
		<i>other</i>	<i>Reserved (don't set)</i>
<i>num_txmb</i>	[2:0]	<i>3'b000</i>	<i>Number of TXMBs is 64 per channel.</i>
		<i>3'b001</i>	<i>Number of TXMBs is 32 per channel.</i>
		<i>3'b010</i>	<i>Number of TXMBs is 16 per channel.</i>
		<i>other</i>	<i>Reserved (don't set)</i>
<i>txq_idow</i>	-	0	<i>The same ID overwrite function of TXQ can be used.</i>
		1	<i>The same ID overwrite function of TXQ cannot be used.</i>

3 Interface Specification

3.1 Interconnection

The below figure shows the interconnectivity of the RS-CAN-FD IP:

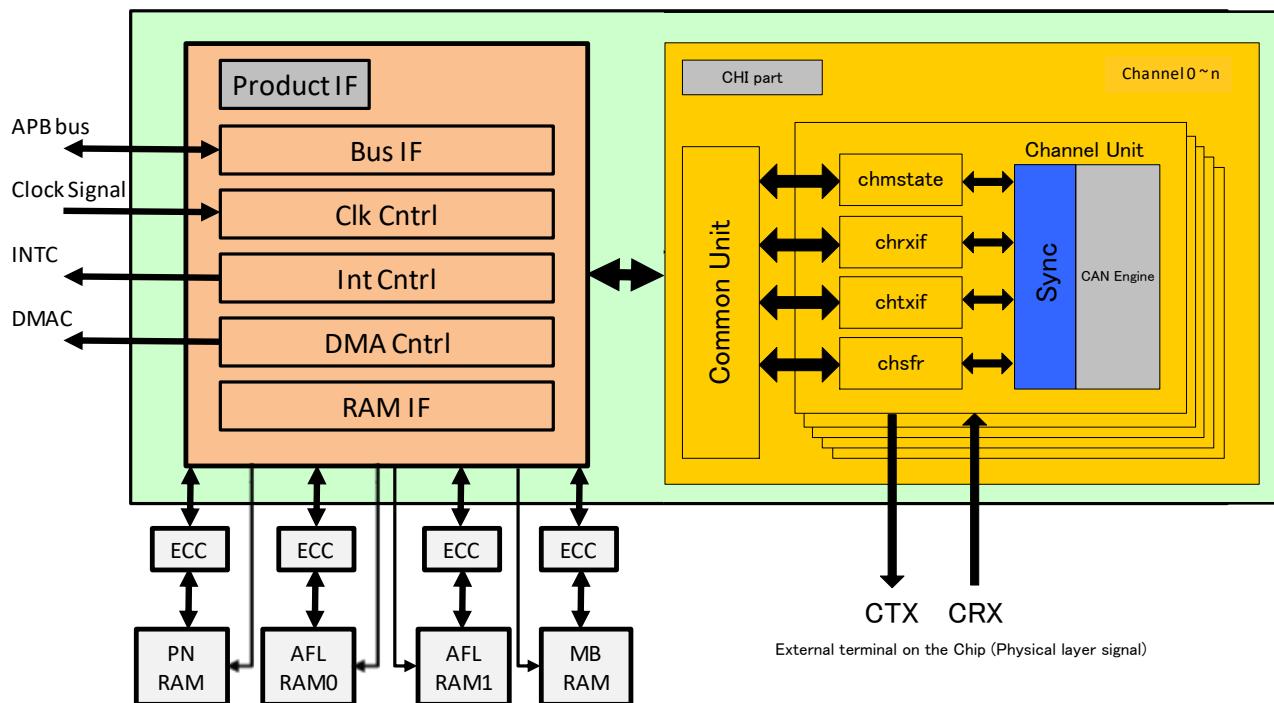


Figure 3.1 RS-CAN-FD IP connectivity

3.2 Interface signals

3.2.1 Reset and Clock control

Signal name	IO	Function	Internal connection	Interface	Initial value	Active level
rstp_n	in	Asynchronous reset (Active Low) (pclk clock domain)	rst_n	Reset control	low	low
rstc_n	in	Asynchronous reset (Active Low) (clkc clock domain)	rst_dllc_n	Reset control	low	low
pclk	in	Peripheral (APB bus) clock	pclk	Clock control	-	-
clk_ram	in	RAM access clock	clk_ram	Clock control	-	-
clkc	in	Clean clock with small clock jitter tolerance sufficient for CAN protocol specification requirements	clkc_dllc	Clock control	-	-
clk_xincan	in	External oscillator clock used for the CAN engine	clka_can	Clock control	-	-
scan_mode	in	Scan mode enable 0: Disabled 1: Enabled	scan_mode	Test control	-	high
scan_enable	in	Clock Gating cell bypass enable for SCAN TEST 0: Capture phase 1: Scan shift	scan_enable	Test control	-	High
test_mode	in	Disable RAM initialisation during test mode 0: RAM init. enabled 1: RAM init. disabled	test_mode	Test control	-	High

Table 3.1 Interface signals for the Reset and Clock control

Note: Initial value '-' means "don't care".

3.2.1.1 Signal explanation

3.2.1.1.1 rstp_n

Reset signal for all FFs of the whole IP, except the Channel Unit (CAN engine).

If rstp_n is 1'b0, all FFs of the whole IP (except the Channel Unit) are reset asynchronously (see Figure. 3.3). It is expected that this input reset signal is released synchronously with pclk from the chip side.

3.2.1.1.2 rstc_n

Reset signal for all FFs of only the CAN engine.

If rstc_n is 1'b0, all FFs of the CAN engine are reset asynchronously (see Figure. 3.3). It is expected that this input reset signal is released synchronously with clkc from the chip side when the PLL is used. Even if the PLL is not used, this input reset signal must be released to start the CAN engine. In this case, this reset input signal must be released synchronously with pclk from the chip side.

3.2.1.1.3 pclk

This is the peripheral clock, which is used for the APB bus.

3.2.1.1.4 clkc

Clean clock with small clock jitter tolerance sufficient for CAN protocol specification requirements.

The clean clock frequency should be less than or equal to the peripheral clock frequency.

In SCAN mode the peripheral clock pclk is used. For this the clean clock clkc and the peripheral clock pclk are connected via a clock multiplexer and clock divider to reach also in the SCAN mode a clock input, which is not faster as the used clock in the RS-CAN-FD IP.

3.2.1.1.5 clk_xincan

External oscillator clock used for the CAN engine

This clock is coming from the input pin xin.

The External oscillator clock frequency should be less than or equal to the peripheral clock frequency.

3.2.1.1.6 scan_mode

SCAN mode enable signal to use during SCAN TEST.

3.2.1.1.7 scan_enable

Clock Gating cell bypass enable for SCAN TEST to use during SCAN TEST.

3.2.1.1.8 test_mode

This signal is used to disable the IP RAM initialisation to avoid conflict of production test and RAM initialisation routine during the production test. When the signal is cleared the IP should be in Reset mode (rstp_n and rstm_n both zero).

3.2.1.1.9 test_mode

This signal is used to disable the IP RAM initialisation to avoid conflict of production test and RAM

3.2.1.1.10 clk_ram

This is the RAM clock, which is used for the MRAM, AFLRAM and PFLRAM.

The RAM clock frequency should be the double frequency of the peripheral clock pclk and should be in phase with pclk.

3.2.1.2 Clock tree

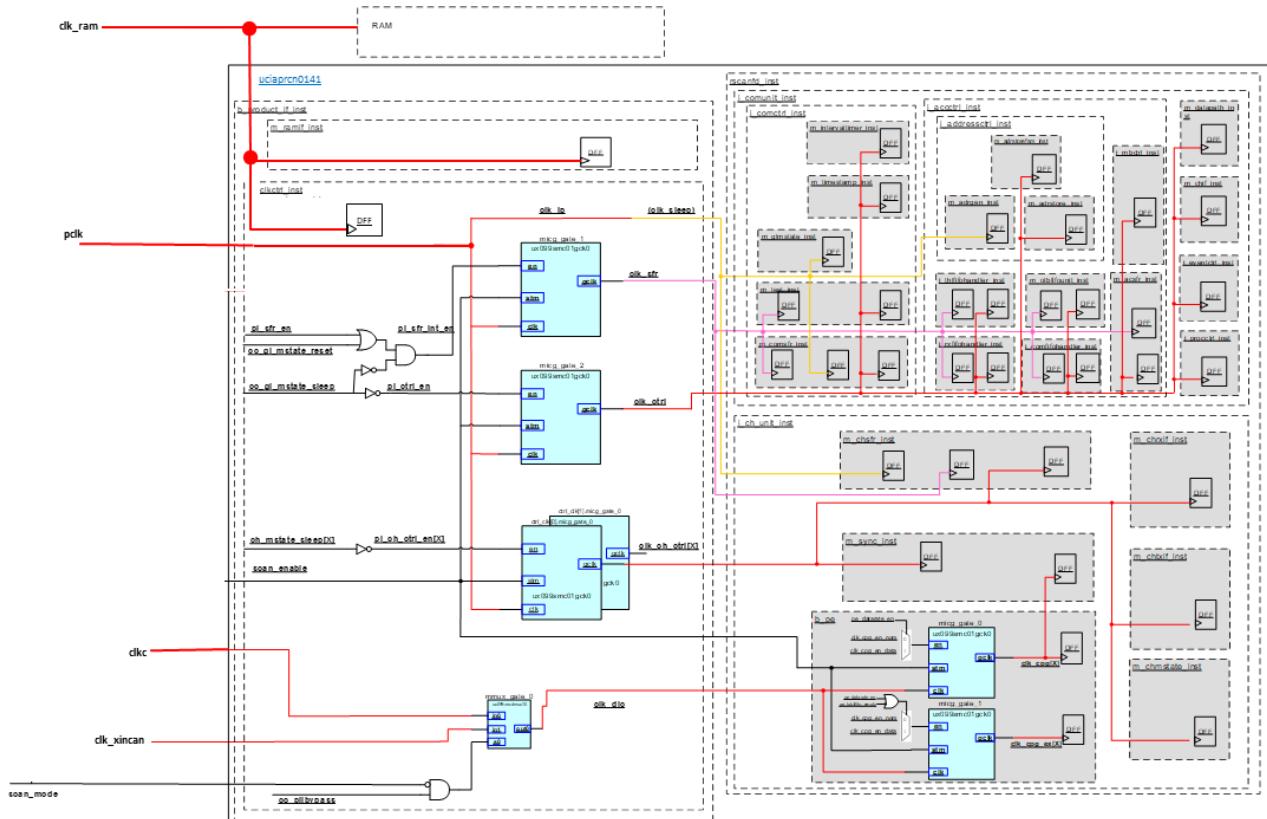


Figure 3.2 Clock tree

3.2.1.3 Reset synchronisation circuit

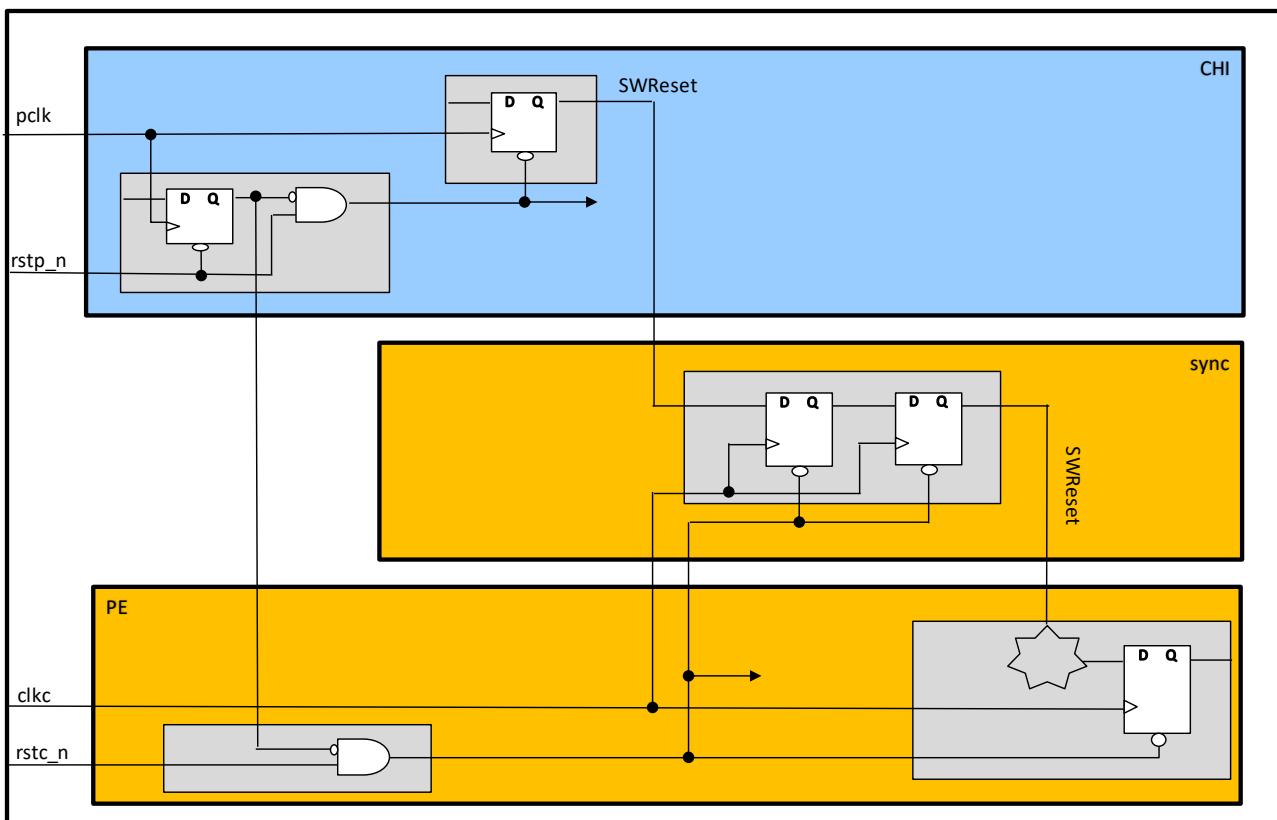


Figure 3.3 Reset synchronisation circuit

Both clock domains have their own reset signals:

`rstp_n` is synchronized to `clk_ctrl` (coming from `pclk`);
`rstc_n` is synchronized to `clk_dlc` (coming from `clk`);

Both resets `rstp_n` and `rstc_n` should always be activated/released together. Only a small gap due to reset synchronization in different clock domain is expected. Otherwise flagging updates of SFR register is inconsistent.

3.2.1.4 RAM initialisation

The RAM initialisation starts within `pclk` cycles of releasing the `rstp_n` (`rstp_n` changes from low to high). During RAM initialisation, the whole RAMs of maximum 12288 Bytes of the AFL RAM0, maximum 12288 Bytes of the AFL RAM1, maximum 5760 Bytes of the PFL RAM and maximum 197888 Bytes of the Message Buffer RAM are written with the value 00h.

The time in `pclk` cycles to initialize the RAM is calculated as follows:

$$197888 / 4 + 2 = 49474 \text{ pclk cycles}$$

The RAM initialisation is implemented as a kind of pipelining write, therefore the access required only 2 `clk_sleep` cycle (refer to Figure. 3-2). Because of this the initialisation of the whole RAM area 197888 Bytes takes 49474 `pclk` cycles.

Note the AFL RAM and PFL RAM are initialised in parallel with the Message Buffer RAM, but the size is smaller, hence the initialisation time comes from the Message Buffer RAM.

(RAM initialisation time changes according to parameter conditions.)

During the RAM initialisation, users should not write to the CAN SFR.

3.2.1.5 Clock restriction

For the CAN communication the following restriction for the clocks should be satisfied:

clk_ram / 2 = pclk ≥ clkc

clk_ram / 2 = pclk ≥ clk_xincan

To avoid missing events the CAN engine clock (clkc or clk_xincan) frequency must be less than the pclk clock frequency.

To avoid loss of CAN messages, the pclk should be connected to a clock with a frequency depending on the CAN communication Baud Rate. The maximum constraint of a baud rate and a pclk clock is shown in a table. Please use it below by the baud rate of this table.

Max channel	Baud rate		pclk	clk_ram
8 channel	1 Mbps Nominal 5Mbps Data	Condition	CHI (pclk) ≥ 80 MHz	clk_ram ≥ 160MHz
		Minimum frequency	80 MHz	160MHz
6 channel	1 Mbps Nominal 8Mbps Data	Condition	CHI (pclk) ≥ 80 MHz	clk_ram ≥ 160MHz
		Minimim frequency	80 MHz	160MHz

Table 3.2 Clock restriction

The min. frequency of clkc and clk_xincan depend on the required Baud Rate. For information how to configure the Baud Rate, please refer to Section 6.1.3.

Note: The operating frequency of the IP is not considering ECC errors. In case of 2bit ECC error during TX-SCAN the scan is aborted and restarted. This continued restart without user interaction could result in event operation overload.

3.2.2 APB bus control

The bus interface signals conform to AMB APB2.0 and also conform to the APB-I/F Rule for M40 Platform Ver.4.0.

Signal name	IO	Function	Internal connection	Interface	Clock domain	Initial value	Active level
psel	in	APB select signal 0: Not selected 1: Selected	psel	APB control	pclk	-	high
paddr[16:0]	in	APB address bus	paddr[16:0]	APB control	pclk	-	-
penable	in	APB enable signal 0: cycle 0 1: second and subsequent cycle	enable	APB control	pclk	-	high
pwrite	in	APB write signal 0: read 1: write	pwrite	APB control	pclk	-	high
pstrb[3:0]	in	APB Byte enable signal	pstrb[3:0]	APB control	pclk	-	-
pwdata[31:0]	in	APB Input Data bus	pwdata[31:0]	APB control	pclk	-	-
prdata[31:0]	out	APB Output Data bus	prdata[31:0]	APB control	pclk	All low	-
pready	out	APB ready signal Bus access extend signal 0: access on-going 1: access completed	pready	APB control	pclk	high	high

Table 3.3 Interface signals for APB control

Note: Initial value ‘-’ means “don’t care”.

The output signal PSLVERR is not required for the RS-CAN-FD IP and because of this it is not generated and the output pin is not available at the RS-CAN-FD IP.

3.2.2.1 Signal explanation

3.2.2.1.1 psel

APB select signal

3.2.2.1.2 paddr[16:0]

APB address bus

3.2.2.1.3 penable

APB enable signal

The signal indicates the second and subsequent cycles of both read and write access.

3.2.2.1.4 pwrite

APB write signal

The signal indicates a read access by 1'b0 and a write access by 1'b1.

3.2.2.1.5 pstrb[3:0]

APB Byte enable signal

3.2.2.1.6 pwdata[31:0]

APB write data bus

The value on pwdata will be written to the RAM or internal register.

3.2.2.1.7 prdata[31:0]

APB read data bus

When the APB bus requests data from the IP, the output data occurs on prdata.

3.2.2.1.8 pready

APB ready signal

The RS-CAN-FD IP will use this signal to extend an APB access. This will be used for APB read access from the RAM.

3.2.2.2 Additional Interface information

The table below shows the needed number of APB access cycle related to target area (SFR or RAM) and the access width (for detailed refer to Section. 3.2.5).

	APB read access		APB write access	
	SFR	RAM	SFR	RAM
Access width (bits)	32	32	8, 16, 32	8, 16
Number of peripheral bus clock (pclk) cycles	2	3	2	2

Table 3.4 APB bus access cycle length

3.2.3 Interrupt Interface

All interrupt signals from the RS-CAN-FD IP are level signal with high active level. These signals are not cleared automatically. To clear an interrupt line it is necessary to clear the interrupt flag in the SFR during the interrupt routine.

A global interrupt can only occur if the RS-CAN-FD IP is in the Global Operation or Global Halt mode. A channel interrupt can only occur when the related CAN channel is in Channel Operation mode.

Signal name	IO	Function	Internal connection	Interface	Clock domain	Initial value	Active level
can_cherr_int[n:0]	out	Channel Error Interrupt lines	can_cherr_int[n:0]	Interrupt control	pclk	All low	high
can_comfrx_int[n:0]	out	COM RX FIFO or TXQ Interrupt lines	can_comfrx_int[n:0]	Interrupt control	pclk	All low	high
can_glerr_int	out	Global Error Interrupt line	can_glerr_int	Interrupt control	pclk	low	high
can_rxf_int	out	RX FIFO interrupt line	can_rxf_int	Interrupt control	pclk	low	high
can_tx_int[n:0]	out	Channel TX interrupt lines	can_tx_int[n:0]	Interrupt control	pclk	All low	high
can_vmtx_int[n:0]	out	Virtual Machine TX interrupt lines	can_vmtx_int[n:0]	Interrupt control	pclk	All low	high
can_vmrx_int[n:0]	out	Virtual Machine RX interrupt lines	can_vmrx_int[n:0]	Interrupt control	pclk	All low	high
can_vmerr_int[n:0]	out	Virtual Machine Error interrupt lines	can_vmerr_int[n:0]	Interrupt control	pclk	All low	high

Table 3.5 Interface signals for Interrupt control

Note: Initial value ‘-’ means “don’t care”.

3.2.3.1 Signal explanation

3.2.3.1.1 can_cherr_int[n:0]

Channel CAN error interrupt lines

When a CAN error occurs, the channel related interrupt signal is set to high level.

The channel error interrupt line for every channel is the logical OR combination of all CAN error flags.

3.2.3.1.2 can_comfrx_int[n:0]

COM RX FIFO or TXQ interrupt lines

Interrupt lines for the Common FIFOs in RX or GW mode or TXQ in TX queue GW mode belonging to the related channel.

3.2.3.1.3 can_glerr_int

Global Error Interrupt line

If a DLC error flag is set or a message lost status bit is set or a message overwrite status bit is set or a TX History Entry Lost status bit is set or a message payload overflow flag bit is set, this interrupt line is set to high level.

3.2.3.1.4 can_rxf_int

RX FIFO interrupt line

This is the interrupt line for the interrupt flag of corresponding RX FIFO for which the interrupt is enabled.

3.2.3.1.5 can_tx_int[n:0]

Channel TX interrupt lines

The logical output of the TX flags related to the channel.

3.2.3.1.6 can_vmtx_int[n:0]

Virtual Machine TX interrupt lines

The transmitting interruption output of related virtual machine.

When FFI mode, this interruption signal becomes valid.

3.2.3.1.7 can_vmrx_int[n:0]

Virtual Machine RX interrupt lines

The reception interruption output of the related virtual machine.

When FFI mode, this interruption signal becomes valid.

3.2.3.1.8 can_vmerr_int[n:0]

Virtual Machine Error interrupt lines

The error interruption output of the related virtual machine.

When FFI mode, this interruption signal becomes valid.

3.2.3.2 Additional Interface information

The following figure shows the interrupt generation and clearing timing:

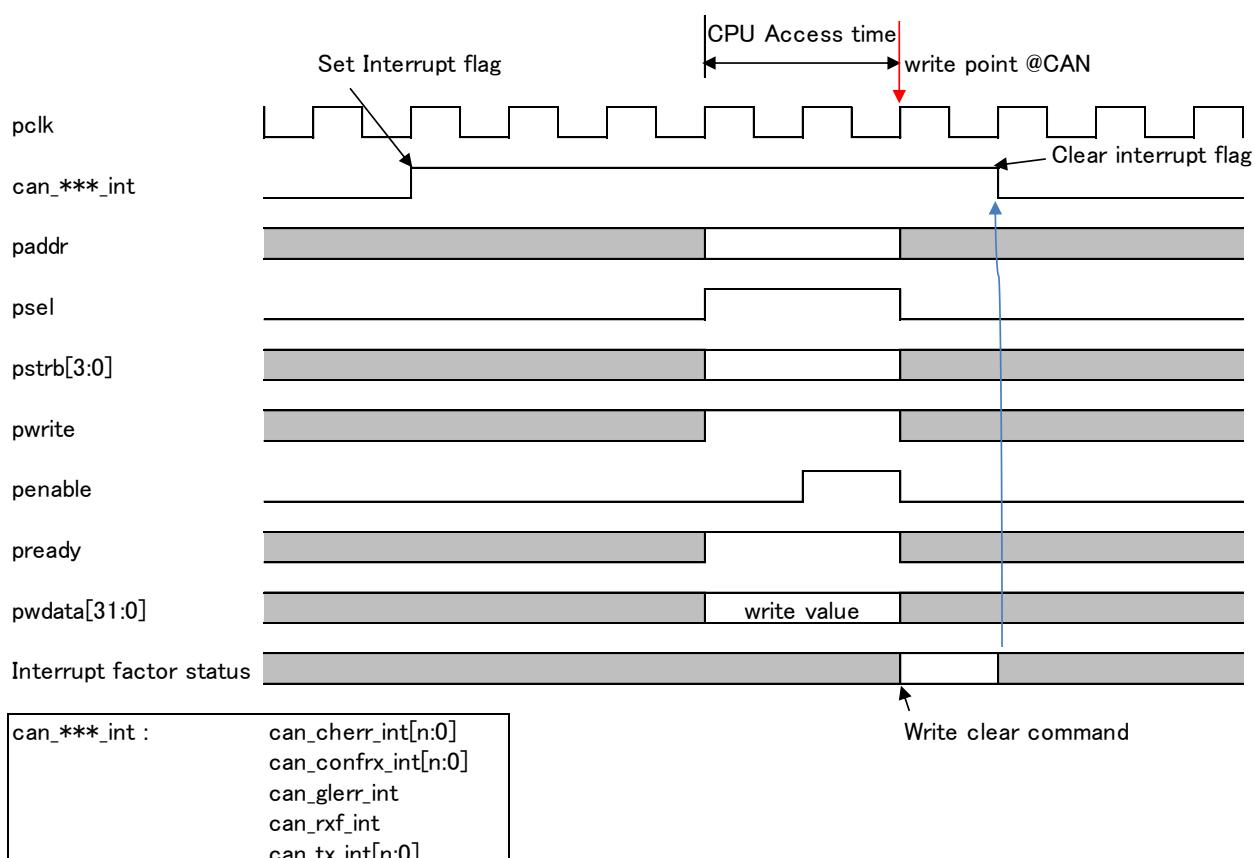


Figure 3.4 Interrupt signal generation and clear timing

3.2.4 DMA Interface

All DMA request signals from the RS-CAN-FD IP are pulse signals with high active level.

Signal name	IO	Function	Internal connection	Interface	Clock domain	Initial value	Active level
can_rf_dmareq[7:0]	out	RX FIFO DMA request	can_rf_dmareq[7:0]	DMAC	pclk	low	high
can_cf_dmareq[n:0]	out	CON FIFO DMA request	can_cf_dmareq[n:0]	DMAC	pclk	low	high

Table 3.6 Interface signals for the DMAC

3.2.4.1 Signal explanation

3.2.4.1.1 can_rf_dmareq[7:0]

This is the DMA transfer request pulse signal for the DMAC of the RX-FIFOs that is set high for 1 pclk cycle.

If no DMAC is used, this RS-CAN-FD IP port should be kept open.

3.2.4.1.2 can_cf_dmareq[n:0]

This is the DMA transfer request pulse signal for the DMAC of the first COM-FIFO of each channel that is set high for 1 pclk cycle.

If no DMAC is used, this RS-CAN-FD IP port should be kept open.

3.2.5 Port Interface and mode select signal

Signal name	IO	Function	Internal connection	Interface	Clock domain	Initial value	Active level
rxd_can[n:0]	in	CAN Bus reception line	rxd_can[n:0]	Port	asynchronous	-	-
<i>rs_canfd_non_iso_sel</i>	In	<i>CAN mode selection source</i>	<i>rs_canfd_non_iso_sel</i>	Port	static	-	-
<i>rs_canfd_non_iso_en</i>	in	<i>ISO mode selection</i>	<i>rs_canfd_non_iso_en</i>	Port	static	-	low
num_ram_chan[3:0]	in	Number of RAM channels	pi_ram_chan[3:0]	Port	static	-	-
can_txclk[n:0]	out	CAN Bit time clock, debug observation signal	can_txclk[n:0]	Port	clkc / clk_xincan	low	-
can_tx_out[n:0]	out	CAN Bus transmission line	can_tx_out[n:0]	Port	clkc / clk_xincan	high	-
can_tx_datarate_en[n:0]	out	CAN-FD Data phase transmission	can_tx_datarate_en[n:0]	Port	clkc / clk_xincan	low	high
can_rx_datarate_en[n:0]	out	CAN-FD Data phase reception	can_rx_datarate_en[n:0]	Port	clkc / clk_xincan	low	high
<i>rs_canfd_bus_if_sel</i>	In	<i>Bus interface selection source</i>	<i>rs_canfd_bus_if_sel</i>	Port	static	-	-
<i>rs_canfd_bus_if_en</i>	in	<i>Bus interface selection</i>	<i>rs_canfd_bus_if_en</i>	Port	static	-	low
rs_classic_only	in	Classical CAN only mode	rs_classic_only	Port	static	-	high
rs_canfd_tol_off	in	FD tolerant mode off	rs_canfd_tol_off	Port	static	-	high
rs_vnnum_fix	in	Virtual machine number interrupt fix	rs_vnnum_fix	Port	static	-	high
<i>num_chan[2:0]</i>	in	<i>Number of channels of FCC parameter</i>	<i>num_chan[2:0]</i>	Port	static	-	high
<i>num_pool[3:0]</i>	in	<i>Number of pool buffers of FCC parameter</i>	<i>num_pool[3:0]</i>	Port	static	-	high
<i>num_afl[2:0]</i>	in	<i>Number of afl entries of FCC parameter</i>	<i>num_afl[2:0]</i>	Port	static	-	high
<i>num_txmb[2:0]</i>	in	<i>Number of txmbs of FCC parameter</i>	<i>num_txmb[2:0]</i>	Port	static	-	high
<i>txq_idow</i>	in	<i>TXQ same ID overwrite function of FCC parameter</i>	<i>txq_idow</i>	Port	static	-	high
can_race_ts[(n+1)*16-1:0]	in	Address in which Timestamp of TSCapture is stored is inputted	can_race_ts[(n+1)*16-1:0]	Port	pclk	-	-
can_race_ts_en[n:0]	in	The enabling signal of External Timestamp of TSCapture	can_race_ts_en[n:0]	Port	pclk	-	high
can_rx_ts_capture[n:0]	out	Timestamp capture signal of the received CAN frame	can_rx_ts_capture[n:0]	Port	pclk	low	high
can_tx_ts_capture[n:0]	out	Timestamp capture signal of the transmitted CAN frame	can_tx_ts_capture[n:0]	Port	pclk	low	high

Table 3.7 Interface signals for the Port interface

Note: Initial value ‘-’ means “don’t care”;

can_txclk is the clock domain for the CAN Bus transmission line, means can_tx_out can only change if can_txclk is high (see Figure. 3-5);

3.2.5.1 Signal explanation

3.2.5.1.1 rxd_can[n:0]

This signal is the receive signal for the CAN communication. This signal needs to be connected to the external port.

This input signal is asynchronously to the internal clocks.

If it is not used, this RS-CAN-FD IP port should be connected to high level.

3.2.5.1.2 rs_canfd_non_iso_sel

This signal selects the source for the CAN-FD CRC Sequence selection. The selection can be done either by a module pin or by a module internal register. For more information refer to table 3.8.

This input signal should be static.

3.2.5.1.3 rs_canfd_non_iso_en

This signal is the CAN-FD CRC Sequences selector input pin. This signal should be connected to either an external mode control register or, depending on the product requirements, connected to either 1'b0 or 1'b1. Depending on the rs_canfd_non_iso_sel input either rs_canfd_non_iso_en or CFDGCRCFCFG.NIE register bit is used for selection as shown in the table below.

rs_canfd_non_iso_sel	rs_canfd_non_iso_en	CFDGCRCFCFG.NIE	CRC Sequences format
1'b0	1'b0	1'bx	<i>CAN-FD ISO 11898-1 (2015)</i>
1'b1	1'bx	1'b0	
1'b0	1'b1	1'bx	<i>Bosch Specification: CAN with Flexible Data-Rate Ver.1.0</i>
1'b1	1'bx	1'b1	<i>(RS-CAN-FD version 1.0 compatible mode)</i>

Table 3.8 CAN-FD CRC Sequences format configuration

This input signal should be static.

3.2.5.1.4 num_ram_chan[3:0]

This signal is the number of RAM channels configuration input pin. This signal should be connected according to the product requirements

Input: num_ram_chan[3:0]	Specification	Output: pi_ram_chan[3:0]
4'b0000	As synthesised	`CAN_NUM_CH
4'b(n+1)	n+1 channel	4'b(n+1)

CAN_NUM_CH: Channel Parameter Value

Table 3.9 num_ram_chan configuration

Note: Users should configure values not greater than n+1.

This input signal should be static.

If this input is set with the number of channels less than the implemented channel number, SW compatibility with RS-CAN in test mode is lost.

3.2.5.1.5 can_txclk[n:0]

This signal is the CAN bit time clock.

In normal communication this signal is not used. In the past it was used to observe the bit time clock for debug purpose; if this is wanted, this signal needs to be connected to an external port.

If it is not used, this RS-CAN-FD IP port should be kept open.

3.2.5.1.6 can_tx_out[n:0]

This signal is the transmitted signal of the CAN communication. This signal needs to be connected to the external port.

This signal is synchronised with the CAN bit time clock (can_txclk).

Because for CAN communication the dominant level is 1'b0 and the recessive level is 1'b1, it must be taken care, that the CAN bus can be destroyed by outputting low level.

3.2.5.1.7 can_tx_datarate_en[n:0]

This signal indicates the transmitting data phase of a CAN-FD frame. It follows the definition of a Data Phase as specified in the ISO 11898-1 (2015). This signal needs to be connected to the external port.

This signal is synchronised with the CAN protocol clock (clkc / clk_xincan). The signal is delayed by 1 TQ due to the registered output drive.

3.2.5.1.8 can_rx_datarate_en[n:0]

This signal indicates the reception data phase of a CAN-FD frame. It follows the definition of a Data Phase as specified in the ISO 11898-1 (2015). This signal needs to be connected to the external port.

This signal is synchronised with the CAN protocol clock (clkc / clk_xincan). The signal is delayed by 1 TQ due to the registered output drive.

3.2.5.1.9 rs_canfd_bus_if_sel

This signal is used to decide the RS-CAN-FD Bus interface.

The signal selects either by rs_canfd_bus_if_en pin or by CFDGBISC.IFSW.

For more information refer to table 3.9.

This signal should be tied to 0 or 1 statically. (0: rs_canfd_bus_if_en, 1: CFDGBISC.IFSW)

3.2.5.1.10 rs_canfd_bus_if_en

This signal is the CAN-FD Bus interface selector input pin.

This signal should be connected to either a Bus Interface control register or, depending on the product requirements, connected to either 1'b0 or 1'b1.

Depending on the rs_canfd_bus_if_sel input either rs_canfd_bus_if_en or CFDGBISC.IFSW register bit is used for selection as shown in the table below.

Rs_canfd_bus_if_sel	rs_canfd_bus_if_en	CFDGBISC.IFSW	RS-CAN-FD Bus Interface
1'b0	1'b0	1'bx	<i>APB Interface</i>
1'b1	1'bx	1'b0	
1'b0	1'b1	1'bx	<i>R-ACE interface</i>
1'b1	1'bx	1'b1	

Table 3.10 CAN-FD Bus Interface configuration

This input signal should be tied to 0 or 1 statically.

3.2.5.1.11 rs_classic_only

This signal forces all channel to change to Classical CAN only mode.

3.2.5.1.12 rs_canfd_tol_off

This signal forces all channel to disable FD tolerant mode.

If rs_classic_only=0, this signal should not set

These signals (rs_classic_only and rs_canfd_tol_off) are the classic only mode configuration input pin.

These signals should be connected to either an external mode control register or, depending on the product



requirements, connected to either 1'b0 or 1'b1.

CAN MODE is controlled as shown in the table below.

set value				selected mode	read value	
rs_classic_only	rs_canfd_tol_off	CnFDCFG.C LOE	CnFDCFG.CF DTE		CnFDCFG.C LOE	CnFDCFG.CF DTE
0	0	0	0	RS-CAN-FD mode	0	0
		1	0	classic only mode	1	0
		1	1	FD tolerant mode	1	1
		0	1	Prohibition		
1	1	not open	not open	classic only mode	not open	not open
1	0	not open	0	classic only mode	not open	0
			1	FD tolerant mode		1
0	1	0	x	Prohibition		
		1		Prohibition		

Table 3.11 CAN-FD mode configuration

If rs_classic_only = 1, User can access to CFDCnFDCFG.FDOE. But this bit value is ignored.

If rs_classic_only = 1 and FD tolerant is disable, User can access to **CFDCnFDCFG.REFE**. But this bit value is ignored.

3.2.5.1.13 rs_vnum_fix

The value of **CFDVMCFGn** is fixed to the number of a channel.

0: FFI mode is controlled per buffer.

1: FFI mode is fixed per channel.

3.2.5.1.14 num_chan

When FCC parameter, this signal controls number of channels.

3.2.5.1.15 num_pool

When FCC parameter, this signal controls number of pool buffers.

3.2.5.1.16 num_afl

When FCC parameter, this signal controls number of AFL entries.

3.2.5.1.17 num_txmb

When FCC parameter, this signal controls number of TXMBs.

3.2.5.1.18 txq_idow

When FCC parameter, this signal controls TXQ same ID overwrite function.

Port Name	bit width	set value	Function
num_chan	[2:0]	3'b000	Number of channels is 8.
		3'b010	Number of channels is 6.
		3'b100	Number of channels is 4.
		3'b110	Number of channels is 2.
		other	Reserved (don't set)
num_pool	[3:0]	4'b0000	Number of pool buffers is 256 per channel.
		4'b0001	Number of pool buffers is 128 per channel.
		4'b0010	Number of pool buffers is 64 per channel.
		4'b0011	Number of pool buffers is 48 per channel.
		4'b0100	Number of pool buffers is 32 per channel.
		other	Reserved (don't set)
num_afl	[2:0]	3'b000	Number of AFL entries is 192 per channel.
		3'b001	Number of AFL entries is 128 per channel.
		3'b010	Number of AFL entries is 64 per channel.
		other	Reserved (don't set)
num_txmb	[2:0]	3'b000	Number of TXMBs is 64 per channel.
		3'b001	Number of TXMBs is 32 per channel.
		3'b010	Number of TXMBs is 16 per channel.
		other	Reserved (don't set)
txq_idow	-	0	The same ID overwrite function of TXQ can be used.
		1	The same ID overwrite function of TXQ cannot be used.

Table 3.12 contents of a setting of the terminal of an FCC parameter

3.2.5.1.19 can_race_ts [(n+1)*16-1:0]

Address in which Timestamp is stored is inputted. Refer gPTPTimeSync's IPSpec for details.

3.2.5.1.20 can_race_ts_en[n:0]

The enabling signal of External Timestamp. Refer gPTPTimeSync's IPSpec for details.

3.2.5.1.21 can_rx_ts_capture[n:0]

Receiving side timestamp capture signal. Refer gPTPTimeSync's IPSpec for details.

3.2.5.1.22 can_tx_ts_capture[n:0]

Transmitting side timestamp capture signal. Refer gPTPTimeSync's IPSpec for details.

3.2.5.2 Additional Interface information

3.2.5.2.1 Timing of CAN communication transmit signal

The following figure shows the timing of the CAN communication transmit signal:

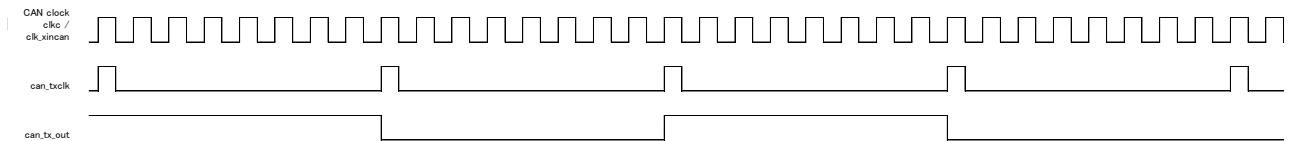


Figure 3.5 CAN communication transmit signal timing

3.2.5.2.2 Internal propagation delay for rxd_can and can_tx_out

To ensure the expected Baud Rate, especially at a higher rate such as 1 or 2 Mbps, it is necessary to minimize the internal propagation delay from the macro's data output pin (can_tx_out[n:0]) to the chip's data output pin.

Note: Customer would like to use no Transceiver Delay compensation feature for Baud Rates below 2 MBps.

In the same way, the internal propagation delay from the chip's data input pin to the macro's data input pin (rxd_can[n:0]) should be as small as possible.

3.2.5.2.3 Timing of data rate enable can_tx_datarate_en[n:0] / can_rx_datarate_en[n:0]

The next diagrams show the assertion condition.

BRS bit

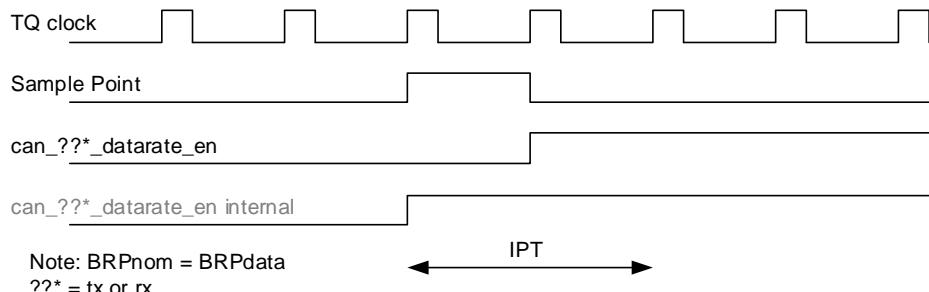


Figure 3.6 can_??*_datarate_en set in BRS bit

CRC delimiter bit

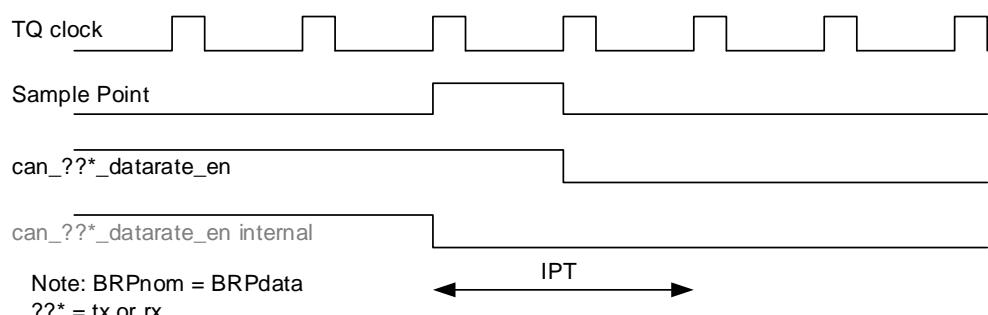


Figure 3.7 can_??*_datarate_en clear in CRC delimiter bit

Error detection

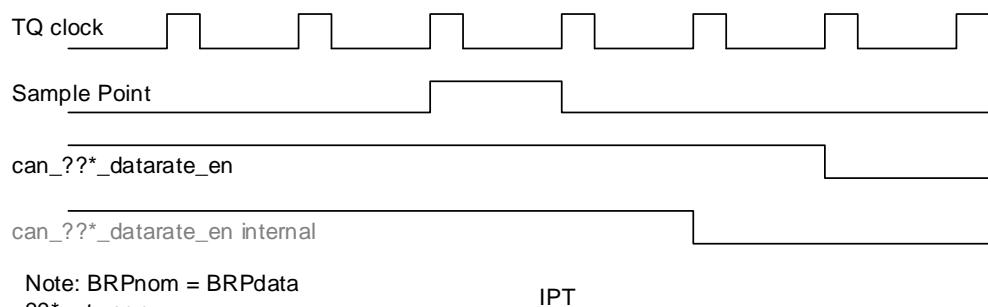


Figure 3.8 can_??*_datarate_en clear in case of Error

3.2.6 RAM Interface

Signal name	IO	Function	Internal connection	Interface	Clock domain	Initial value	Active level
afram_q[31:0]	in	AFL RAM0 read data	afram_q[31:0]	ECC	clk_ram	-	-
afram1_q[31:0]	in	AFL RAM1 read data	afram1_q[31:0]	ECC	clk_ram	-	-
can_addr_afram[11:0]	out	AFL RAM address	can_addr_afram[11:0]	ECC & RAM	clk_ram	All low	-
can_cen_afram_n	out	AFL RAM0 cell select signal	can_cen_afram_n	ECC & RAM	clk_ram	high	low
can_cen_afram1_n	out	AFL RAM1 cell select signal	can_cen_afram1_n	ECC & RAM	clk_ram	high	low
can_data_afram[31:0]	out	AFL RAM write data	can_data_afram[31:0]	ECC	clk_ram	All low	-
can_wen_afram_n	out	AFL RAM0 write enable signal	can_wen_afram_n	ECC & RAM	clk_ram	high	Low
can_wen_afram1_n	out	AFL RAM1 write enable signal	can_wen_afram1_n	ECC & RAM	clk_ram	high	Low
mram_q[31:0]	in	Message RAM read data	mram_q[31:0]	ECC	clk_ram	-	-
mram_ecc_err	in	ECC error status	mram_ecc_err	ECC	clk_ram	-	-
can_addr_mram[15:0]	out	Message RAM address	can_addr_mram[15:0]	ECC & RAM	clk_ram	All low	-
can_cen_mram_n	out	Message RAM cell select signal	can_cen_mram_n	ECC & RAM	clk_ram	high	low
can_data_mram[31:0]	out	Message RAM write data	can_data_mram[31:0]	ECC	clk_ram	All low	-
can_wen_mram_n	out	Message RAM write enable signal	can_wen_mram_n	ECC & RAM	clk_ram	high	Low
pfram_q[31:0]	in	PNF RAM read data	pfram_q[31:0]	ECC	clk_ram	-	-
can_addr_pfram[10:0]	out	PNF RAM address	can_addr_pfram[10:0]	ECC & RAM	clk_ram	All low	-
can_cen_pfram_n	out	PNF RAM cell select signal	can_cen_pfram_n	ECC & RAM	clk_ram	high	low
can_data_pfram[31:0]	out	PNF RAM write data	can_data_pfram[31:0]	ECC	clk_ram	All low	-
can_wen_pfram_n	out	PNF RAM write enable signal	can_wen_pfram_n	ECC & RAM	clk_ram	high	Low

Table 3.13 Interface signals for the ECC interface

Note: Initial value ‘-’ means “don’t care”;

3.2.6.1 Signal explanation

3.2.6.1.1 afram_q[31:0]

This is the data output signal of the CAN AFL RAM0.

It should be synchronized with clk_ram.

3.2.6.1.2 afram1_q[31:0]

This is the data output signal of the CAN AFL RAM1.

It should be synchronized with clk_ram.

3.2.6.1.3 can_addr_afram[11:0]

This is the address bus signal for the CAN AFL RAM.

It is synchronized by clk_ram and set by the CAN IP when the CAN AFL RAM is accessed.

The RAM is addressed always in Longword (32bit per address).

3.2.6.1.4 can_cen_afram_n

This is the RAM cell select signal for the CAN AFL RAM0.

It is low active, synchronized by clk_ram and is used to enable the CAN AFL RAM0 when CPU or the CAN IP is accessing to the CAN AFL RAM0.

3.2.6.1.5 can_cen_afram1_n

This is the RAM cell select signal for the CAN AFL RAM1.

It is low active, synchronized by clk_ram and is used to enable the CAN AFL RAM1 when CPU or the CAN IP is accessing to the CAN AFL RAM1.

3.2.6.1.6 can_data_afram[31:0]

This is the write data bus to the CAN AFL RAM.

It is synchronized by clk_ram and set by the CAN IP when the CAN AFL RAM is written.

3.2.6.1.7 can_wen_afram_n

This is the write enable signal for the CAN AFL RAM0.

It is low active, synchronized by clk_ram and is used when CPU or the CAN IP is writing to the CAN AFL RAM0.

From CAN IP the access width is 32-bit every time. From CPU theoretically 8- and 16-bit are also possible, but because of ECC usage, the RS-CAN-FD IP converts the CPU access in a 32-bit access every time.

3.2.6.1.8 can_wen_afram1_n

This is the write enable signal for the CAN AFL RAM1.

It is low active, synchronized by clk_ram and is used when CPU or the CAN IP is writing to the CAN AFL RAM1.

From CAN IP the access width is 32-bit every time. From CPU theoretically 8- and 16-bit are also possible, but because of ECC usage, the RS-CAN-FD IP converts the CPU access in a 32-bit access every time.

3.2.6.1.9 mram_q[31:0]

This is the data output signal of the CAN Message Buffer RAM.

It should be synchronized with clk_ram.

3.2.6.1.10 mram_ecc_err

This is the RAM ECC error input control signal.

It is high active, synchronised with clk_ram and is used to indicate the error status of the CAN Message Buffer RAM.

Refer to section 3.2.6.2.1 for a description of how to connect and use this signal.

3.2.6.1.11 can_addr_mram[15:0]

This is the address bus signal for the CAN Message Buffer RAM.

It is synchronized by clk_ram and set by the CAN IP when the CAN Message Buffer RAM is accessed.

The RAM is addressed always in Longword (32bit per address).

3.2.6.1.12 can_cen_mram_n

This is the RAM cell select signal for the CAN Message Buffer RAM.

It is low active, synchronized by clk_ram and is used to enable the CAN Message Buffer RAM when CPU or the CAN IP is accessing to the CAN Message Buffer RAM.

3.2.6.1.13 can_data_mram[31:0]

This is the write data bus to the CAN Message Buffer RAM.

It is synchronized by clk_ram and set by the CAN IP when the CAN Message Buffer RAM is written.

3.2.6.1.14 can_wen_mram_n

This is the write enable signal for the CAN Message Buffer RAM.

It is low active, synchronized by clk_ram and is used when CPU or the CAN IP is writing to the CAN

Message Buffer RAM.

3.2.6.1.15 pflram_q[31:0]

This is the data output signal of the CAN PFL RAM.

It should be synchronized with clk_ram.

3.2.6.1.16 can_addr_pflram[10:0]

This is the address bus signal for the CAN PFL RAM.

It is synchronized by clk_ram and set by the CAN IP when the CAN PFL RAM is accessed.

The RAM is addressed always in Longword (32bit per address).

3.2.6.1.17 can_cen_pflram_n

This is the RAM cell select signal for the CAN PFL RAM.

It is low active, synchronized by clk_ram and is used to enable the CAN PFL RAM when CPU or the CAN IP is accessing to the CAN PFL RAM.

3.2.6.1.18 can_data_pflram[31:0]

This is the write data bus to the CAN PFL RAM.

It is synchronized by clk_ram and set by the CAN IP when the CAN PFL RAM is written.

3.2.6.1.19 can_wen_pflram_n

This is the write enable signal for the CAN PFL RAM.

It is low active, synchronized by clk_ram and is used when CPU or the CAN IP is writing to the CAN PFL RAM.

From CAN IP the access width is 32-bit every time. From CPU theoretically 8- and 16-bit are also possible, but because of ECC usage, the RS-CAN-FD IP converts the CPU access in a 32-bit access every time.

3.2.6.2 Additional Interface information

3.2.6.2.1 ECC usage

The RS-CAN-FD IP is used together with the RAM as data buffer. To have the possibility to identify bit errors in the RAM area, ECC should be used.

Only in case the 2bit ECC error is detected during a Message Buffer access when the transmit scan is ongoing.

In this case the related Channel Transmit Request will be suppressed to avoid of inconsistent Data transmission.

In Figure 3.9 the timing diagram of the ECC module used in the M40PF is used. EC7TPREF is the ECC error signal used in this design and should be connected to mram_ecc_err.

With the signal EC7TERE the peripheral IP could mask the Error status signal to be active only during the masked time. In order to be independent from the actual ECC module implementation, the RS-CAN-FD IP does not make use of this masking signal. If such masking signal is present at the ECC module it should be connected in that way that the ECC signal is always shown.

According to the timing diagram below in this case the EC7TPREF signal is identical to the Error Status (internal) signal.

Only requirement from the CAN module to the ECC module is that the ECC error signal is valid in the first clock cycle when the read data is provided.

The APB bus of an ECC module must connect the APB bus of the same frequency as a RAM clock.

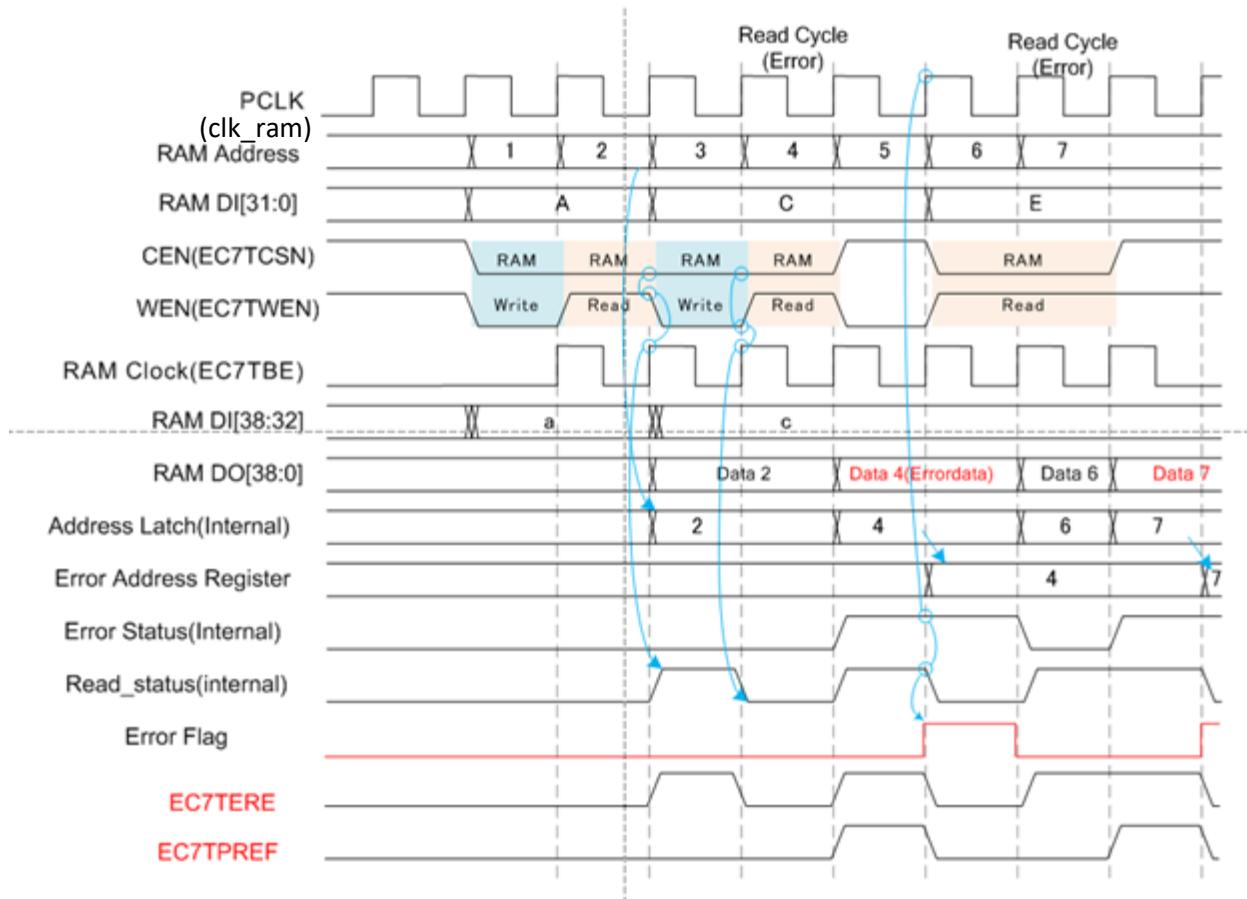


Figure 3.9 ECC module timing diagram

3.2.6.2.2 RAM access timing

CPU access duration via APB:

a CPU 32-bit read or write (8-, 16-, and 32-bit) access to SFR register in the RS-CAN-FD IP takes 2 APB bus cycles;

a CPU 32-bit read access from the RAM area takes 3 APB bus cycles;

a CPU 32-bit write access to the RAM area takes 2 APB bus cycles;

a CPU 8- or 16-bit write access to the RAM area takes 2 APB bus cycles;

Note: The timing diagrams in case of 'CPU write' to 'read only' area could be different to the shown diagrams.

As these diagrams show only valid access modes. This kind off invalid access should be avoided or not considered from timing point of view. A write will never take longer than 2 APB bus cycle.

The timing diagrams for the different access types (32-bit read, 8- or 16-bit write and 32-bit write) are shown as follows:

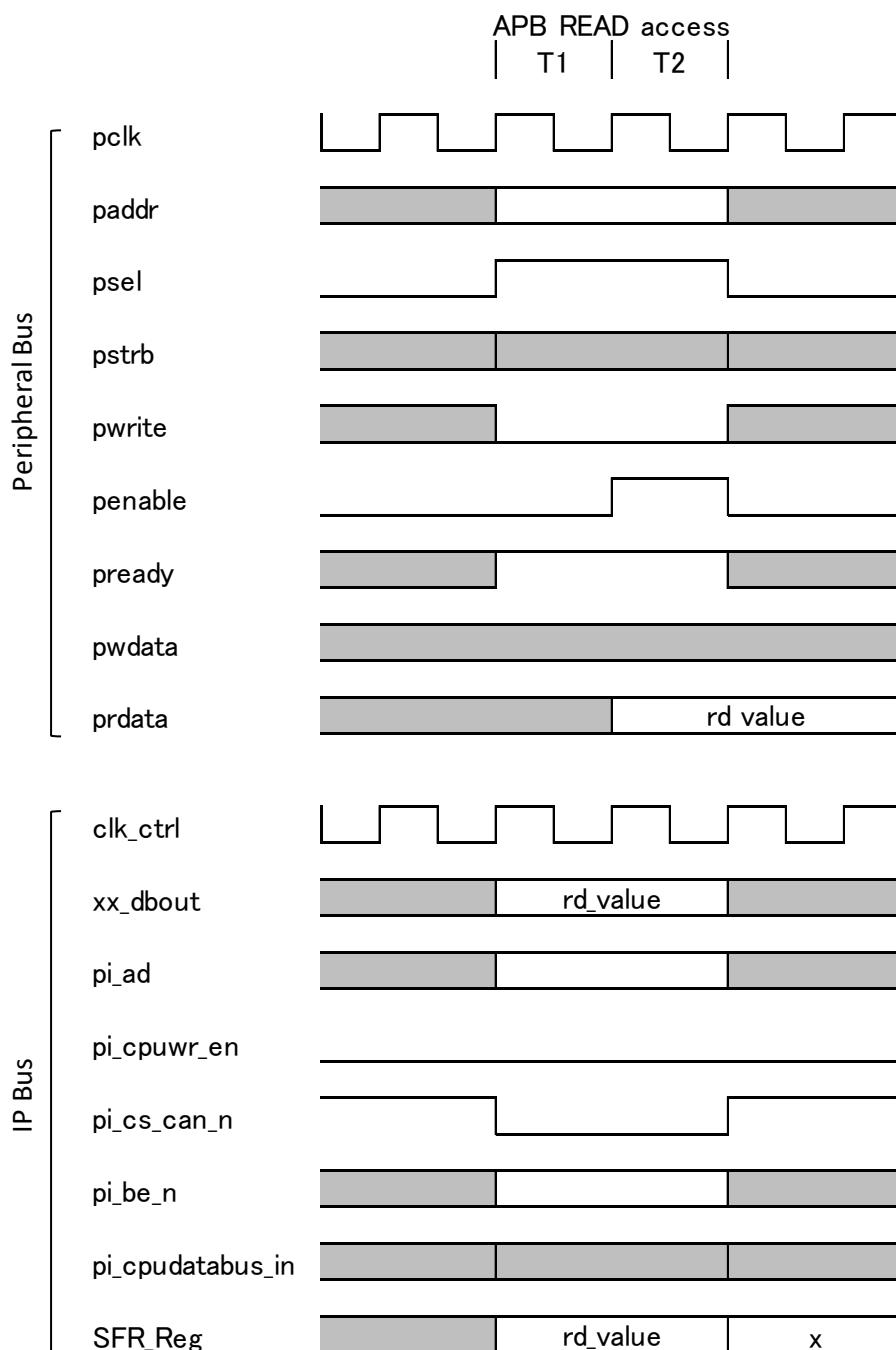


Figure 3.10 SFR read by CPU access

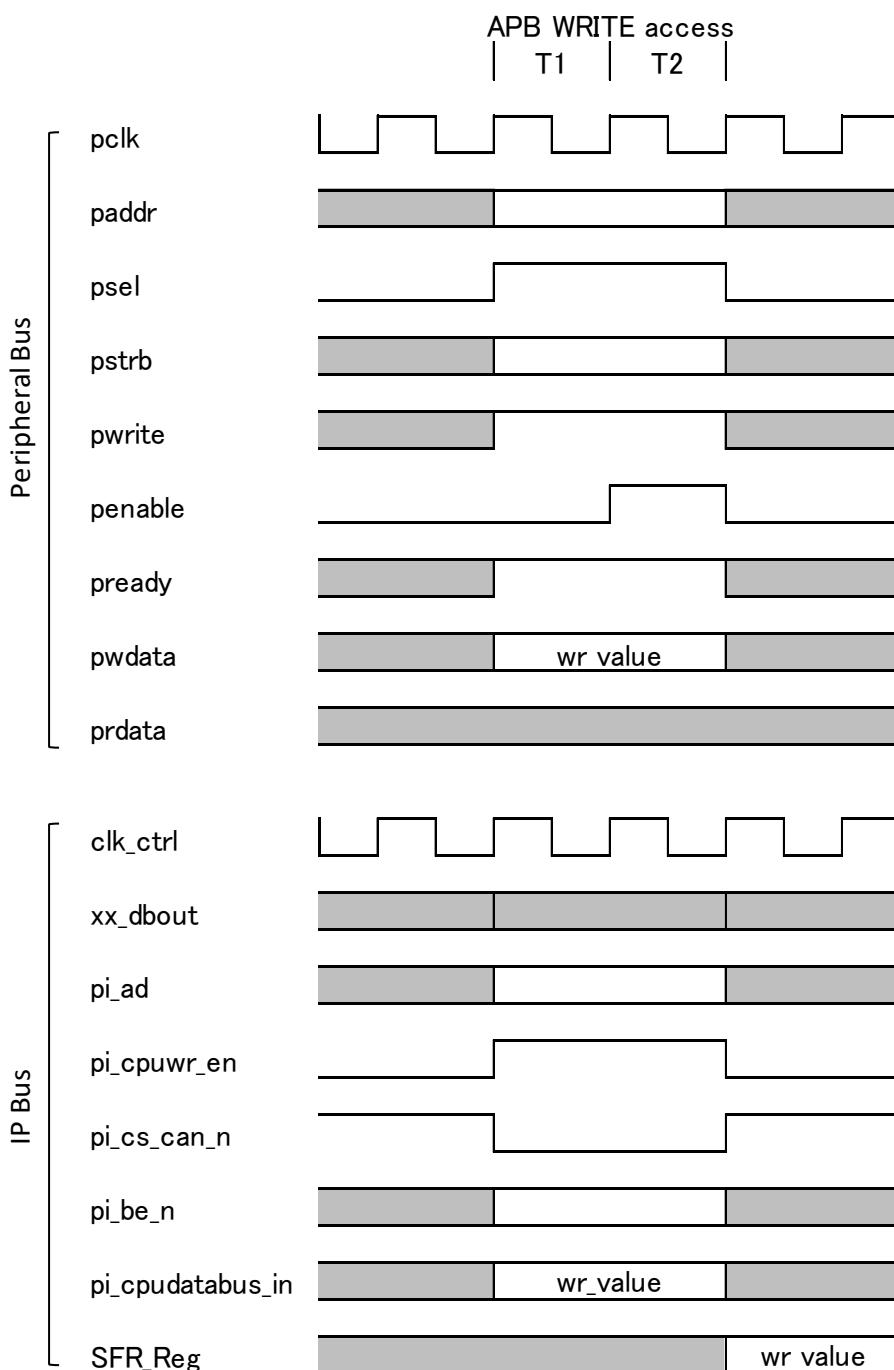
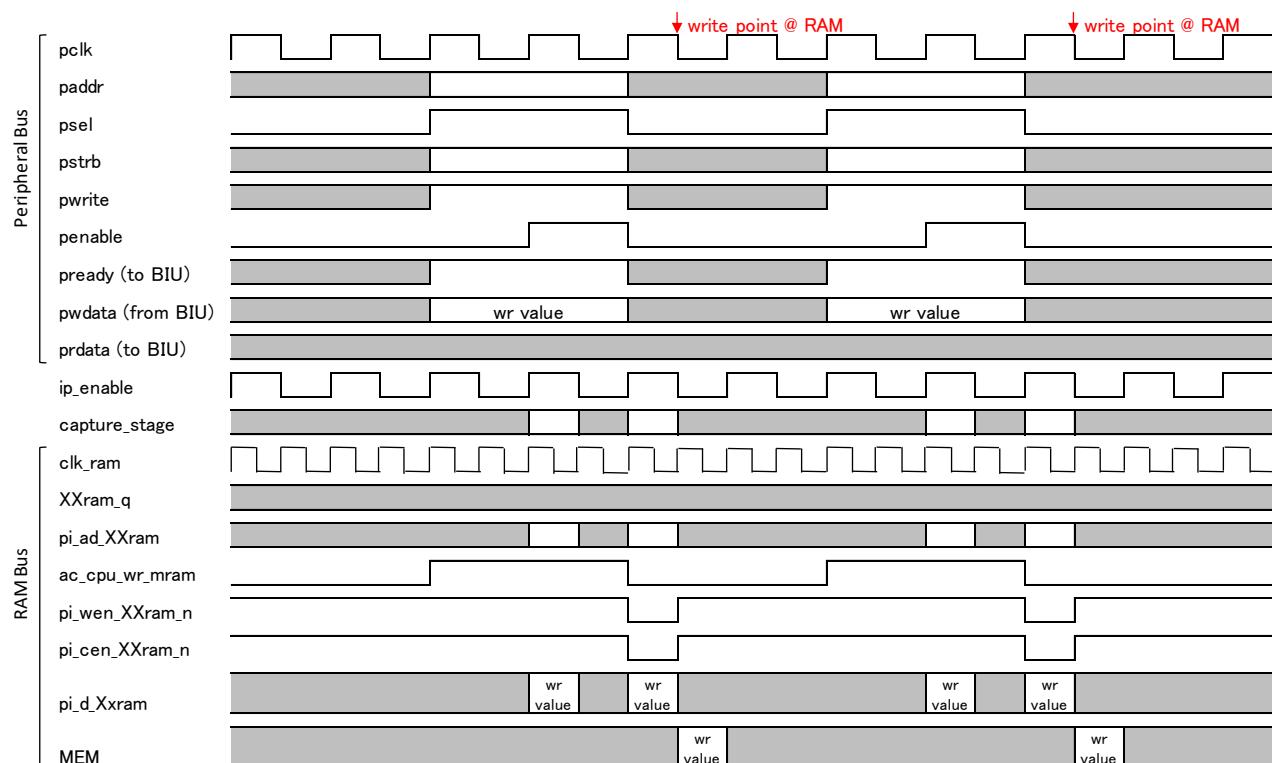
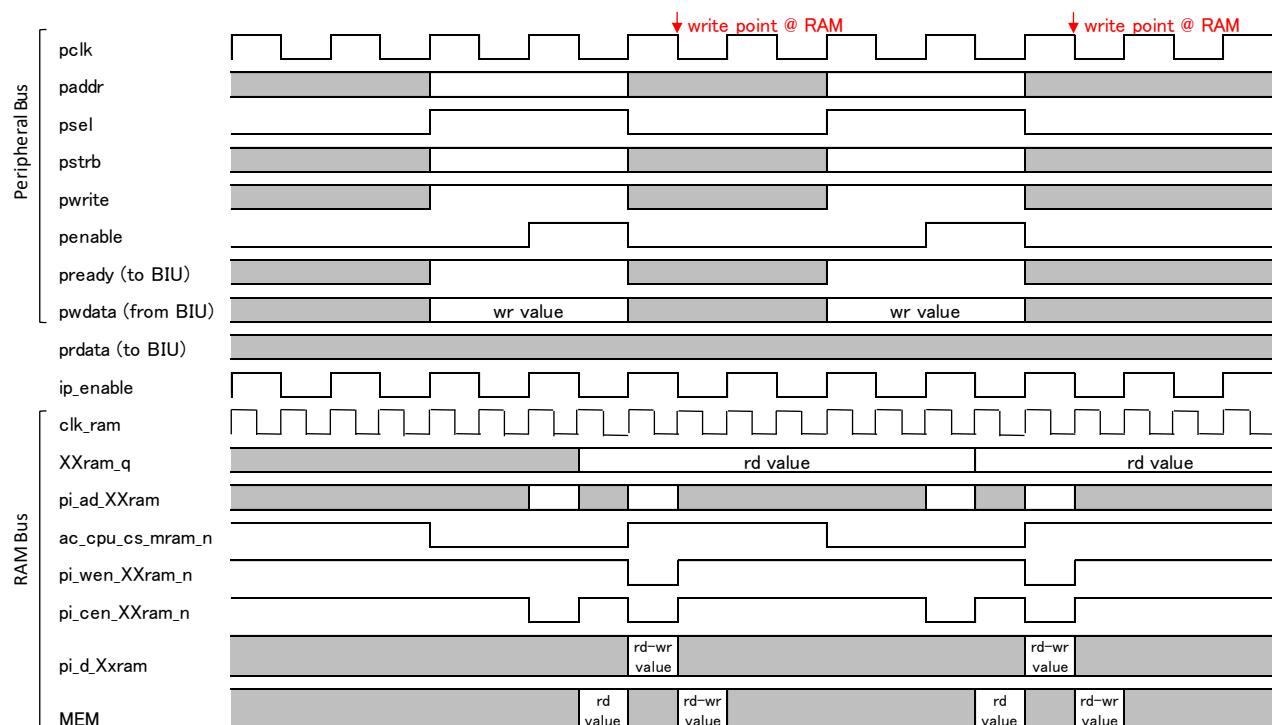


Figure 3.11 SFR write by CPU access


Figure 3.12 CPU 32-bit write to RAM

Figure 3.13 CPU 16-bit write to RAM

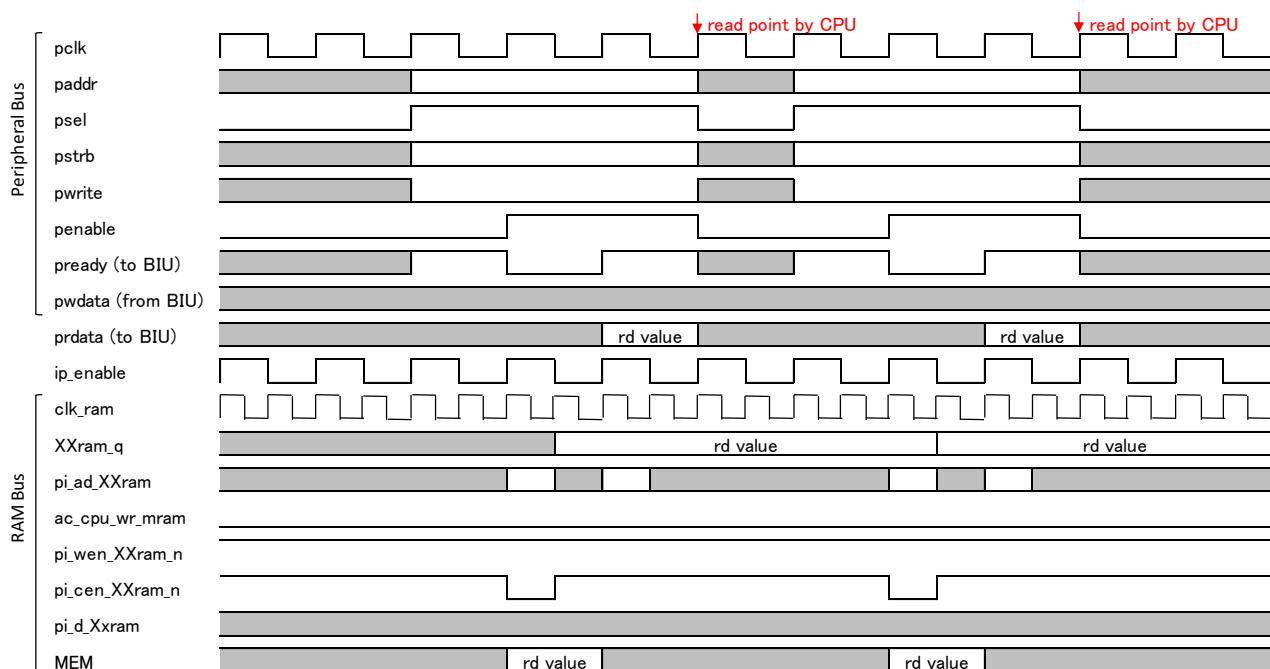


Figure 3.14 CPU read from RAM

3.2.6.2.3 AFLRAM split

The AFLRAM is divided in two parts to accelerate the AFL access process:

AFLRAM0 stores information of **CFDGAFLID** and **CFDGAFLP0**.

AFLRAM1 stores information of **CFDGAFLM** and **CFDGAFLP1**.

3.2.7 R-ACE Interface

The R-ACE bus interface signals conform to AMB APB2.0 and also conform to the APB-I/F Rule for M40 Platform Ver.4.0.

In addition, it is an addition about output signal and input signal required of R-ACE.

Signal name	IO	Function	Internal connection	Interface	Clock domain	Initial value	Active level
r_psel	in	APB select signal 0: Not selected 1: Selected	r_psel	APB control	pclk	-	high
r_paddr[16:0]	in	APB address bus	r_paddr[16:0]	APB control	pclk	-	-
r_penable	in	APB enable signal 0: cycle 0 1: second and subsequent cycle	r_penable	APB control	pclk	-	high
r_pwrite	in	APB write signal 0: read 1: write	r_pwrite	APB control	pclk	-	high
r_pstrb[3:0]	in	APB Byte enable signal	r_pstrb[3:0]	APB control	pclk	-	-
r_pwdata[31:0]	in	APB Input Data bus	r_pwdata[31:0]	APB control	pclk	-	-
r_prdata[31:0]	out	APB Output Data bus	r_prdata[31:0]	APB control	pclk	All low	-
r_pready	out	APB ready signal Bus access extend signal 0: access on-going 1: access completed	r_pready	APB control	pclk	high	high
ac_comfifo_msg_count[((n+1)*3*8)-1:0]	out	Direct access to the COMFIFO "msg_count";	ac_comfifo_m sg_count[((n+1)*3*8)-1:0]		pclk	-	-
ac_comfifo_d epth[((n+1)*3 *8)-1:0]	out	Direct access to the COMFIFO "depth";	ac_comfifo_de pth[((n+1)*3*8)-1:0]		pclk	-	-
can_ch_free[n:0]	out	Direct access to the CAN status;	can_ch_free[n :0]		pclk	-	-
can_rx FIFO_e mpty[7:0]	out	Direct access to the RXFIFO status "empty";	can_rx FIFO_em pty[7:0]		pclk	-	-

<i>can_thl_empty[n:0]</i>	<i>out</i>	<i>Direct access to the THL FIFO, to see there is a new entry;</i>	<i>can_thl_empty[n:0]</i>		<i>pclk</i>	-	-
<i>can_race_ts[((n+1)*16-1:0)]</i>	<i>in</i>	<i>Address in which Timestamp of TSCapture is stored is inputted;</i>	<i>can_race_ts[((n+1)*16-1:0)]</i>		<i>pclk</i>	-	-
<i>can_race_ts_en[n:0]</i>	<i>in</i>	<i>The enabling signal of External Timestamp of TSCapture;</i>	<i>can_race_ts_en[n:0]</i>		<i>pclk</i>	-	<i>high</i>
<i>can_rx_ts_capture[n:0]</i>	<i>out</i>	<i>Timestamp capture signal of the received CAN frame;</i>	<i>can_rx_ts_capture[n:0]</i>		<i>pclk</i>	<i>low</i>	<i>high</i>
<i>can_tx_ts_capture[n:0]</i>	<i>out</i>	<i>Timestamp capture signal of the transmitted CAN frame</i>	<i>can_tx_ts_capture[n:0]</i>		<i>pclk</i>	<i>low</i>	<i>high</i>
<i>ac_txqueue0_msg_count[((n+1)*6)-1:0]</i>	<i>out</i>	<i>Direct access to the TX Queue0 “msg_count”;</i>	<i>ac_txqueue0_msg_count[((n+1)*6)-1:0]</i>		<i>pclk</i>	-	-
<i>ac_txqueue0_depth[((n+1)*5)-1:0]</i>	<i>out</i>	<i>Direct access to the TX Queue0 “depth”;</i>	<i>ac_txqueue0_depth[((n+1)*5)-1:0]</i>		<i>pclk</i>	-	-
<i>ac_txqueue1_msg_count[((n+1)*6)-1:0]</i>	<i>out</i>	<i>Direct access to the TX Queue1 “msg_count”;</i>	<i>ac_txqueue1_msg_count[((n+1)*6)-1:0]</i>		<i>pclk</i>	-	-
<i>ac_txqueue1_depth[((n+1)*5)-1:0]</i>	<i>out</i>	<i>Direct access to the TX Queue1 “depth”;</i>	<i>ac_txqueue1_depth[((n+1)*5)-1:0]</i>		<i>pclk</i>	-	-
<i>ac_txqueue2_msg_count[((n+1)*6)-1:0]</i>	<i>out</i>	<i>Direct access to the TX Queue2 “msg_count”;</i>	<i>ac_txqueue2_msg_count[((n+1)*6)-1:0]</i>		<i>pclk</i>	-	-
<i>ac_txqueue2_depth[((n+1)*5)-1:0]</i>	<i>out</i>	<i>Direct access to the TX Queue2 “depth”;</i>	<i>ac_txqueue2_depth[((n+1)*5)-1:0]</i>		<i>pclk</i>	-	-
<i>ac_txqueue3_msg_count[((n+1)*6)-1:0]</i>	<i>out</i>	<i>Direct access to the TX Queue3 “msg_count”;</i>	<i>ac_txqueue3_msg_count[((n+1)*6)-1:0]</i>		<i>pclk</i>	-	-
<i>ac_txqueue3_depth[((n+1)*5)-1:0]</i>	<i>out</i>	<i>Direct access to the TX Queue3 “depth”;</i>	<i>ac_txqueue3_depth[((n+1)*5)-1:0]</i>		<i>pclk</i>	-	-

Table 3.14 Interface signals for the R-ACE interface

Note: Initial value ‘-’ means “don’t care”;

3.2.7.1 Signal explanation

3.2.7.1.1 r_psel

APB select signal

3.2.7.1.2 r_paddr[16:0]

APB address bus

3.2.7.1.3 r_penable

APB enable signal

The signal indicates the second and subsequent cycles of both read and write access.

3.2.7.1.4 r_pwrite

APB write signal

The signal indicates a read access by 1'b0 and a write access by 1'b1.

3.2.7.1.5 r_pstrb[3:0]

APB Byte enable signal

3.2.7.1.6 r_pwdata[31:0]

APB write data bus

The value on pwdata will be written to the RAM or internal register.

3.2.7.1.7 r_prdata[31:0]

APB read data bus

When the APB bus requests data from the IP, the output data occurs on prdata.

3.2.7.1.8 r_pready

APB ready signal

The RS-CAN-FD IP will use this signal to extend an APB access.

This will be used for APB read access from the RAM.

3.2.7.1.9 ac_comfifo_msg_count[((n+1)*3*8)-1:0]

This output is a message count number of comfifo.

3.2.7.1.10 ac_comfifo_depth[((n+1)*3*8)-1:0]

This output is a depth number of comfifo.

3.2.7.1.11 can_ch_free[n:0]

This output shows the status of channel.

3.2.7.1.12 can_rx fifo_empty[7:0]

This output shows the empty status of RXFIFO.

3.2.7.1.13 can_thl_empty[n:0]

This output shows the empty status of THL FIFO.

3.2.7.1.14 can_race_ts [(n+1)*16-1:0]

Address in which Timestamp is stored is inputted.

3.2.7.1.15 can_race_ts_en[n:0]

The enabling signal of External Timestamp

3.2.7.1.16 can_rx_ts_capture[n:0]

Receiving side timestamp capture signal

3.2.7.1.17 can_tx_ts_capture[n:0]

Transmitting side timestamp capture signal

3.2.7.1.18 ac_txqueue0_msg_count[((n+1)*6)-1:0]

This output is a message count number of TX Queue0.

3.2.7.1.19 ac_txqueue0_depth[((n+1)*5)-1:0]

This output is a depth number of TX Queue0. This signal outputs a TXQCC0.TXQDC bit.

3.2.7.1.20 ac_txqueue1_msg_count[((n+1)*6)-1:0]

This output is a message count number of TX Queue1.

3.2.7.1.21 ac_txqueue1_depth[((n+1)*5)-1:0]

This output is a depth number of TX Queue1. This signal outputs a TXQCC1.TXQDC bit.

3.2.7.1.22 ac_txqueue2_msg_count[((n+1)*6)-1:0]

This output is a message count number of TX Queue2.

3.2.7.1.23 ac_txqueue2_depth[((n+1)*5)-1:0]

This output is a depth number of TX Queue2. This signal outputs a TXQCC2.TXQDC bit.

3.2.7.1.24 ac_txqueue3_msg_count[((n+1)*6)-1:0]

This output is a message count number of TX Queue3.

3.2.7.1.25 ac_txqueue3_depth[((n+1)*5)-1:0]

This output is a depth number of TX Queue3. This signal outputs a TXQCC3.TXQDC bit.

4 Register Overview

4.1 Register Table

The 'Value after Reset' shown in the table for the RAM area, consisting of **CFDGAFIDr**, **CFDGAFLMr**, **CFDGFLP0r**, **CFDGFLP1r**, **CFDRMBCPb**, **CFDRFMBCPb**, **CFDCFMBCPb**, **CFDTMBCPb**, **CFDGPFLIDr**, **CFDGPFLMr**, **CFDGPFLP0r**, **CFDGPFLP1r**, **CFDGPFLPTr**, **CFDGPFLPD0r**, **CFDGPFLPD1r**, **CFDGPFLPM0r**, **CFDGPFLPM1r**, **CFDTHLACC0n**, **CFDTHLACC1n** and **CFDRPGACCK** is True, after, initialisation after HW Reset. Refer to Section 6.2 for details of the initialisation process.

Register address is not modified although channel, TXMB and AFL entry change with a parameter. There are the register and bit which are deleted by the parameter. Please refer to Section 5.6 and Section 5.7. Moreover, there are the register and bit in which a setting range is modified by the parameter. Please refer to Section 5.8.

Also for the RAM area, if a write access with a size of 8 or 16 bits is done then the RS-CAN-FD module does a read modify write access to the RAM location, because the RAM requires a 32bit access via the ECC module.

In case of single bit error the correct data will be written back.

In case of multiple bit errors unknown data will be written back.

Note 1: The RAM area is initialised after HW reset, refer Section 6.2

Users should not access the space where the register is not assigned.

The read data from the space where the register is not assigned is unknown.

4.2 Legend

For all repetitive registers and bits, a lowercase index is used to indicate which slice is being referenced. If an index is being used, it is defined and described below the Register Table it is being used in.

There is one global index used across all the registers and bits that need it.

N Channel Index

Register Name	Symbol	Value After Reset	Address N/A = Not Applicable(after)	Access Size
Channel n Nominal Bitrate Configuration Register	CFDCnNCFG	00000000h	0000h + n*0010h	8, 16, 32
Channel n Control Registers	CFDCnCTR	00000005h	0004h + n*0010h	8, 16, 32
Channel n Status Registers	CFDCnSTS	00000005h	0008h + n*0010h	8, 16, 32
Channel n Error Flag Registers	CFDCnERFL	00000000h	000Ch + n*0010h	8, 16, 32
Global IP Version Register	CFDGIPV	00000143h	0080h	8, 16, 32
Global Configuration Register	CFDGCFG	00000000h	0084h	8, 16, 32
Global Control Register	CFDGCTR	00000005h	0088h	8, 16, 32
Global Status Register	CFDGSTS	0000000Dh	008Ch	8, 16, 32
Global Error Flag Register	CFDGERFL	00000000h	0090h	8, 16, 32
Global Timestamp Counter Register	CFDGTSC	00000000h	0094h	16, 32
Global Acceptance Filter List Entry Control Register	CFDGAFLECTR	00000000h	0098h	8, 16, 32
Global Acceptance Filter List Configuration Register w	CFDGAFLCFGw	00000000h	009Ch + w*0004h	8, 16, 32
RX Message Buffer Number Register	CFDRMNB	00000000h	00ACh	8, 16, 32

RX Message Buffer New Data Register t	CFDRMNDt	00000000h	00B0h + t*0004h	8, 16, 32
RX FIFO Configuration / Control Registers a	CFDRFCCa	00000000h	00C0h + a*0004h	8, 16, 32
RX FIFO Status Registers a	CFDRFSTSa	00000001h	00E0h + a*0004h	8, 16, 32
RX FIFO Pointer Control Registers a	CFDRFPCTRa	00000000h	0100h + a*0004h	8, 16, 32
Common FIFO Configuration / Control Registers d	CFDCFCCd	00000000h	0120h + d*0004h	8, 16, 32
Common FIFO Configuration / Control Enhancement Registers d	CFDCFCCEd	00000000h	0180h + d*0004h	8, 16, 32
Common FIFO Status Registers d	CFDCFSTSd	00000001h	01E0h + d*0004h	8, 16, 32
Common FIFO Pointer Control Registers d	CFDCFPCTRd	00000000h	0240h + d*0004h	8, 16, 32
FIFO Empty Status Register	CFDFESTS	FFFFFFFFh	02A0h	8, 16, 32
FIFO Full Status Register	CFDFFSTS	00000000h	02A4h	8, 16, 32
FIFO Message Lost Status Register	CFDFMSTS	00000000h	02A8h	8, 16, 32
RX FIFO Interrupt Flag Status Register	CFDRFISTS	00000000h	02Ach	8, 16, 32
Common FIFO RX Interrupt Flag Status Register	CFDCFRISTS	00000000h	02B0h	8, 16, 32
Common FIFO TX Interrupt Flag Status Register	CFDCFTISTS	00000000h	02B4h	8, 16, 32
Common FIFO One Frame RX Interrupt Flag Status Register	CFDCFOFRISTS	00000000h	02B8h	8, 16, 32
Common FIFO One Frame TX Interrupt Flag Status Register	CFDCFOFTISTS	00000000h	02BCh	8, 16, 32
Common FIFO Message Over Write Status Register	CFDCFMOGSTS	00000000h	02C0h	8, 16, 32
FIFO FDC Full Status Register	CFDFFFSTS	00000000h	02C4h	8, 16, 32
TX Message Buffer Control Registers i	CFDTMCI	00h	02D0h + i*0001h	8
TX Message Buffer Status Registers j	CFDTMSTSj	00h	07D0h + j*0001h	8
TX Message Buffer Transmission Request Status Register f	CFDTMTRSTSf	00000000h	0CD0h + f*0004h	8, 16, 32
TX Message Buffer Transmission Abort Request Status Register f	CFDTMTARSTSf	00000000h	0D70h + f*0004h	8, 16, 32
TX Message Buffer Transmission Completion Status Register f	CFDTMTCSTSf	00000000h	0E10h + f*0004h	8, 16, 32
TX Message Buffer Transmission Abort Status Register f	CFDTMTASTSf	00000000h	0EB0h + f*0004h	8, 16, 32
TX Message Buffer Interrupt Enable Configuration Register f	CFDTMIECf	00000000h	0F50h + f*0004h	8, 16, 32
TX Queue Configuration / Control Registers 0 [n]	CFDTXQCC0[n]	00000000h	1000h + n*0004h	8, 16, 32
TX Queue Status Registers 0 [n]	CFDTXQSTS0[n]	00000001h	1020h + n*0004h	8, 16, 32
TX Queue Pointer Control Registers 0 [n]	CFDTXQPCTR0[n]	00000000h	1040h + n*0004h	8, 16, 32
TX Queue Configuration / Control Registers 1 [n]	CFDTXQCC1[n]	00000000h	1060h + n*0004h	8, 16, 32
TX Queue Status Registers 1 [n]	CFDTXQSTS1[n]	00000001h	1080h + n*0004h	8, 16, 32
TX Queue Pointer Control Registers 1 [n]	CFDTXQPCTR1[n]	00000000h	10A0h + n*0004h	8, 16, 32
TX Queue Configuration / Control Registers 2 [n]	CFDTXQCC2[n]	00000000h	10C0h + n*0004h	8, 16, 32
TX Queue Status Registers 2 [n]	CFDTXQSTS2[n]	00000001h	10E0h + n*0004h	8, 16, 32
TX Queue Pointer Control Registers 2 [n]	CFDTXQPCTR2[n]	00000000h	1100h + n*0004h	8, 16, 32
TX Queue Configuration / Control Registers 3 [n]	CFDTXQCC3[n]	00000000h	1120h + n*0004h	8, 16, 32
TX Queue Status Registers 3 [n]	CFDTXQSTS3[n]	00000001h	1140h + n*0004h	8, 16, 32
TX Queue Pointer Control Registers 3 [n]	CFDTXQPCTR3[n]	00000000h	1160h + n*0004h	8, 16, 32

TX Queue Empty Status Register	CFDTXQUESTS	FFFFFFFh	1180h	8, 16, 32
TX Queue Full Interrupt Status Register	CFDTXQFISTS	00000000h	1184h	8, 16, 32
TX Queue Message Lost Status Register	CFDTXQMSTS	00000000h	1188h	8, 16, 32
TX Queue Message Overwrite Status Register	CFDTXQOWSTS	00000000h	118Ch	8, 16, 32
TX Queue Interrupt Status Register	CFDTXQISTS	00000000h	1190h	8, 16, 32
TX Queue One Frame TX Interrupt Status Register	CFDTXQOFTISTS	00000000h	1194h	8, 16, 32
TX Queue One Frame RX Interrupt Status Register	CFDTXQOFRISTS	00000000h	1198h	8, 16, 32
TX Queue Full Status Register	CFDTXQFSTS	00000000h	119Ch	8, 16, 32
TX History List Configuration / Control Register n	CFDTHLCCn	00000000h	1200h + n*0004h	8, 16, 32
TX History List Status Register n	CFDTHLSTS _n	00000001h	1220h + n*0004h	8, 16, 32
TX History List Pointer Control Registers n	CFDTHLPCTR _n	00000000h	1240h + n*0004h	8, 16, 32
Global TX Interrupt Status Register v	CFDGTINTSTS _v	00000000h	1300h + v*4	8, 16, 32
Global Test Configuration Register	CFDGTSTCFG	00000000h	1308h	8, 16, 32
Global Test Control Register	CFDGTSTCTR	00000000h	130Ch	8, 16, 32
Global FD Configuration register	CFDGFD CFG	00000000h	1314h	8, 16, 32
Global FD CRC Configuration register	CFDGCRCCFG	00000000h	1318h	8, 16, 32
Global Lock Key Register	CFDGLOCKK	00000000h	131Ch	16, 32
<i>Global OTB FIFO Configuration / Status Register</i>	<i>CFDGLOTB</i>	<i>00000101h</i>	<i>1320h</i>	<i>8, 16, 32</i>
Global AFL Ignore Entry Register	CFDGAFLIGNENT	00000000h	1324h	8, 16, 32
Global AFL Ignore Control Register	CFDGAFLIGNCTR	00000000h	1328h	16, 32
DMA Transfer Control Register	CFDCDTCT	00000000h	1330h	8, 16, 32
DMA Transfer Status Register	CFDCDTSTS	00000000h	1334h	8, 16, 32
DMA TX Transfer Control Register	CFDCDTTCT	00000000h	1340h	8, 16, 32
DMA TX Transfer Status Register	CFDCDTTSTS	00000000h	1344h	8, 16, 32
Global RX Interrupt Status Register n	CFDGRINTSTS _n	00000000h	1350h + n*0004h	8, 16, 32
Pretended Network Filter List Entry control Register	CFDGPFLECTR	00000000h	1370h	8, 16, 32
Pretended Network Filter List Entry Configuration Register u	CFDGPFLCFG _u	00000000h	1374h + u * 0004h	8, 16, 32
Global SW reset Register	CFDGRSTC	00000000h	1380h	16, 32
Global Flexible CAN mode Configuration Register	CFDGFCMC	00000000h	1384h	8, 16, 32
<i>Global Bus Interface Select Configuration Register</i>	<i>CFDGBISC</i>	<i>00000000h</i>	<i>1388h</i>	<i>16, 32</i>
Global Flexible transmission buffer assignment Configuration Register	CFDGFTBAC	00000000h	138Ch	8, 16, 32
Global Virtual Machine Mode configuration Register	CFDGFFIMC	00000000h	1390h	16, 32
Global Virtual Machine Error Interrupt Select Register	CFDGVM EIS	00000000h	1394h	8, 16, 32
Global Virtual Machine Common FIFO TXQ configuration Register n	CFDVCMCFG _n	00000000h	13A0h + n*0004h	8, 16, 32
Global Virtual Machine RX FIFO configuration Register	CFDVVMRFCFG	00000000h	13C0h	8, 16, 32
Virtual Machine Interrupt Status Register n	CFDVMISTS _n	00000000h	13E0h + n*0004h	8, 16, 32
Channel n Data Bitrate Configuration Register	CFDCnDCFG	00000000h	1400h + n*0020h	8, 16, 32

Channel n CAN-FD Configuration Register	CFDCnFDCFG	00000000h	1404h + n*0020h	8, 16, 32
Channel n CAN-FD Control Register	CFDCnFDCTR	00000000h	1408h + n*0020h	8, 16, 32
Channel n CAN-FD Status Register	CFDCnFDSTS	00000000h	140Ch + n*0020h	8, 16, 32
Channel n CAN-FD CRC Register	CFDCnFDCRC	00000000h	1410h + n*0020h	8, 16, 32
Channel n Bus load Control Register	CFDCnBLCT	00000000h	1418h + n*0020h	8, 16, 32
Channel n Bus load Status Register	CFDCnBLSTS	00000000h	141Ch + n*0020h	8, 16, 32
Global Acceptance Filter List ID Registers r = [1…10]h	CFDGAFLIDr	00000000h ^{Note1}	1800h + (r-1)*0010h	8, 16, 32
Global Acceptance Filter List Mask Registers r = [1…10]h	CFDGAFLMr	00000000h ^{Note1}	1804h + (r-1)*0010h	8, 16, 32
Global Acceptance Filter List Pointer 0 Registers r = [1…10]h	CFDGAFLP0r	00000000h ^{Note1}	1808h + (r-1)*0010h	8, 16, 32
Global Acceptance Filter List Pointer 1 Registers r = [1…10]h	CFDGAFLP1r	00000000h ^{Note1}	180Ch + (r-1)*0010h	8, 16, 32
Global Pretended Network Filter List ID Registers s = [1…4]	CFDGPFLIDs	00000000h ^{Note1}	1A00h + (s-1) * 0040h	8, 16, 32
Global Pretended Network Filter List Mask Registers s = [1…4]	CFDGPFLMs	00000000h ^{Note1}	1A04h + (s-1) * 0040h	8, 16, 32
Global Pretended Network Filter List Pointer 0 Registers s = [1…4]	CFDGPFLP0s	00000000h ^{Note1}	1A08h + (s-1) * 0040h	8, 16, 32
Global Pretended Network Filter List Pointer 1 Registers s = [1…4]	CFDGPFLP1s	00000000h ^{Note1}	1A0Ch + (s-1) * 0040h	8, 16, 32
Global Pretended Network Filter List Filter Payload Type Registers s = [1…4]	CFDGPFLPTs	00000000h ^{Note1}	1A10h + (s-1) * 0040h	8, 16, 32
Global Pretended Network Filter List Payload Data 0 Registers s = [1…4]	CFDGPFLPD0s	00000000h ^{Note1}	1A14h + (s-1) * 0040h	8, 16, 32
Global Pretended Network Filter List Payload Mask 0 Registers s = [1…4]	CFDGPFLPM0s	00000000h ^{Note1}	1A18h + (s-1) * 0040h	8, 16, 32
Global Pretended Network Filter List Payload Data 1 Registers s = [1…4]	CFDGPFLPD1s	00000000h ^{Note1}	1A1Ch + (s-1) * 0040h	8, 16, 32
Global Pretended Network Filter List Payload Mask 1 Registers s = [1…4]	CFDGPFLPM1s	00000000h ^{Note1}	1A20h + (s-1) * 0040h	8, 16, 32
Channel n TX History List Access Registers 0	CFDTHLACC0[n]	00000000h ^{Note1}	8000h + n*0008h	8, 16, 32
Channel n TX History List Access Registers 1	CFDTHLACC1[n]	00000000h ^{Note1}	8004h + n*0008h	8, 16, 32
RAM Test Page Access Registers k	CFDRPGACCK	00000000h ^{Note1}	8400h + k*0004h	8, 16, 32
RX Message Buffer ID Registers	CFDRMID	00000000h ^{Note1}	Refer to Figure 4.2	8, 16, 32
RX Message Buffer Pointer Registers	CFDRMPTR	00000000h ^{Note1}	Refer to Figure 4.2	8, 16, 32
RX Message Buffer CAN-FD Status Register	CFDRMFSTS	00000000h ^{Note1}	Refer to Figure 4.2	8, 16, 32
RX Message Buffer Data Field p Registers	CFDRMDFp	00000000h ^{Note1}	Refer to Figure 4.2	8, 16, 32
RX FIFO Access ID Registers	CFDRFID	00000000h ^{Note1}	Refer to Figure 4.2	8, 16, 32
RX FIFO Access Pointer Register	CFDRFPTR	00000000h ^{Note1}	Refer to Figure 4.2	8, 16, 32
RX FIFO Access CAN-FD Status Register	CFDRFFDSTS	00000000h ^{Note1}	Refer to Figure 4.2	8, 16, 32
RX FIFO Access Data Field p Registers	CFDRFDFp	00000000h ^{Note1}	Refer to Figure 4.2	8, 16, 32
Common FIFO Access ID Registers	CFDCFID	00000000h ^{Note1}	Refer to Figure 4.2	8, 16, 32
Common FIFO Access Pointer Registers	CFDCFPT	00000000h	Refer to Figure 4.2	8, 16, 32

Common FIFO Access CAN-FD Control/Status Register	CFDCFFDCSTS	00000000h ^{Note1}	Refer to Figure 4.2	8, 16, 32
Common FIFO Access Data Field p Registers	CFDCFDFp	00000000h ^{Note1}	Refer to Figure 4.2	8, 16, 32
TX Message Buffer ID Registers	CFDTMID	00000000h ^{Note1}	Refer to Figure 4.2	8, 16, 32
TX Message Buffer Pointer Registers	CFDTMPTR	00000000h ^{Note1}	Refer to Figure 4.2	8, 16, 32
TX Message Buffer CAN-FD Control Register	CFDTMFDCTR	00000000h ^{Note1}	Refer to Figure 4.2	8, 16, 32
TX Message Buffer Data Field p Registers	CFDTMDFp	00000000h ^{Note1}	Refer to Figure 4.2	8, 16, 32
<i>RX FIFO Access ID Registers for Emulation</i>	<i>CFDRFIDE</i>	<i>00000000h^{Note1}</i>	<i>Refer to Figure 4.2</i>	<i>8, 16, 32</i>
<i>RX FIFO Access Pointer Register for Emulation</i>	<i>CFDRFPTR</i>	<i>00000000h^{Note1}</i>	<i>Refer to Figure 4.2</i>	<i>8, 16, 32</i>
<i>RX FIFO Access CAN-FD Status Register for Emulation</i>	<i>CFDRFFDSTSE</i>	<i>00000000h^{Note1}</i>	<i>Refer to Figure 4.2</i>	<i>8, 16, 32</i>
<i>RX FIFO Access Data Field p Registers for Emulation</i>	<i>CFDRFDpE</i>	<i>00000000h^{Note1}</i>	<i>Refer to Figure 4.2</i>	<i>8, 16, 32</i>
<i>COM FIFO Access ID Registers for Emulation</i>	<i>CFDCFIDE</i>	<i>00000000h^{Note1}</i>	<i>Refer to Figure 4.2</i>	<i>8, 16, 32</i>
<i>COM FIFO Access Pointer Registers for Emulation</i>	<i>CFDCFPTRE</i>	<i>00000000h^{Note1}</i>	<i>Refer to Figure 4.2</i>	<i>8, 16, 32</i>
<i>COM FIFO Access CAN-FD Control/Status Register for Emulation</i>	<i>CFDCFFDCSTSE</i>	<i>00000000h^{Note1}</i>	<i>Refer to Figure 4.2</i>	<i>8, 16, 32</i>
<i>COM FIFO Access Data Field p Registers for Emulation</i>	<i>CFDCFDFpE</i>	<i>00000000h^{Note1}</i>	<i>Refer to Figure 4.2</i>	<i>8, 16, 32</i>

Note 1: The RAM area is initialised after HW reset, refer chapter 6.2

4.3 Register description

4.3.1 CFDCnNCFG

Channel n Nominal Bitrate Configuration Register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
	NTSEG2[6:0]							NTSEG1[7]
Value after reset	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
	NTSEG1[6:0]							NSJW[6]
Value after reset	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
	NSJW[5:0]							NBRP[9:8]
Value after reset	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
	NBRP[7:0]							
Value after reset	0	0	0	0	0	0	0	0
Bit	Symbol	Bit name	Function					R/W
b[31:25]	NTSEG2	Timing Segment 2	7'b0000000: Reserved 7'b0000001: 2Tq : 7'b1111110: 127 Tq 7'b1111111: 128 Tq					R/W
b[24:17]	NTSEG1	Timing Segment 1	8'b00000000: Reserved 8'b00000001: 2Tq 8'b00000010: 3Tq 8'b00000011: 4Tq : 8'b11111110: 255 Tq 8'b11111111: 256 Tq					R/W
b[16:10]	NSJW	Resynchronization Jump Width	7'b0000000: 1Tq 7'b0000001: 2Tq : 7'b1111110: 127 Tq 7'b1111111: 128 Tq					R/W
b[9:0]	NBRP	Channel Nominal Baud Rate Prescaler	Nominal Baud Rate Prescaler division ratio					R/W

This register is used to configure the transmission / reception nominal Baud Rate parameters of the channels.

4.3.1.1 CFDCnNCFG.NBRP

Channel Nominal Baud Rate Prescaler

These bits are used to define the peripheral bus clock periods contained in a Time Quantum.

This bit cannot be written in CH_OPERATION or CH_SLEEP mode.

Users should write to these bits, only when the RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

4.3.1.2 CFDCnNCFG.NSJW

Resynchronization Jump Width

These bits set the synchronization jump width. A value from 1 to 128 time quanta can be set.

This bit cannot be written in CH_OPERATION or CH_SLEEP mode.

Users should write to these bits, only when the RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

4.3.1.3 CFDCnNCFG.NTSEG1

Timing Segment 1

These bits are used to set the segment TSEG1 to compensate for edges on the CAN Bus with a positive phase error.

It also contains the propagation segment.

This bit cannot be written in CH_OPERATION or CH_SLEEP mode.

Users should write to these bits, only when the RS-CAN-FD channel is in CH_RESET or CH_HALT mode.
Users should configure a Tq value, only between 2 and 256 inclusive. See Section 6.1.2 for more details.

4.3.1.4 CFDCnNCFG.NTSEG2

Timing Segment 2

These bits are used to set the segment TSEG2 to compensate for edges on the CAN Bus with a negative phase error.

This bit cannot be written in CH_OPERATION or CH_SLEEP mode.

Users should write to these bits, only when the RS-CAN-FD channel is in CH_RESET or CH_HALT mode.
Users should configure a Tq value, only between 2 and 128 inclusive.

4.3.2 CFDCnCTR

Channel n Control Registers

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
	ROM	CRCT	TRR	TRH	TRWE	CTMS[1:0]	CTME	
Value after reset	0	0	0	0	0	0	0	0

	b23	b22	b21	b20	b19	b18	b17	b16
	ERRD	BOM[1:0]	-	TDCVFIE	SOCOIE	ECCOIE	TAIE	
Value after reset	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8
	ALIE	BLIE	OLIE	BORIE	BOEIE	EPIE	EWIE	BEIE
Value after reset	0	0	0	0	0	0	0	0

	b7	b6	b5	b4	b3	b2	b1	b0
	-	-	-	-	RTBO	CSLPR	CHMDC[1:0]	
Value after reset	0	0	0	0	0	1	0	1

Bit	Symbol	Bit name	Function	R/W
b[31]	ROM	Restricted Operation Mode	1'b0: Restricted Operation Mode disabled 1'b1: Restricted Operation Mode enabled	R/W
b[30]	CRCT	CRC Error Test	1'b0: First data bit of reception stream not inverted 1'b1: First data bit of reception stream inverted	R/W
b[29]	TRR	TEC/REC Reset	1'b0: Error counter normal operation 1'b1: Error counter reset	R/W
b[28]	TRH	TEC/REC Hold	1'b0: Error counter normal operation 1'b1: Error counter frozen	R/W
b[27]	TRWE	TEC/REC Write Enable	1'b0: Error Counter write disabled 1'b1: Error Counter write enabled	R/W
b[26:25]	CTMS	Channel Test Mode Select	2'b00: Basic test mode 2'b01: Listen-Only mode 2'b10: Self test mode 0 (External Loop back mode) 2'b11: Self test mode 1 (Internal Loop back mode)	R/W
b[24]	CTME	Channel Test Mode Enable	1'b0: Channel Test Mode disabled 1'b1: Channel Test Mode enabled	R/W
b[23]	ERRD	Channel Error Display	1'b0: Only the 1 st set of error codes displayed 1'b1: Accumulated error codes displayed	R/W
b[22:21]	BOM	Channel Bus-Off Mode	2'b00: Normal mode (comply with ISO 11898-1) 2'b01: Entry to Halt Mode automatically at Bus-Off start 2'b10: Entry to Halt Mode automatically at Bus-Off end 2'b11: Entry to Halt Mode (during Bus-Off Recovery Period) by S/W	R/W
b[20]	-	Reserved	This bit is read as 0b. The write value should be always 0b.	R/W
b[19]	TDCVFIE	Transceiver Delay Compensation Violation Interrupt enable	1'b0: Transceiver Delay Compensation Violation Interrupt disabled 1'b1: Transceiver Delay Compensation Violation Interrupt enabled	R/W
b[18]	SOCOIE	Successful Occurrence Counter Overflow Interrupt enable	1'b0: Successful occurrence counter overflow interrupt disabled 1'b1: Successful occurrence counter overflow interrupt enabled	R/W
b[17]	ECCOIE	Error occurrence counter overflow Interrupt enable	1'b0: Error occurrence counter overflow Interrupt disabled 1'b1: Error occurrence counter overflow Interrupt enabled	R/W
b[16]	TAIE	Transmission abort Interrupt Enable	1'b0: TX abort Interrupt disabled 1'b1: TX abort Interrupt enabled	R/W
b[15]	ALIE		1'b0: Arbitration Lost Interrupt disabled	R/W

		Arbitration Lost Interrupt Enable	1'b1: Arbitration Lost Interrupt enabled	
b[14]	BLIE	Bus Lock Interrupt Enable	1'b0: Bus Lock Interrupt disabled 1'b1: Bus Lock Interrupt enabled	R/W
b[13]	OLIE	Overload Interrupt Enable	1'b0: Overload Interrupt disabled 1'b1: Overload Interrupt enabled	R/W
b[12]	BORIE	Bus-Off Recovery Interrupt Enable	1'b0: Bus-Off Recovery Interrupt disabled 1'b1: Bus-Off Recovery Interrupt enabled	R/W
b[11]	BOEIE	Bus-Off Entry Interrupt Enable	1'b0: Bus-Off Entry Interrupt disabled 1'b1: Bus-Off Entry Interrupt enabled	R/W
b[10]	EPIE	Error Passive Interrupt Enable	1'b0: Error Passive Interrupt disabled 1'b1: Error Passive Interrupt enabled	R/W
b[9]	EWIE	Error Warning Interrupt Enable	1'b0: Error Warning Interrupt disabled 1'b1: Error Warning Interrupt enabled	R/W
b[8]	BEIE	Bus Error Interrupt Enable	1'b0: Bus Error Interrupt disabled 1'b1: Bus Error Interrupt enabled	R/W
b[7:4]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[3]	RTBO	Return from Bus-Off	1'b0: Channel is not forced to return from Bus-Off 1'b1: Channel is forced to return from Bus-Off	R/W
b[2]	CSLPR	Channel Sleep Request	1'b0: Channel Sleep Request disabled 1'b1: Channel Sleep Request enabled	R/W
b[1:0]	CHMDC	Channel Mode Control	2'b00: Channel Operation Mode request 2'b01: Channel Reset request 2'b10: Channel Halt request 2'b11: Keep current value	R/W

Each Channel control register is used to control the modes of the related channel. It is used to enable generation of interrupts if errors are detected on the CAN bus connected to this channel. It is also used to configure the channel in test mode.

4.3.2.1 CFDCnCTR.CHMDC

Channel Mode Control

CFDCnCTR.CHMDC bits can be used to configure the modes of the CAN channel. CAN mode transitions are described in more details in Section 5.3

Setting **CFDCnCTR.CHMDC** to 2'b11 by the CPU has no effect.

When the RS-CAN-FD module is in GL_HALT mode, these bits can only be set to 2'b10 or 2'b01. This bit cannot be set in CH_SLEEP mode.

CFDCnCTR.CHMDC can change automatically in case of transition to Halt mode due to **CFDCnCTR.BOM** settings.

If CPU write access to **CFDCnCTR.CHMDC** happens at the same time when the CAN channel is about to enter Halt Mode (at the start of Bus-Off when **CFDCnCTR.BOM**=2'b01, or at the end of Bus-Off when **CFDCnCTR.BOM** =2'b10), then the CPU write access will have the highest priority.

The CAN channel changes the value of **CFDCnCTR.CHMDC** within the Channel Control Registers for the above cases only if the **CFDCnCTR.CHMDC** value is 2'b00 (Operation Mode).

4.3.2.2 CFDCnCTR.CSLPR

Channel Sleep Request

A sleep mode request is generated for the corresponding CAN channel, when the **CFDCnCTR.CSLPR** bit is 1'b1.

A request to exit sleep mode is generated for the corresponding CAN channel, when the **CFDCnCTR.CSLPR** bit is 1'b0.

Users can only write to this bit when the related RS-CAN-FD channel is in CH_RESET or CH_SLEEP mode.

4.3.2.3 CFDCnCTR.RTBO

Return from Bus-Off

When the protocol controller of the CAN channel enters Bus-Off state, users can force it to recover from Bus-Off state by setting the **CFDCnCTR.RTBO** bit in the Channel Control Register to 1'b1.

The error state changes from Bus-Off state to integrating with a maximum delay of 1 CAN Bit time.

If the **CFDCnCTR.RTBO** bit is set to 1'b1, the REC and TEC registers are initialized and the Bus-Off status bit (Channel Bus-off Status, **CFDCnSTS.BOSTS**) is set to 1'b0.

The other registers are not initialized by this command. Even if **CFDCnCTR.BORIE** is set, a Bus-Off recovery interrupt is not generated by this recovery from the Bus-Off state.

This bit cannot be set in CH_SLEEP mode.

Setting this bit in any state (other than Bus-Off) will have no effect and the bit will be cleared immediately.

Read value is always 0.

Return from Bus-Off command should be used only when **CFDCnCTR.BOM** is set to 2'b00.

Users should only write to this bit when the related RS-CAN-FD channel is in CH_OPERATION mode.

This bit is automatically cleared once set by the System SW.

4.3.2.4 CFDCnCTR.BEIE

Bus Error Interrupt Enable

An error interrupt is generated, when the **CFDCnCTR.BEIE** bit and the **CFDCnERFL.BEF** are both 1'b1.

This bit cannot be set in CH_SLEEP mode.

Users should write to this bit only when the related RS-CAN-FD channel is in CH_RESET mode.

4.3.2.5 CFDCnCTR.EWIE

Error Warning Interrupt Enable

An error interrupt is generated, when the **CFDCnCTR.EWIE** bit and the **CFDCnERFL.EWF** are both 1'b1.

This bit cannot be set in CH_SLEEP mode.

Users should write to this bit only when the related RS-CAN-FD channel is in CH_RESET mode.

4.3.2.6 CFDCnCTR.EPIE

Error Passive Interrupt Enable

An error interrupt is generated, when the **CFDCnCTR.EPIE** bit and the **CFDCnERFL.EPF** are both 1'b1.

This bit cannot be set in CH_SLEEP mode.

Users should write to this bit only when the related RS-CAN-FD channel is in CH_RESET mode.

4.3.2.7 CFDCnCTR.BOEIE

Bus-Off Entry Interrupt Enable

An error interrupt is generated, when the **CFDCnCTR.BOEIE** bit and the **CFDCnERFL.BOEF** are both 1'b1.

This bit cannot be set in CH_SLEEP mode.

Users should write to this bit only when the related RS-CAN-FD channel is in CH_RESET mode.

4.3.2.8 CFDCnCTR.BORIE

Bus-Off Recovery Interrupt Enable

An error interrupt is generated, when the **CFDCnCTR.BORIE** bit and the **CFDCnERFL.BORF** are both 1'b1.

This bit cannot be set in CH_SLEEP mode.

Users should write to this bit only when the related RS-CAN-FD channel is in CH_RESET mode.

4.3.2.9 CFDCnCTR.OLIE

Overload Interrupt Enable

An error interrupt is generated, when the **CFDCnCTR.OLIE** bit and the **CFDCnERFL.OVLF** are both 1'b1.

This bit cannot be set in CH_SLEEP mode.

Users should write to this bit only when the related RS-CAN-FD channel is in CH_RESET mode.

4.3.2.10 CFDCnCTR.BLIE

Bus Lock Interrupt Enable

An error interrupt is generated, when the **CFDCnCTR.BLIE** bit and the **CFDCnERFL.BLF** are both 1'b1.

This bit cannot be set in CH_SLEEP mode.

Users should write to this bit only when the related RS-CAN-FD channel is in CH_RESET mode.

4.3.2.11 CFDCnCTR.ALIE

Arbitration Lost Interrupt Enable

An error interrupt is generated, when the **CFDCnCTR.ALIE** bit and the **CFDCnERFL.ALF** are both 1'b1.

This bit cannot be set in CH_SLEEP mode.

Users should write to this bit only when the related RS-CAN-FD channel is in CH_RESET mode.

4.3.2.12 CFDCnCTR.TAIE

Transmission abort Interrupt Enable

An interrupt is generated, when the **CFDCnCTR.TAIE** bit is 1'b1 and a transmission is successfully aborted from a TX MB belonging to the corresponding CAN channel.

This bit cannot be set in CH_SLEEP mode.

Users should write to this bit only when the related RS-CAN-FD channel is in CH_RESET mode.

4.3.2.13 CFDCnCTR.EOCOIE

Error occurrence counter overflow Interrupt enable

An error interrupt is generated, when the **CFDCnCTR.EOCOIE** bit is 1'b1 and the **CFDCnFDSTS.EOCO** bit belonging to the corresponding CAN channel is 1'b1.

This bit cannot be set in CH_SLEEP mode.

Users should write to this bit only when the related RS-CAN-FD channel is in CH_RESET mode.

4.3.2.14 CFDCnCTR.SOCOIE

Successful Occurrence Counter Overflow Interrupt enable

An error interrupt is generated, when the **CFDCnCTR.SOCOIE** bit is 1'b1 and the **CFDCnFDSTS.SOCO** bit belonging to the corresponding CAN channel is 1'b1.

This bit cannot be set in CH_SLEEP mode.

Users should write to this bit only when the related RS-CAN-FD channel is in CH_RESET mode.

4.3.2.15 CFDCnCTR.TDCVFIE

Transceiver Delay Compensation Violation Interrupt enable

An error interrupt is generated, when the **CFDCnCTR.TDCVFIE** bit is 1'b1 and the **CFDCnFDSTS.TDCVF** bit belonging to the corresponding CAN channel is 1'b1.

This bit cannot be set in CH_SLEEP mode.

Users should write to this bit only when the related RS-CAN-FD channel is in CH_RESET mode.

Users should not set to this bit when Classical only mode.

4.3.2.16 CFDCnCTR.BOM

Channel Bus-Off Mode

These bits control the timing of the recovery from Bus-Off mode of the RS-CAN-FD Channel.

This bit cannot be set in CH_SLEEP mode.

Users should write to this bit only when the related RS-CAN-FD channel is in CH_RESET mode.

4.3.2.17 CFDCnCTR.ERRD

Channel Error Display

This bit controls the display mode of the error flag bits (bits [14:8]) in the Channel Error Flag Register (**CFDCnERFL**).

If the **CFDCnCTR.ERRD** bit is 1'b0 and more than one error occurs at the same time, then the error flag bits will set for all the errors that occurred at the same time. No further errors are flagged until **CFDCnERFL[14:8]** is cleared.

This bit cannot be set in CH_SLEEP mode.

Users should write to this bit only when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

4.3.2.18 CFDCnCTR.CTME

Channel Test Mode Enable

This bit is set to enable the channel test modes.

Users can only write to this bit when the related RS-CAN-FD channel is in CH_HALT mode

This bit cannot be set in CH_SLEEP mode.

The bit is cleared automatically when the related RS-CAN-FD channel transits to CH_RESET mode.

4.3.2.19 CFDCnCTR.CTMS

Channel Test Mode Select

The **CFDCnCTR.CTMS** bits are used to select the required test mode.

These bits cannot be set in CH_SLEEP or CH_RESET mode.

Users should write to these bits only when the related RS-CAN-FD channel is in CH_HALT mode.

The field is cleared automatically when the related RS-CAN-FD channel moves to CH_RESET mode.

4.3.2.20 CFDCnCTR.TRWE

TEC/REC Write Enable

*Users can only write data into the corresponding TEC / REC register if the **CFDCnCTR.TRWE** and **CFDCnCTR.CTME** bits are set.*

*Users can only write into TEC/REC through the TEC register. After writing to the TEC register, wait until the **CFDCnCTR.TRWE** has been automatically cleared before leaving Halt Mode.*

This bit cannot be set in CH_SLEEP or CH_RESET mode.

Write '0' has no effect.

Users should write to this bit only when the related RS-CAN-FD channel is in CH_HALT mode.

This bit is cleared automatically when the internal registers (TEC / REC) are updated.

4.3.2.21 CFDCnCTR.TRH

TEC/REC Hold

The value of TEC / REC can be frozen if CFDCnCTR.TRH bit and CFDCnCTR.CTME bit are set.

This bit cannot be set in CH_SLEEP or CH_RESET mode.

Users should write to this bit only when the related RS-CAN-FD channel is in CH_HALT mode.

The bit is cleared automatically when the related RS-CAN-FD channel is in CH_RESET mode.

4.3.2.22 CFDCnCTR.TRR

TEC/REC Reset

TRR Bit (TEC/REC Reset)

TRWE	TRH	TRR	Effect on REC and TEC
1'b1	don't care	1'b0	<i>Written with the value input by the CPU</i>
1'b1	don't care	1'b1	<i>Prohibited</i>
1'b0	1'b0	1'b0	<i>Behaviour in line with CAN specification</i>
1'b0	1'b1	1'b0	<i>Hold the current value</i>
1'b0	1'b0	1'b1	<i>Cleared to zero and remain at zero</i>
1'b0	1'b1	1'b1	<i>Cleared to zero and remain at zero</i>

Table 4.1 Priority of TEC/REC Test Signals

This bit cannot be set in CH_SLEEP or CH_RESET mode.

Users should write to this bit only when the related RS-CAN-FD channel is in CH_HALT mode.

The bit is cleared automatically when the related RS-CAN-FD channel is in CH_RESET mode.

The value of TEC / REC is reset if CFDCnCTR.TRR bit and CFDCnCTR.CTME bit are both 1'b1.

4.3.2.23 CFDCnCTR.CRCT

CRC Error Test

This bit is used to check the internal CRC generator logic of the protocol controller.

It inverts the first bit (ID bit) of the CAN message data stream being received, so that the internal generated CRC result will not match the received CRC value of the frame. Users should refer to the bit stuffing rule when using this feature, as there is the possibility of receiving a stuff error (due to the inversion) rather than a CRC error.

The internal generated CRC value is always observed in the following registers:

CFDCnERFL.CRCREG (classical CAN frames)

CFDCnFDCRC.CRCREG (CAN-FD frames)

There are some limitations when using this bit:

It is not possible to use this feature with CAN nodes connected to the MCU externally, only with nodes connected to the Internal CAN Bus communication can be used.

One CAN node will send a reference message and the receiver node(s) can invert one bit of incoming bit stream

Note, the transmitter and receiver mode are sharing the same CRC generator, therefore, it is not necessary to consider the modes separately when testing this limitation.

The CRC Error Test Mode is enabled if **CFDCnCTR.CRCT** (new control signal which is inverting the first bit of the bit stream) and **CFDCnCTR.CTME** are both 1'b1.

This bit cannot be set in CH_SLEEP mode.

Users can write to this bit only when the related RS-CAN-FD channel is in CH_HALT mode.

This bit is cleared automatically when the related RS-CAN-FD channel is in CH_RESET mode.

4.3.2.24 CFDCnCTR.ROM

Restricted Operation Mode

The Restricted Operation Mode is enabled if **CFDCnCTR.ROM** and **CFDCnCTR.CTME** are both 1'b1.

This bit cannot be set in CH_SLEEP mode.

Users should write to this bit only when the related RS-CAN-FD channel is in CH_HALT mode.

This mode should only be used in Basic Test mode **CFDCnCTR.CTMS(1:0)** = 2'b00.

This bit is cleared automatically when the related RS-CAN-FD channel is in CH_RESET mode.

Users should not set to this bit when Classical only mode.

4.3.3 CFDCnSTS

Channel n Status Registers

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
TEC[7:0]								
Value after reset	0	0	0	0	0	0	0	0
REC[7:0]								
Value after reset	0	0	0	0	0	0	0	0
ESIF								
Value after reset	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
	COMSTS	RECSTS	TRMSTS	BOSTS	EPSTS	CSLPSTS	CHLTSTS	CRSTSTS
Value after reset	0	0	0	0	0	1	0	1

Bit	Symbol	Bit name	Function	R/W
b[31:24]	TEC	Transmission Error Count	This register increments or decrements the counter value according to error status of the CAN channel during Transmission.	R/W
b[23:16]	REC	Reception Error Count	This register increments or decrements the counter value according to error status of the CAN channel during Reception.	R
b[15:9]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[8]	ESIF	Error State Indication Flag	1'b0: No CAN-FD message has been received with the ESI flag was set 1'b1: At least 1 CAN-FD message was received where the ESI flag was set	R/W
b[7]	COMSTS	Channel Communication Status	1'b0: Channel is not ready for communication 1'b1: Channel is ready for communication	R
b[6]	RECSTS	Channel Receive Status	1'b0: Channel is not receiving 1'b1: Channel is receiving	R
b[5]	TRMSTS	Channel Transmit Status	1'b0: Channel is not transmitting 1'b1: Channel is transmitting	R
b[4]	BOSTS	Channel Bus-Off Status	1'b0: Channel not in Bus-Off state 1'b1: Channel in Bus-Off state	R
b[3]	EPSTS	Channel Error Passive Status	1'b0: Channel not in Error Passive state. 1'b1: Channel in Error Passive state.	R
b[2]	CSLPSTS	Channel SLEEP Status	1'b0: Channel not in Sleep Mode 1'b1: Channel in Sleep Mode	R
b[1]	CHLTSTS	Channel HALT Status	1'b0: Channel not in Halt Mode 1'b1: Channel in Halt Mode	R
b[0]	CRSTSTS	Channel RESET Status	1'b0: Channel not in Reset Mode 1'b1: Channel in Reset Mode	R

Each Channel Status Register shows the mode, Error and TX / RX status of the related channel together with its Reception and Transmission Error Count values.

4.3.3.1 CFDCnSTS.CRSTSTS

Channel RESET Status

Indicates if the related CAN channel is in Reset mode.

This bit is set automatically when the related CAN channel enters Channel Reset Mode. If the mode is changed from Reset Mode to Sleep Mode, **CFDCnSTS.CRSTSTS** remains set to 1'b1.

This bit is cleared automatically when the related CAN channel exits the Channel Reset Mode, except when changing to Sleep Mode.

4.3.3.2 CFDCnSTS.CHLTSTS

Channel HALT Status

Indicates if the related CAN channel is in Halt mode.

This bit is set automatically when the related CAN channel enters Halt Mode.

This bit is cleared automatically when the related CAN channel exits Halt Mode.

4.3.3.3 CFDCnSTS.CSLPSTS

Channel SLEEP Status

Indicates if the related CAN channel is in Sleep mode.

This bit is set automatically when the related CAN channel enters Sleep Mode.

This bit is cleared automatically when the related CAN channel exits Sleep Mode.

4.3.3.4 CFDCnSTS.EPSTS

Channel Error Passive Status

Indicates if the related CAN channel has entered the error passive state.

This bit is set automatically when the value of the CAN Transmission or Reception Counter Register exceeds the value of 8'd127.

This bit is cleared automatically when the related CAN channel exits the Error Passive state or enters Reset Mode.

4.3.3.5 CFDCnSTS.BOSTS

Channel Bus-Off Status

Indicates if the related CAN channel has entered the error Bus-Off state.

This bit is set automatically when the value of the related CAN Transmission Error Count Register exceeds 8'd255 and the related CAN channel is in the Bus-Off state (CAN Transmission Error Count Register > 8'd255).

This bit is cleared automatically when the related CAN channel exits the Bus-Off state.

4.3.3.6 CFDCnSTS.TRMSTS

Channel Transmit Status

Indicates if the related CAN channel is transmitting a message.

This bit is set automatically when the related CAN channel is operating as a transmitter node or is in the Bus-Off state.

This bit is cleared automatically when the related CAN channel is in the bus-idle state or starts operating as a receiver node.

4.3.3.7 CFDCnSTS.RECSTS

Channel Receive Status

Indicates if the related CAN channel is receiving a message.

This bit is set automatically when the related CAN channel is operating as a receiver node.

This bit is cleared automatically when the related CAN channel is in the bus-idle state or starts operating as a transmitter node.

4.3.3.8 CFDCnSTS.COMSTS

Channel Communication Status

Indicates if the related CAN channel is ready for communication.

This bit is set automatically when the related CAN channel is ready to perform communication following the detection of eleven consecutive recessive bits after leaving the Reset or Halt mode.

This bit is cleared automatically when the related CAN channel is in Reset or Halt Mode.

Note: This bit is 1'b1 during Bus-Off status.

4.3.3.9 CFDCnSTS.ESIF

Error State Indication Flag

This flag is set when the ESI bit was sampled recessive for a reception CAN message without any error. Note that in case of Loopback or Mirror Mode the self transmitted messages are considered as reception messages.

If the set from the CAN channel occurs simultaneously with the clear by the write access, then the bit is set.

The bit is cleared by writing 1'b0 to it.

This bit is cleared automatically when the related CAN channel is in Reset.

Users can only write to this bit when the related RS-CAN-FD channel is in CH_HALT or CH_OPERATION mode.

Do not use bit clear instruction for clearing this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Then the other bits remain set to 1'b1.

4.3.3.10 CFDCnSTS.REC

Reception Error Count

The value of the REC error counter is displayed.

The value in Bus-Off state is indeterminate.

These bits are cleared automatically when RS-CAN-FD module enters GL_RESET or CAN channel is in CH_RESET mode.

4.3.3.11 CFDCnSTS.TEC

Transmission Error Count

The value of TEC error counter is displayed.

Users should only write to these bits when in Test Mode and RS-CAN-FD channel is in CH_HALT mode.

If the CFDCnCTR.TRR bit was set, then TEC & REC will be cleared when CAN channel enters CH_OPERATION mode. This test feature should only be used in Halt Mode.

Users should not read data from this register while CFDCnCTR.TRWE is set.

In Test mode, when CFDCnCTR.CTME is set and CFDCnCTR.TRWE is set, the TEC and REC can be written through this register. The value can be read after the CAN channel enters CH_OPERATION mode.

These bits are cleared automatically when RS-CAN-FD module is in GL_RESET or CAN channel is in CH_RESET mode.

4.3.4 CFDCnERFL

Channel n Error Flag Registers

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
	-				CRCREG[14:8]			
Value after reset	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
					CRCREG[7:0]			
Value after reset	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
	-	ADERR	B0ERR	B1ERR	CERR	AERR	FERR	SERR
Value after reset	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
	ALF	BLF	OVLF	BORF	BOEF	EPF	EWF	BEF
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31]	-	Reserved	This bit is read as 0b. The write value should be always 0b.	R/W
b[30:16]	CRCREG	CRC Register value	These bits show the CRC value calculated for the CAN2.0 CAN Frame.	R
b[15]	-	Reserved	This bit is read as 0b. The write value should be always 0b.	R/W
b[14]	ADERR	Acknowledge Delimiter Error	1'b0: Channel Ack Del Error not detected 1'b1: Channel Ack Del Error detected	R/W
b[13]	B0ERR	Bit 0 Error	1'b0: Channel Bit 0 Error not detected 1'b1: Channel Bit 0 Error detected	R/W
b[12]	B1ERR	Bit 1 Error	1'b0: Channel Bit 1 Error not detected 1'b1: Channel Bit 1 Error detected	R/W
b[11]	CERR	CRC Error	1'b0: Channel CRC Error not detected 1'b1: Channel CRC Error detected	R/W
b[10]	AERR	Acknowledge Error	1'b0: Channel Ack Error not detected 1'b1: Channel Ack Error detected	R/W
b[9]	FERR	Form Error	1'b0: Channel Form Error not detected 1'b1: Channel Form Error detected	R/W
b[8]	SERR	Stuff Error	1'b0: Channel stuff Error not detected 1'b1: Channel stuff Error detected	R/W
b[7]	ALF	Arbitration Lost Flag	1'b0: Channel Arbitration Lost not detected 1'b1: Channel Arbitration Lost detected	R/W
b[6]	BLF	Bus Lock Flag	1'b0: Channel Bus Lock not detected 1'b1: Channel Bus Lock detected	R/W
b[5]	OVLF	Overload Flag	1'b0: Channel Overload not detected 1'b1: Channel Overload detected	R/W
b[4]	BORF	Bus-Off Recovery Flag	1'b0: Channel Bus-Off Recovery not detected 1'b1: Channel Bus-Off Recovery detected	R/W
b[3]	BOEF	Bus-Off Entry Flag	1'b0: Channel Bus-Off Entry not detected 1'b1: Channel Bus-Off Entry detected	R/W

b[2]	EPF	Error Passive Flag	1'b0: Channel Error Passive not detected 1'b1: Channel Error Passive detected	R/W
b[1]	EWF	Error Warning Flag	1'b0: Channel Error Warning not detected 1'b1: Channel Error Warning detected	R/W
b[0]	BEF	Bus Error Flag	1'b0: Channel Bus Error not detected 1'b1: Channel Bus Error detected	R/W

Each Channel Error Flag register shows the status of various error conditions detectable regardless of the setting of the related CAN channel Error Interrupt Enable Register. It also shows the status of the various bus errors detectable by the CAN channel. Refer to the CAN specification (ISO 11898-1) to check when each error condition can occur.

For this register, when only a single bit is to be cleared by the program, do not use bit clear instruction – use the MOV instruction to ensure that only the specified bit is cleared. Then the other bits remain set to 1.

An example in assembler language for clearing the **CFDCnERFL.BEF** bit:

```
mov.b #0Feh, CFDCnERFL ;
```

4.3.4.1 CFDCnERFL.BEF

Bus Error Flag

Indicates a detection of a CAN channel Bus error state, flagged by the bits [14:8] in this register.

This bit can only be cleared by writing 1'b0 to it when it is 1'b1.

This bit can only be set by RS-CAN-FD module logic.

Writing 1'b1 has no influence on the bit value.

The bit is set automatically when a Bus Error is detected.

If the set from the CAN channel occurs simultaneously with the clear by the write access, then the bit is set.

The bit is cleared by writing 1'b0 to it.

It is cleared automatically when the related RS-CAN-FD channel is in CH_RESET mode.

Users can only write to this bit when the related RS-CAN-FD channel is in CH_HALT or CH_OPERATION mode.

4.3.4.2 CFDCnERFL.EWF

Error Warning Flag

This bit indicates if an Error Warning condition has been detected for the CAN channel.

This bit can only be cleared by writing 1'b0 to it when it is 1'b1.

This bit can only be set by RS-CAN-FD module logic.

Writing 1'b1 has no influence on the bit value.

The bit is set automatically when either TEC or REC exceeds 8'd95.

The setting of this bit only occurs when the TEC or REC initially exceed 8'd95. Hence if the TEC or REC remain > 8'd95 and the EWF bit is cleared by the system SW, it will not be set again until both the TEC and REC go below 8'd96 and either TEC or REC cross over again from a value ≤ 8'd95 to a value > 8'd95.

If the set condition occurs simultaneously with the clear condition, the bit is set.

The bit is cleared by writing 1'b0 to it.

It is cleared automatically when the related RS-CAN-FD channel is in CH_RESET mode.

Users can only write to this bit when the related RS-CAN-FD channel is in CH_HALT or CH_OPERATION mode.

4.3.4.3 CFDCnERFL.EPF

Error Passive Flag

Indicates a detection of a CAN channel Error Passive State.

This bit can only be cleared by writing 1'b0 to it when it is 1'b1.

This bit can only be set by RS-CAN-FD module logic.

Writing 1'b1 has no influence on the bit value.

The bit is set automatically when the CAN error state becomes Error Passive State.

The setting of this bit only occurs when the TEC or REC initially exceed 8'd127. Hence if the TEC or REC remain > 8'd127 and the **CFDCnERFL.EPF** bit is cleared by the system SW, it will not be set again until both the TEC and REC go below 8'd128 and either TEC or REC cross over again from a value \leq 8'd127 to a value $>$ 8'd127.

If the set condition occurs simultaneously with the clear condition, the bit is set.

The bit is cleared by writing 1'b0 to it.

It is cleared automatically when the related RS-CAN-FD channel is in CH_RESET mode.

Users can only write to this bit when the related RS-CAN-FD channel is in CH_HALT or CH_OPERATION mode.

4.3.4.4 CFDCnERFL.BOEF

Bus-Off Entry Flag

Indicates a detection of a CAN channel Bus-Off Entry State.

This bit can only be cleared by writing 1'b0 to it when it is 1'b1.

This bit can only be set by RS-CAN-FD module logic.

Writing 1'b1 has no influence on the bit value.

The bit is set automatically when the CAN error state enters the Bus-Off State.

The bit is cleared by writing 1'b0 to it.

It is cleared automatically when the related RS-CAN-FD channel is in CH_RESET mode.

If the set condition occurs simultaneously with the clear condition, the bit is set.

Users can only write to this bit when the related RS-CAN-FD channel is in CH_HALT or CH_OPERATION mode.

4.3.4.5 CFDCnERFL.BORF

Bus-Off Recovery Flag

Indicates a detection of a CAN channel Bus-Off Recovery State

This bit can only be cleared by writing 1'b0 to it when it is 1'b1.

This bit can only be set by RS-CAN-FD module logic.

Writing 1'b1 has no influence on the bit value.

The bit is set automatically if CAN channel recovers from Bus-Off state in the following conditions:

When **CFDCnCTR.BOM** is 2'b00 and normal recovery (11 consecutive recessive bits X 128 times detected) occurs.

When **CFDCnCTR.BOM** is 2'b10 and normal recovery (11 consecutive recessive bits X 128 times detected) occurs.

When **CFDCnCTR.BOM** is 2'b11 and normal recovery (11 consecutive recessive bits X 128 times detected) occurs.

The bit is not set if CAN channel recovers from Bus-Off state in the following conditions:

When CAN Reset Mode is requested

When **CFDCnCTR.RTBO** is instructed (the CAN channel returns to error active)

When **CFDCnCTR.BOM** is 2'b01

When **CFDCnCTR.BOM** is 2'b11 and a Halt Request is asserted before the CAN channel reaches the end of the Bus-Off State.

The bit is cleared by writing 1'b0 to it.

It is cleared automatically when the related RS-CAN-FD channel is in CH_RESET mode.

If the set condition occurs simultaneously with the clear condition, the bit is set.

Users can only write to this bit when the related RS-CAN-FD channel is in CH_HALT or CH_OPERATION mode.

4.3.4.6 CFDCnERFL.OVLF

Overload Flag

Indicates a detection of a CAN channel Overload State.

This bit can only be cleared by writing 1'b0 to it when it is 1'b1.

This bit can only be set by RS-CAN-FD module logic.

Writing 1'b1 has no influence on the bit value.

The bit is set automatically when an overload condition is detected.

If the set condition occurs simultaneously with the clear condition, the bit is set.

The bit is cleared by writing 1'b0 to it.

It is cleared automatically when the related RS-CAN-FD channel is in CH_RESET mode.

Users can only write to this bit when the related RS-CAN-FD channel is in CH_HALT or CH_OPERATION mode.

4.3.4.7 CFDCnERFL.BLF

Bus Lock Flag

Indicates a detection of a CAN channel BusLock condition.

This bit can only be cleared by writing 1'b0 to it when it is 1'b1.

This bit can only be set by RS-CAN-FD module logic.

Writing 1'b1 has no influence on the bit value.

The bit is set automatically when 32 consecutive dominant bits are detected on the CAN bus while the CAN channel is in Operation Mode.

If the set condition occurs simultaneously with the clear condition, the bit is set.

The bit is cleared by writing 1'b0 to it.

It is cleared automatically when the related RS-CAN-FD channel is in CH_RESET mode.

Users can only write to this bit when the related RS-CAN-FD channel is in CH_HALT or CH_OPERATION mode.

4.3.4.8 CFDCnERFL.ALF

Arbitration Lost Flag

Indicates a detection of a CAN channel Bus Arbitration Lost condition.

This bit can only be cleared by writing 1'b0 to it when it is 1'b1.

This bit can only be set by RS-CAN-FD module logic.

Writing 1'b1 has no influence on the bit value.

The bit is set automatically when an arbitration lost condition is detected on the CAN bus while the CAN channel is in Operation Mode.

If the set condition occurs simultaneously with the clear condition, the bit is set.

The bit is cleared by writing 1'b0 to it.

It is cleared automatically when the related RS-CAN-FD channel is in CH_RESET mode.

Users can only write to this bit when the related RS-CAN-FD channel is in CH_HALT or CH_OPERATION mode.

4.3.4.9 CFDCnERFL.SERR

Stuff Error

Indicates a detection of a CAN Stuff Error.

This bit can only be cleared by writing 1'b0 to it when it is 1'b1.

This bit can only be set by RS-CAN-FD module logic.

Writing 1'b1 has no influence on the bit value.

For clearing this bit, the following sequence should be used:

1. Clear the corresponding flag bit
2. Read if the flag bit is cleared
3. If yes, continue, else, go back to step 1.

This bit is set automatically when a Stuff error is detected.

If **CFDCnCTR.ERRD** bit is 1'b1, then this bit will be set if the set and clear conditions occur at the same time for this bit.

The bit is cleared by writing 1'b0 to it.

It is cleared automatically when the related RS-CAN-FD channel is in CH_RESET mode.

If **CFDCnCTR.ERRD** bit is 1'b0 and the set and clear conditions occur at the same time for this bit, it will be cleared if a bit at **CFDCnERFL[14:8]** is already set and it will be set if **CFDCnERFL[14:8]** is 7'b0.

Users can only write to this bit when the related RS-CAN-FD channel is in CH_HALT or CH_OPERATION mode.

4.3.4.10 CFDCnERFL.FERR

Form Error

Indicates a detection of a CAN Form Error.

This bit can only be cleared by writing 1'b0 to it when it is 1'b1.

This bit can only be set by RS-CAN-FD module logic.

Writing 1'b1 has no influence on the bit value.

For clearing this bit, the following sequence should be used:

1. Clear the corresponding flag bit
2. Read if the flag bit is cleared
3. If yes, continue, else, go back to step 1.

This bit is set automatically when a Form error is detected.

If **CFDCnCTR.ERRD** bit is 1'b1, then this bit will be set if the set and clear conditions occur at the same time for this bit.

The bit is cleared by writing 1'b0 to it.

It is cleared automatically when the related RS-CAN-FD channel is in CH_RESET mode.

If **CFDCnCTR.ERRD** bit is 1'b0 and the set and clear conditions occur at the same time for this bit, it will be cleared if a bit at **CFDCnERFL[14:8]** is already set and it will be set if **CFDCnERFL[14:8]** is 7'b0.

Users can only write to this bit when the related RS-CAN-FD channel is in CH_HALT or CH_OPERATION mode.

4.3.4.11 CFDCnERFL.AERR

Acknowledge Error

Indicates a detection of a CAN Acknowledge Error.

This bit can only be cleared by writing 1'b0 to it when it is 1'b1.

This bit can only be set by RS-CAN-FD module logic.

Writing 1'b1 has no influence on the bit value.

For clearing this bit, the following sequence should be used:

1. Clear the corresponding flag bit
2. Read if the flag bit is cleared
3. If yes, continue, else, go back to step 1.

This bit is set automatically when an Acknowledge error is detected.

If **CFDCnCTR.ERRD** bit is 1'b1, then this bit will be set if the set and clear conditions occur at the same time for this bit.

The bit is cleared by writing 1'b0 to it.

It is cleared automatically when the related RS-CAN-FD channel is in CH_RESET mode.

If **CFDCnCTR.ERRD** bit is 1'b0 and the set and clear conditions occur at the same time for this bit, it will be cleared if a bit at **CFDCnERFL[14:8]** is already set and it will be set if **CFDCnERFL[14:8]** is 7'b0.

Users can only write to this bit when the related RS-CAN-FD channel is in CH_HALT or CH_OPERATION mode.

4.3.4.12 CFDCnERFL.CERR

CRC Error

Indicates a detection of a CAN CRC Error.

This bit can only be cleared by writing 1'b0 to it when it is 1'b1.

This bit can only be set by RS-CAN-FD module logic.

Writing 1'b1 has no influence on the bit value.

For clearing this bit, the following sequence should be used:

1. Clear the corresponding flag bit
2. Read if the flag bit is cleared
3. If yes, continue, else, go back to step 1.

This bit is set automatically when a CRC error is detected.

If **CFDCnCTR.ERRD** bit is 1'b1, then this bit will be set if the set and clear conditions occur at the same time for this bit.

The bit is cleared by writing 1'b0 to it.

It is cleared automatically when the related RS-CAN-FD channel is in CH_RESET mode.

If **CFDCnCTR.ERRD** bit is 1'b0 and the set and clear conditions occur at the same time for this bit, it will be cleared if a bit at **CFDCnERFL[14:8]** is already set and it will be set if **CFDCnERFL[14:8]** is 7'b0.

Users can only write to this bit when the related RS-CAN-FD channel is in CH_HALT or CH_OPERATION mode.

4.3.4.13 CFDCnERFL.B1ERR

Bit 1 Error

Indicates a detection of a Recessive bit error.

This bit can only be cleared by writing 1'b0 to it when it is 1'b1.

This bit can only be set by RS-CAN-FD module logic.

Writing 1'b1 has no influence on the bit value.

For clearing this bit, the following sequence should be used:

1. Clear the corresponding flag bit
2. Read if the flag bit is cleared
3. If yes, continue, else, go back to step 1.

This bit is set automatically when a Recessive bit error (expected recessive bit, sampled as dominant bit) is detected.

If **CFDCnCTR.ERRD** bit is 1'b1, then this bit will be set if the set and clear conditions occur at the same time for this bit.

The bit is cleared by writing 1'b0 to it.

It is cleared automatically when the related RS-CAN-FD channel is in CH_RESET mode.

If **CFDCnCTR.ERRD** bit is 1'b0 and the set and clear conditions occur at the same time for this bit, it will be cleared if a bit at **CFDCnERFL[14:8]** is already set and it will be set if **CFDCnERFL[14:8]** is 7'b0.

Users can only write to this bit when the related RS-CAN-FD channel is in CH_HALT or CH_OPERATION mode.

4.3.4.14 CFDCnERFL.B0ERR

Bit 0 Error

Indicates a detection of a Dominant bit error

This bit can only be cleared by writing 1'b0 to it when it is 1'b1.

This bit can only be set by RS-CAN-FD module logic.

Writing 1'b1 has no influence on the bit value.

For clearing this bit, the following sequence should be used:

1. Clear the corresponding flag bit
2. Read if the flag bit is cleared
3. If yes, continue, else, go back to step 1.

This bit is set automatically when a Dominant bit error (expected dominant bit, sampled as recessive bit) is detected.

If **CFDCnCTR.ERRD** bit is 1'b1, then this bit will be set if the set and clear conditions occur at the same time for this bit.

The bit is cleared by writing 1'b0 to it.

It is cleared automatically when the related RS-CAN-FD channel is in CH_RESET mode.

If **CFDCnCTR.ERRD** bit is 1'b0 and the set and clear conditions occur at the same time for this bit, it will be cleared if a bit at **CFDCnERFL[14:8]** is already set and it will be set if **CFDCnERFL[14:8]** is 7'b0.

Users can only write to this bit when the related RS-CAN-FD channel is in CH_HALT or CH_OPERATION mode.

4.3.4.15 CFDCnERFL.ADERR

Acknowledge Delimiter Error

Indicates a detection of a Acknowledge Delimiter bit error.

This bit can only be cleared by writing 1'b0 to it when it is 1'b1.

This bit can only be set by RS-CAN-FD module logic.

Writing 1'b1 has no influence on the bit value.

For clearing this bit, the following sequence should be used:

1. Clear the corresponding flag bit
2. Read if the flag bit is cleared
3. If yes, continue, else, go back to step 1.

This bit is set automatically when a form error is detected during the Acknowledge Delimiter state of frame transmission.

If **CFDCnCTR.ERRD** bit is 1'b1, then this bit will be set if the set and clear conditions occur at the same time for this bit.

The bit is cleared by writing 1'b0 to it.

It is cleared automatically when the related RS-CAN-FD channel is in CH_RESET mode.

If **CFDCnCTR.ERRD** bit is 1'b0 and the set and clear conditions occur at the same time for this bit, it will be cleared if a bit at **CFDCnERFL[14:8]** is already set and it will be set if **CFDCnERFL[14:8]** is 7'b0.

Users can only write to this bit when the related RS-CAN-FD channel is in CH_HALT or CH_OPERATION mode.

4.3.4.16 CFDCnERFL.CRCREG

CRC Register value

The calculated CRC value can be read from these bits, only when **CFDCnCTR.CTME** is 1'b1 for the channel.

If **CFDCnCTR.CTME** bit is 1'b0, then these bits are always read as 0.

These bits show the CAN2.0 CRC value calculated by the RS-CAN-FD channel logic if the CTME bit is enabled.

The **CFDCnERFL.CRCREG** value is updated in the 1st bit of CRC field of the CAN Frame (reception as well as transmission).

These bits are cleared automatically when the related channel is in CH_RESET mode.

4.3.5 CFDCnDCFG

Channel n Data Bitrate Configuration Register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24					
	-	-	-	-		DSJW[3:0]							
Value after reset	0	0	0	0	0	0	0	0					
	b23	b22	b21	b20	b19	b18	b17	b16					
	-	-	-	-		DTSEG2[3:0]							
Value after reset	0	0	0	0	0	0	0	0					
	b15	b14	b13	b12	b11	b10	b9	b8					
	-	-	-			DTSEG1[4:0]							
Value after reset	0	0	0	0	0	0	0	0					
	b7	b6	b5	b4	b3	b2	b1	b0					
					DBRP[7:0]								
Value after reset	0	0	0	0	0	0	0	0					

Bit	Symbol	Bit name	Function	R/W
b[31:28]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[27:24]	DSJW	Resynchronization Jump Width	4'b0000: 1 Tq 4'b0001: 2 Tq : 4'b1111: 16 Tq	R/W
b[23:20]	-	Reserved	This bit is read as 0b. The write value should be always 0b.	R/W
b[19:16]	DTSEG2	Timing Segment 2	4'b0000: Reserved 4'b0001: 2 Tq : 4'b1110: 15 Tq 4'b1111: 16 Tq	R/W
b[15:13]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[12:8]	DTSEG1	Timing Segment 1	5'b00000: Reserved 5'b00001: 2Tq 5'b00010: 3Tq 5'b00011: 4Tq : 5'b11110: 31 Tq 5'b11111: 32 Tq	R/W
b[7:0]	DBRP	Channel Data Baud Rate Prescaler	Data Baud Rate Prescaler division ratio	R/W

This register is used to configure the transmission / reception Data Baud Rate parameters of the channels.

The channel of Classical only mode does not have to perform the configuration of this register.

4.3.5.1 CFDCnDCFG.DBRP

Channel Data Baud Rate Prescaler

These bits are used to define the peripheral bus clock periods contained in a Time Quantum.

This bit cannot be written in CH_OPERATION or CH_SLEEP mode.

Users should write to these bits, only when the RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

4.3.5.2 CFDCnDCFG.DTSEG1

Timing Segment 1

These bits are used to set the segment TSEG1 to compensate for edges on the CAN Bus with a positive phase error.

It also contains the propagation segment.

Any value from 2 to 32 time quanta can be configured.

This bit cannot be written in CH_OPERATION or CH_SLEEP mode.

Users should write to these bits, only when the RS-CAN-FD channel is in CH_RESET or CH_HALT mode.
Users should not write any other value to these bits. See Section 6.1.2 for more details.

4.3.5.3 CFDCnDCFG.DTSEG2

Timing Segment 2

These bits are used to set the segment TSEG2 to compensate for edges on the CAN Bus with a negative phase error.

Any value from 2 to 16 time quanta can be written to these bits.

This bit cannot be written in CH_OPERATION or CH_SLEEP mode.

Users should write to these bits, only when the RS-CAN-FD channel is in CH_RESET or CH_HALT mode.
Users should not write any other value to these bits.

4.3.5.4 CFDCnDCFG.DSJW

Resynchronization Jump Width

These bits set the synchronization jump width. A value from 1 to 16 time quanta can be set.

This bit cannot be written in CH_OPERATION or CH_SLEEP mode.

Users should write to these bits, only when the RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

4.3.6 CFDCnFDCFG

Channel n CAN-FD Configuration Register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
	CFDTE	CLOE	REFE	FDOE	-	GWBRs	GWDF	GWEN
Value after reset	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
	TDCO[7:0]							
Value after reset	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
	-	-	RPNMD[1:0]		-	ESIC	TDCE	TDCOC
Value after reset	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
	-	-	-	-	-	EOCCFG[2:0]		
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31]	CFDTE	CAN-FD Tolerance enable	1'b0: CAN-FD Tolerance mode disabled. 1'b1: CAN-FD Tolerance mode enabled	R/W
b[30]	CLOE	Classical CAN only enable	1'b0: Classical only mode disabled 1'b1: Classical only mode enabled	R/W
b[29]	REFE	RX edge filter enable	1'b0: RX edge filter disabled 1'b1: RX edge filter enabled	R/W
b[28]	FDOE	FD only enable	1'b0: FD only mode disabled 1'b1: FD only mode enabled	R/W
b[27]	-	Reserved	This bit is read as 0b. The write value should be always 0b.	R/W
b[26]	GWBRs	Gateway BRS configuration bit	1'b0: GW frame is transmitted with BRS = 1'b0 1'b1: GW frame is transmitted with BRS = 1'b1	R/W
b[25]	GWDF	Gateway FDF configuration bit	1'b0: GW frame is transmitted as Classical CAN frame 1'b1: GW frame is transmitted as CAN-FD frame	R/W
b[24]	GWEN	CAN2.0, CAN-FD <> CAN2.0, CAN-FD Multi Gateway Enable	1'b0: Multi Gateway Disabled 1'b1: Multi Gateway Enable	R/W
b[23:16]	TDCO	Transceiver Delay Compensation Offset	Transceiver delay compensation offset value	R/W
b[15:14]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[13:12]	RPNMD	Return Pretended Network Filter Mode	2'b00: Return to Acceptance Filter Mode 2'b01: Return to Pretended Network Filter ID only and Acceptance Filter Mode 2'b10: Return to Pretended Network Filter and Acceptance Filter Mode 2'b11: Return to Pretended Network Filter Mode (Not return)	R/W
b[11]	-	Reserved	This bit is read as 0b. The write value should be always 0b.	R/W
b[10]	ESIC	Error State Indication Configuration	1'b0: The ESI bit in the frame will be representing the Error state of the node itself 1'b1: The ESI bit in the frame will be representing the Error state of message buffer if the node itself is not in error passive. If the node is in Error Passive then the ESI bit will be driven by the node itself	R/W
b[9]	TDCE	Transceiver Delay Compensation Enable	1'b0: Transceiver Delay Compensation disabled 1'b1: Transceiver Delay Compensation enabled	R/W
b[8]	TDCOC	Transceiver Delay Compensation Offset Configuration	1'b0: Measured + offset 1'b1: offset only	R/W
b[7:3]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[2:0]	EOCCFG		3'b000: All Transmitter or Receiver CAN Frames	R/W

Error Occurrence Counter Configuration	3'b001: All Transmitter CAN Frames
	3'b010: All Receiver CAN Frames
	3'b011: Reserved
	3'b100: Only Transmitter or Receiver CAN-FD Data-Phase (fast bits)
	3'b101: Only Transmitter CAN-FD Data-Phase (fast bits)
	3'b110: Only Receiver CAN-FD Data-Phase (fast bits)
	3'b111: Reserved

The Channel n CAN-FD Configuration Register is used to configure which communication direction (transmitter and/or receiver) errors will be counted.

4.3.6.1 CFDCnFDCFG.EOCCFG

Error Occurrence Counter Configuration

These bits select which type of CAN frame configuration and direction, protocol errors are counted in.

This bit cannot be written in CH_OPERATION or CH_SLEEP mode.

Users should only write to these bits when the RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

4.3.6.2 CFDCnFDCFG.TDCOC

Transceiver Delay Compensation Offset Configuration

This bit selects which offset is used when defining the position of the secondary sample point (SSP) for the RS-CAN-FD channel. If the bit is set to 1'b0 then the position of the SSP is the measured Transceiver delay plus the fixed offset. If the bit is 1'b1 then the position of the SSP is defined only by the offset.

This bit cannot be written in CH_OPERATION or CH_SLEEP mode.

Users should only write to these bits when the RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

Users should not set to this bit when Classical only mode.

4.3.6.3 CFDCnFDCFG.TDCE

Transceiver Delay Compensation Enable

This bit enables the Transceiver Delay Compensation for the RS-CAN-FD channel.

This bit cannot be written in CH_OPERATION or CH_SLEEP mode.

Users should only write to these bits when the RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

Users should not set to this bit when Classical only mode.

4.3.6.4 CFDCnFDCFG.ESIC

Error State Indication Configuration

Bus controllers to be used as CAN to CAN gateways shall support that in every forwarded CAN FD message the ESI flag is not changed to reflect the status of the gateway/bridge/router but instead the ESI flag is sent as it was in the original message.

This Error State Indication Configuration, controls the sending of either the own ESI flag information or the message ESI flag information (**CFDCFFDCSTS.CFESI** or **CFDTMFDFCTR.TMESI**)

This bit cannot be written in CH_OPERATION or CH_SLEEP mode.

Users should only write to these bits when the RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

Users should not set to this bit when Classical only mode.

4.3.6.5 CFDCnFDCFG.RPNMD

Return Pretended Network Filter Mode

This bit selects the mode which returns from Pretended Network mode (**CFDCnFDSTS.PNSTS=11b**).

When the received frame passes the Pretended Network filter (ID & Payload) in Pretended Network mode and reception of a frame has completed, RS-CAN-FD changes the mode from Pretended Network mode to selected mode.

This bit cannot be written in CH_OPERATION or CH_SLEEP mode.

4.3.6.6 CFDCnFDCFG.TDCO

Transceiver Delay Compensation Offset

These bits set the secondary sample point offset. How this value is used, depends on the **CFDCnFDCFG.TDCOC** setting.

If **CFDCnFDCFG.TDCOC = 1'b0**, the Transceiver Delay Compensation result is equal to the Trv_Delay (measured delay) + the value in **CFDCnFDCFG.TDCO**, rounded down to the nearest integer number of time quanta. Otherwise, the Transceiver Delay Compensation result is equal to the value in **CFDCnFDCFG.TDCO**. Refer to Section 6.1.5 for details on how **CFDCnFDCFG.TDCO** is used.

The actual offset value is interpreted as TDCO + 1. E.g if 4 is set in TDCO, the offset is 5 clock cycle.

Clock cycle is 1 cycle of CAN channel DLL clock.

This bit cannot be written in CH_OPERATION or CH_SLEEP mode.

Users should write to these bits, only when the RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

Users should not set to this bit when Classical only mode.

4.3.6.7 CFDCnFDCFG.GWEN

CAN2.0, CAN-FD <> CAN2.0, CAN-FD Multi Gateway Enable

When this bit is enabled then a multi gateway is enabled. Message received on one node can be routed to another node by using the COM FIFO when they are configured as GW FIFO (**CFDCFCCd.CFM = 2'b10**).

Moreover, when TX Queue is set as GW mode, the message received on one node can be stored in TX Queue, and it can be sent to another node.

The FDF and BRS bit of the routed message can be changed by the configuration value of the **CFDCnFDCFG.GWFDF** and **CFDCnFDCFG.GWBRS**.

By this the transmitted value of these bits can be replaced.

Example :

CFDCnFDCFG.GWEN = 1 on channel y

CFDCnFDCFG.GWFDF = 1

If a Classical CAN frame is received on channel x and routed to a GW FIFO or TX Queue of channel y. Then this CAN frame is sent on Channel y as a CAN-FD frame, due to the **CFDCnFDCFG.GWFDF** bit.

The table below shows how the message information is changed depending on received and configured data.

Routed CAN frame	Routed Received DLC	CAN BRS	Configured CFDCnFDCFG.GWFDF bit	GW message DLC	GW message BRS bit	GW message frame type
CAN2.0	≤ 8	N/A	1	≤ 8	Based on configuration CFDCnFDCFG.GWBRS	FD
CAN2.0	> 8	N/A	1	$= 8$	Based on configuration CFDCnFDCFG.GWBRS	FD
FD	≤ 8	?	1	≤ 8	Based on configuration	FD

					CFDCnFDCFG.GWB RS	
FD	> 8	?	1	> 8	Based on configuration CFDCnFDCFG.GWB RS	FD
CAN2.0	≤ 8	N/A	0	≤ 8	N/A	CAN2.0
CAN2.0	> 8	N/A	0	> 8	N/A	CAN2.0
FD	≤ 8	?	0	≤ 8	N/A	CAN2.0
FD	> 8	?	0	= 8	N/A	CAN2.0

Note this Gateway is limited to 8 byte data payload for different frame type. If routing and target frame type is same then the DLC value is untouched.

In case the source frame was a CAN-FD with more than 8 data byte, then on classical destination node the Data payload is cut down to 8 byte. Only 8 bytes of top data performs GW transmission, and the remaining byte cancels it.

In case the source frame was a CLASSICAL with a DLC value > 8, then on CAN-FD destination node the DLC value itself is changed to 8.

Note: Other transmission buffer than the GW-FIFO are not affected by this feature.

Remote frames should not be routed via the Gateway, when **CFDCnFDCFG.GWEN** is set.

When a destination node is **CFDCnFDCFG.FDOE=1**, set **CFDCnFDCFG.GWEN=1** and **CFDCnFDCFG.GWFDF=1**.

When a destination node is **CFDCnFDCFG.CLOE=1**, set **CFDCnFDCFG.GWEN=1** and **CFDCnFDCFG.GWFDF=0**.

This bit cannot be written in CH_OPERATION or CH_SLEEP mode.

Users should only write to these bits when the RS-CAN-FD channel is in CH_RESET mode.

4.3.6.8 CFDCnFDCFG.GWFDF

Gateway FDF configuration bit

When the bit **CFDCnFDCFG.GWEN** is set to 1'b1 then the FDF bit of the transmitting GW frame will be replaced by the value of **CFDCnFDCFG.GWFDF**.

This bit cannot be written in CH_OPERATION or CH_SLEEP mode.

Users should only write to these bits when the RS-CAN-FD channel is in CH_RESET mode.

Users should not set to this bit when Classical only mode.

4.3.6.9 CFDCnFDCFG.GBRS

Gateway BRS configuration bit

When the bit **CFDCnFDCFG.GWEN** is set to 1'b1 then the BRS bit of the transmitting GW frame will be replaced by the value of **CFDCnFDCFG.GBRS**.

In classical CAN frames the BRS bit has no meaning

This bit cannot be written in CH_OPERATION or CH_SLEEP mode.

Users should only write to these bits when the RS-CAN-FD channel is in CH_RESET mode.

Users should not set to this bit when Classical only mode.

4.3.6.10 CFDCnFDCFG.FDOE

FD only enable

This bit enables the reception and transmission of CAN-FD only frames. If enabled then communication in Classical CAN frame format is disabled. Transmission of classical CAN frames is not possible, the FDF bit of the message buffer is don't care (**CFDCFFDCSTS.CFFDF / CFDTMFDFCTR.TMFDF**). If messages with classical CAN frame format is received then the protocol controller will treat them as invalid frames and response with error frames.

In case a Classical frame is configured for transmitting, the FDF bit is sent as recessive, so a FD frame is sent. If the DLC is configured bigger than 8 the remaining data bytes are padded with CCh.

User should not set **CFDCnFDCFG.FDOE** and **CFDCnFDCFG.CLOE** simultaneously.

This bit cannot be written in CH_OPERATION, CH_HALT and CH_SLEEP mode.

4.3.6.11 CFDCnFDCFG.REFE

RX edge filter enable

This bit enables the RX edge filter during the IDLE detection (bus integration). When it is enabled then 2 consecutive dominant TQ required to detect a synchronization edge.

This bit cannot be written in CH_OPERATION, CH_HALT and CH_SLEEP mode.

Users should not set to this bit when Classical only mode *and CFDCnFDCFG.CFDTE = 1'b0 (not enable CAN-FD Tolerance mode)*.

4.3.6.12 CFDCnFDCFG.CLOE

Classical CAN only enable

This bit enables the Classical CAN only mode. If this bit is 1'b1, then the protocol controller can only send Classical Frames and will react with a Form or CRC error on FD frames.

User should not set **CFDCnFDCFG.CLOE** and **CFDCnFDCFG.FDOE** simultaneously.

CFDCnFDCFG.CLOE	CFDCnFDCFG.FDOE	channel mode
0	0	CAN-FD mode
0	1	FD only mode
1	0	Classical CAN only mode
1	1	Reserved

This bit cannot be written in CH_OPERATION, CH_HALT and CH_SLEEP mode.

Users should only write to these bits when the RS-CAN-FD channel is in CH_RESET mode.

4.3.6.13 CFDCnFDCFG.CFDTE

CAN-FD Tolerance enable

This bit enables the CAN-FD Tolerance mode. This bit is required only for Classical CAN only mode. (CFDCnFDCFG.CLOE = 1'b1)

If this bit is 1'b0, then the protocol controller can only send Classical Frames and will react with a Form or CRC error on FD frames.

If this bit is 1'b1, then the protocol controller behaves according to the ISO 11898-1 (DIS 2015). If the FDF bit is detected recessive then the protocol controller enters the protocol exception state, and will try to integrate back to the CAN communication; the module is CAN-FD tolerant.

This bit cannot be written in CH_OPERATION, CH_HALT and CH_SLEEP mode.

Users should only write to these bits when the RS-CAN-FD channel is in CH_RESET mode.

4.3.7 CFDCnFDCTR

Channel n CAN-FD Control Register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
KEY[7:0]								
Value after reset	0	0	0	0	0	0	0	0
PNMDC[1:0]								
Value after reset	-	-	-	-	-	-	0	0
Value after reset	0	0	0	0	0	0	0	0
SOCCLR EOCCCLR								
Value after reset	-	-	-	-	-	-	0	0
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:24]	KEY	Key code	These bits control the right or wrong of rewriting of PNMDC bit.	W
b[23:18]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[17:16]	PNMDC	Pretended Network Filter Mode Control	2'b00: Acceptance Filter Mode request 2'b01: Pretended Network Filter ID only and Acceptance Filter Mode request 2'b10: Pretended Network Filter and Acceptance Filter Mode request 2'b11: Pretended Network Filter Mode request	R/W
b[15:2]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[1]	SOCCLR	Successful Occurrence Counter Clear	1'b0: No Successful Occurrence Counter clear 1'b1: Clear Successful Occurrence Counter	R/W
b[0]	EOCCCLR	Error Occurrence Counter Clear	1'b0: No Error Occurrence Counter clear 1'b1: Clear Error Occurrence Counter	R/W

The Channel n CAN-FD Control Register is used to control the error and successful occurrence counters.

4.3.7.1 CFDCnFDCTR.EOCCCLR

Error Occurrence Counter Clear

This bit is used to clear the error occurrence counter.

This bit cannot be set in CH_SLEEP mode or CH_RESET.

Read value is always 0.

This bit is cleared automatically by the RS-CAN-FD logic.

This bit is cleared automatically when the related RS-CAN-FD channel is in CH_RESET mode.

4.3.7.2 CFDCnFDCTR.SOCCCLR

Successful Occurrence Counter Clear

This bit is used to clear the Successful Occurrence Counter.

This bit cannot be set in CH_SLEEP mode or CH_RESET.

Read value is always 0.

This bit is cleared automatically by the RS-CAN-FD logic.

This bit is cleared automatically when the related RS-CAN-FD channel is in CH_RESET mode.

4.3.7.3 CFDCnFDCTR.PNMDC

Pretended Network Filter Mode Control

This bit selects the mode about Acceptance Filter or Pretended Network Filter.

This bit is set the value of **CFDCnFDCTR.PNMDC** automatically when the received frame passes the Pretended Network filter in Pretended Network mode and reception of a frame has completed.

If the set from the HW occurs simultaneously with the set by the CPU, then the bit is set by HW.

These bits are cleared automatically when the related RS-CAN-FD channel is in CH_RESET mode.

Users should only write the value of **CFDCnFDCTR.PNMDC**, or the value of Pretended Network mode (2'b11) to these bits.

4.3.7.4 CFDCnFDCTR.KEY

Key code

When C4h is written in these bits, the write of **CFDCnFDCTR.PNMDC** bit becomes available.

Read value from these bits is always 8'h0.

Users should write a **CFDCnFDCTR.PNMDC** bit and the **CFDCnFDCTR.KEY** bit simultaneously.

4.3.8 CFDCnFDSTS

Channel n CAN-FD Status Register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
SOC[7:0]								
Value after reset	0	0	0	0	0	0	0	0
EOC[7:0]								
Value after reset	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
Value after reset	0	0	0	0	0	0	0	0
TDCR[7:0]								
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:24]	SOC	Successful occurrence counter register	These bits show the successful occurrence counter value.	R
b[23:16]	EOC	Error occurrence counter register	These bits show the error occurrence counter value.	R
b[15]	TDCVF	Transceiver Delay Compensation Violation Flag	1'b0: Transceiver Delay Compensation Violation has not occurred 1'b1: Transceiver Delay Compensation Violation has occurred	R/W
b[14]	-	Reserved	This bit is read as 0b. The write value should be always 0b.	R/W
b[13:12]	PNSTS	Pretended Network Filter State	2'b00: Acceptance Filter Mode 2'b01: Pretended Network Filter ID only and Acceptance Filter Mode 2'b10: Pretended Network Filter and Acceptance Filter Mode 2'b11: Pretended Network Filter Mode	R
b[11:10]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[9]	SOCO	Successful occurrence counter overflow	1'b0: Successful occurrence counter has not overflowed 1'b1: Successful occurrence counter has overflowed	R/W
b[8]	EOCO	Error occurrence counter overflow	1'b0: Error occurrence counter has not overflowed 1'b1: Error occurrence counter has overflowed	R/W
b[7:0]	TDCR	Transceiver Delay Compensation Result	Transceiver delay compensation result	R

The Channel n CAN-FD Status Register indicates the transceiver compensation delay result and its related FIFO message lost status.

4.3.8.1 CFDCnFDSTS.TDCR

Transceiver Delay Compensation Result

The measured delay is a multiple of the CAN channel DLL clock.

This result depends on the **CFDCnFDCFG.TDCOC** configuration and the offset value in **CFDCnFDCFG.TDCO**. Refer to Section 6.1.5 for details on how this value is derived.

These bits are set when the transceiver delay has been measured.

These bits are updated at the falling edge between FDF and res bit if **CFDCnFDCFG.TDCOC = 1'b0** and the transceiver delay compensation is enabled (**CFDCnFDCFG.TDCE = 1'b1**).

These bits are cleared automatically when the related RS-CAN-FD channel is in CH_RESET mode.

4.3.8.2 CFDCnFDSTS.EOCO

Error occurrence counter overflow

Indicates if the related CAN channel error occurrence counter has overflowed.

Writing 1'b1 has no influence on the bit value.

This bit is set automatically when **CFDCnFDSTS.EOC** is 8'hFF and a CAN bus error is detected based on the configuration defined in **CFDCnFDCFG.EOCCFG**.

This bit is cleared by writing a 1'b0 to it.

This bit is cleared automatically when the related RS-CAN-FD channel is in CH_RESET mode.

Users can only write to this bit when the related RS-CAN-FD channel is in CH_HALT or CH_OPERATION mode.

Do not use bit clear instruction for clearing this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Then the other bits remain set to 1'b1.

4.3.8.3 CFDCnFDSTS.SOCO

Successful occurrence counter overflow

Indicates if the related CAN channel successful occurrence counter has overflowed.

Writing 1'b1 has no influence on the bit value.

This bit is set automatically when **CFDCnFDSTS.SOC** is 8'hFF and, successful message reception or successful message transmission occurs.

This bit is cleared by writing a 1'b0 to it.

This bit is cleared automatically when the related RS-CAN-FD channel is in CH_RESET mode.

Users can only write to this bit when the related RS-CAN-FD channel is in CH_HALT or CH_OPERATION mode.

Do not use bit clear instruction for clearing this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Then the other bits remain set to 1'b1.

4.3.8.4 CFDCnFDSTS.PNSTS

Pretended Network Filter State

This bit indicates the current mode about Acceptance Filter or Pretended Network Filter.

In the case of under reception, this bit reflects the value of **CFDCnFDCTR.PNMDC** after reception.

In the other case, this bit reflects the value of **CFDCnFDCTR.PNMDC** immediately.

These bits are cleared automatically when the related RS-CAN-FD channel is in CH_RESET mode.

4.3.8.5 CFDCnFDSTS.TDCVF

Transceiver Delay Compensation Violation Flag

The RS-CAN-FD module is capturing internally the transmitted data bit by a Bit.

This data is compared against the received CAN-Bus level which is delayed by the Transceiver loop delay.

The Transceiver delay has some variation depending on physical parameters like temperature. The result flag **CFDCnFDSTS.TDCR** will be updated by every message. Hence temporary max. delay violation could be missed. This bit is capturing this violation.

Writing 1'b1 has no influence on the bit value.

This bit is set automatically when the Transceiver Delay Compensation is greater than the max. delay compensation (6 data bit times – 2clk_dlc) and the internal Bit is overrun.

This bit is cleared by writing a 1'b0 to it.

This bit is cleared automatically when the related RS-CAN-FD channel is in CH_RESET mode.

Users can only write to this bit when the related RS-CAN-FD channel is in CH_HALT or CH_OPERATION mode.

Do not use bit clear instruction for clearing this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Then the other bits remain set to 1'b1.

4.3.8.6 CFDCnFDSTS.EOC

Error occurrence counter register

This counter is used together with the **CFDCnFDSTS.SOC** counter to support an option for host-controlled fall-back to payload bit rate identical to arbitration bit rate when messages utilizing reduced payload bit length experience significantly higher error rates compared to other messages.

This higher error rate can be detected depending on the configuration of the **CFDCnFDCFG EOCCFG**

These bits are set only by RS-CAN-FD module logic.

Writing any value has no influence on these bits.

These bits are updated (incremented) when an error occurs, according to the configuration of the **CFDCnFDCFG EOCCFG** bits. When the counter reaches the value of 8'hFF then updating is stopped.

These bits are cleared by writing 1'b1 to **CFDCnFDCTR EOCCCLR**

These bits are cleared automatically when the related RS-CAN-FD channel is in CH_RESET mode.

4.3.8.7 CFDCnFDSTS.SOC

Successful occurrence counter register

These bits are set only by RS-CAN-FD module logic.

Writing any value has no influence on these bits.

These bits are updated (incremented) when the occurrence of any error-free messages on the bus is detected. Any means (received or transmitted). When the counter reaches the value of 8'hFF then updating is stopped.

Note in case of Loopback mode the counter would be incremented twice.

These bits are cleared by writing 1'b1 to **CFDCnFDCTR SOCCLR**

These bits are cleared automatically when the related RS-CAN-FD channel is in CH_RESET mode.

4.3.9 CFDCnFDCRC

Channel n CAN-FD CRC Register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
	-	-	-	-		SCNT[3:0]		
Value after reset	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
	-	-	-		CRCREG[20:16]			
Value after reset	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
				CRCREG[15:8]				
Value after reset	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
				CRCREG[7:0]				
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:28]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[27:24]	SCNT	Stuff bit count	These bits shows the stuff bit count (mod 8) for the CAN FD frame	R
b[23:21]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[20:0]	CRCREG	CRC Register value	These bits show the CRC value calculated for the CAN-FD Frame.	R

The CRC Register holds the CRC value calculated for the CAN-FD Frame

4.3.9.1 CFDCnFDCRC.CRCREG

CRC Register value

The calculated CRC value can be read from these bits, only when **CFDCnCTR.CTME** is 1'b1 for the channel.

If **CFDCnCTR.CTME** bit is 1'b0, then these bits are always read as 0.

If CRC_17 (17 bit CRC) is used, then bits [20:17] are always read as 0.

These bits show the CRC value calculated by the RS-CAN-FD channel logic if the **CFDCnCTR.CTME** bit is enabled.

The **CFDCnFDCRC.CRCREG** value is updated in the 1st bit of CRC field of the CAN-FD Frame (reception as well as transmission).

These bits are cleared automatically when the related RS-CAN-FD channel is in CH_RESET mode.

4.3.9.2 CFDCnFDCRC.SCNT

Stuff bit count

These bits show the Stuff Count value of the CAN-FD frame. It shows the number of inserted stuff bits (modulo 8, Gray-coded) for a CAN FD frame if the **CFDCnCTR.CTME** bit is enabled on **CFDCnFDCRC.SCNT[3:1]**. And the corresponding Parity bit to this counter value on **CFDCnFDCRC.SCNT[0]**.

If **CFDCnCTR.CTME** bit is 1'b0, then these bits are always read as 0.

The **CFDCnFDCRC.SCNT** value is updated in the 1st bit of CRC field of the CAN-FD Frame (reception as well as transmission).

These bits are cleared automatically when the related RS-CAN-FD channel is in CH_RESET mode.

4.3.10 CFDGIPV

Global IP Version Register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
	-	-			PSI[13:8]			
Value after reset	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
				PSI[7:0]				
Value after reset	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
	-	-	-	-	-	-	IPT[1:0]	
Value after reset	0	0	0	0	0	0	0	1
	b7	b6	b5	b4	b3	b2	b1	b0
				IPV[7:0]				
Value after reset	0	1	0	0	0	0	1	1

Bit	Symbol	Bit name	Function	R/W
b[31:30]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[29:16]	PSI	Parameter Status Information	These bits shows the status of a parameter.	R
b[15:10]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[9:8]	IPT	IP Type Release Number	These bits show the IP type used in the Product	R
b[7:0]	IPV	IP Version Release Number	These bits show the IP version used in the Product	R

The Global IP Version Register show the release version of the RS-CAN-FD IP

4.3.10.1 CFDGIPV.IPV

IP Version Release Number

These bits **IPV[7:4]** show the main release number and bits **IPV[3:0]** show the sub release number.

Examples:

*8'h10 = IP Release 1.0.
 8'h11 = IP Release 1.1.
 8'h20 = IP Release 2.0.
 8'h21 = IP Release 2.1
 8'h30 = IP Release 3.0
 8'h31 = IP Release 3.a
 8'h40 = IP Release 4.0
 8'h41 = IP Release 4.a
 8'h42 = IP Release 4.1
 8'h43 = IP Release 4.1a*

4.3.10.2 CFDGIPV.IPT

IP Type Release Number

2'b00 = IP Type RS-CAN

2'b01 = IP Type RS-CAN-FD

In the case of external terminal rs_classic_only=0, IPT is set to 2'b01.

In the case of external terminal rs_classic_only=1, IPT is set to 2'b00.

CFDGIPV.IPT	CFDGIPV.IP V	RSCANFD Ver	Protocol engine Ver	IP Name	Specification
2'b01	8'h10	RS-CANFDv1	RS-CANFD_PE V1.0	uciaprcn007 0	Bosch Specification: CAN with Flexible Data-Rate Ver.1.0
	8'h20	RS-CANFDv2	RS-CANFD_PE V2.0	uciaprcn009 0	CAN-FD ISO 11898-1 (DIS 2015)
	8'h30	RS-CANFDv3	RS-CANFD_PE V2.0	uciaprcn011 0	CAN-FD ISO 11898-1 (DIS 2015) Buffer enhancement version
	8'h31	RS-CANFDv3a	RS-CANFD_PE V2.0	uciaprcn011 1	CAN-FD ISO 11898-1 (2015) GW function enhancement version (DENSO spec include)
	8'h40	RS-CANFDv4	RS-CANFD_PE V3.0	uciaprcn013 0	CAN-FD ISO 11898-1 (2015) GW function enhancement version (crossdomain spec include))
	8'h41	RS-CANFDv4.a RS-CANFDv4.b RS-CANFD V4.0	RS-CANFD_PE V3.0	uciaprcn013 1 uciaprcn013 2	CAN-FD ISO 11898-1 (2015) GW function enhancement version (crossdomain spec include and bug fix version))
	8'h42	RS-CANFD V4.1	RS-CANFD_PE V3.0	uciaprcn014 0	CAN-FD ISO 11898-1 (2015) parameter setting version
	8'h43	RS-CANFD V4.1a	RS-CANFD_PE V3.0	uciaprcn014 1	CAN-FD ISO 11898-1 (2015) parameter setting version bit extension version of can_race_ts terminal

*8'h43 of CFDGIPV.IPV overlaps with the version of RS-CANFD Lite.
Therefore, CFDGIPV.IPT is modified when revising RS-CANFD Lite.*

4.3.10.3 CFDGIPV.PSI

Parameter Status Information

These bits **PSI [13:0]** show the status of a parameter.

These bits are valid for IPV=8'h40 or more.

b[29:27] Number of channel

3'b000 = channel number 8
3'b010 = channel number 6
3'b100 = channel number 4
3'b110 = channel number 2
other = Reserved

b[26:24] Number of TXMB

3'b000 = TXMB number 64/ch
3'b001 = TXMB number 32/ch
3'b010 = TXMB number 16/ch
other = Reserved

b[23:21] Number of AFL entry

3'b000 = AFL entry number 192/ch
3'b001 = AFL entry number 128/ch
3'b010 = AFL entry number 64/ch
other = Reserved

b[20:17] Number of Pool Buffer

4'b0000 = Pool Buffer number 256/ch
4'b0001 = Pool Buffer number 128/ch
4'b0010 = Pool Buffer number 64/ch
4'b0011 = Pool Buffer number 48/ch
4'b0100 = Pool Buffer number 32/ch
other = Reserved

b[16] Same ID overwrite function of TXQ

1'b0 = Implementation
1'b1 = Not implementation

4.3.11 CFDGCFG

Global Configuration Register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
ITRCP[15:8]								
Value after reset	0	0	0	0	0	0	0	0
ITRCP[7:0]								
Value after reset	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
TSBTCS[2:0]				TSSS	TSP[3:0]			
Value after reset	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset	-	-	CMPOC	DCS	MME	DRE	DCE	TPRI

Bit	Symbol	Bit name	Function	R/W
b[31:16]	ITRCP	Interval Timer Reference Clock Prescaler	FIFO Interval timer prescaler value	R/W
b[15:13]	TSBTCS	Timestamp Bit Time Channel Select	3'b000: select clock from Channel 0 3'b001: select clock from Channel 1 3'b010: select clock from Channel 2 3'b011: select clock from Channel 3 3'b100: select clock from Channel 4 3'b101: select clock from Channel 5 3'b110: select clock from Channel 6 3'b111: select clock from Channel 7	R/W
b[12]	TSSS	Timestamp Source Select	1'b0: Source clock for Timestamp counter is peripheral clock 1'b1: Source clock for Timestamp counter is bit time clock	R/W
b[11:8]	TSP	Timestamp Prescaler	4'b0000: Timestamp Prescaler = 1 4'b0001: Timestamp Prescaler = 2 4'b0010: Timestamp Prescaler = 4 4'b0011: Timestamp Prescaler = 8 : 4'b1101: Timestamp Prescaler = 8192 4'b1110: Timestamp Prescaler = 16384 4'b1111: Timestamp Prescaler = 32768	R/W
b[7:6]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[5]	CMPOC	CAN-FD message Payload overflow configuration	1'b0: Message is rejected 1'b1: Message payload is cut to fit to configured message size	R/W
b[4]	DCS	Data Link Controller Clock Select	1'b0: Internal clean clock 1'b1: External Clock source connected to clk_xincan pin	R/W
b[3]	MME	Mirror Mode Enable	1'b0: Mirror Mode disabled 1'b1: Mirror Mode enabled	R/W
b[2]	DRE	DLC Replacement Enable	1'b0: DLC replacement disabled 1'b1: DLC replacement enabled	R/W
b[1]	DCE	DLC Check Enable	1'b0: DLC check disabled	R/W

			1'b1: DLC check enabled	
b[0]	TPRI	Transmission Priority	1'b0: ID Priority	R/W
			1'b1: Message Buffer Number Priority	

The Global Configuration Register is used to select the transmission priority to be used for all the TX Message Buffers and the clock source for the CAN protocol engine of all CAN channels. It is also used to select the source for the timestamp clock and to configure the frequency for the timestamp clock and interval timer reference clock.

4.3.11.1 CFDGCFG.TPRI

Transmission Priority

This bit selects the transmission priority for all CAN channels.

Users cannot write to this bit in GL_SLEEP mode.

Message Buffer Number Priority should not be used together with TX Queue transmission.

Users should only write to this bit when RS-CAN-FD module is in GL_RESET mode.

4.3.11.2 CFDGCFG.DCE

DLC Check Enable

This bit enables the DLC check for all CAN channels.

Users cannot write to this bit in GL_SLEEP mode.

Users should only write to this bit when RS-CAN-FD module is in GL_RESET mode.

4.3.11.3 CFDGCFG.DRE

DLC Replacement Enable

If this bit is 1'b1 and **CFDGCFG.DCE** is 1'b1, then RS-CAN-FD will store the configured value (**CFDGAFLP0r.GAFLDLC**) of DLC in the destination RX Message Buffer or FIFO buffer if the DLC check passes. Otherwise the DLC value in the destination RX Message Buffer or FIFO buffer is unchanged.

Users cannot write to this bit in GL_SLEEP mode.

Users should only write to this bit when RS-CAN-FD module is in GL_RESET mode.

4.3.11.4 CFDGCFG.MME

Mirror Mode Enable

This bit enables the Mirror Mode for all CAN channels.

Users cannot write to this bit in GL_SLEEP mode.

Users should only write to this bit when RS-CAN-FD module is in GL_RESET mode.

4.3.11.5 CFDGCFG.DCS

Data Link Controller Clock Select

This bit selects the clock source for the CAN communications. Internal clean clock has a smaller clock jitter than the Peripheral clock (pclk).

Users cannot write to this bit in GL_SLEEP & GL_OPERATION mode.

Users should only write to this bit when RS-CAN-FD module is in GL_RESET mode.

4.3.11.6 CFDGCFG.CMPOC

CAN-FD message Payload overflow configuration

The received message payload is always compared with the available message payload size in the Message Buffer. This bit controls the message payload acceptance mechanism in the case when the received payload is higher than the Message Buffer payload size **CFDRMNB.RMPLS**, **CFDRFCCA.RFPLS**, **CFDCFCCd.CFPLS**.

Users cannot write to this bit in GL_SLEEP & GL_OPERATION mode.

Users should only write to this bit when RS-CAN-FD module is in GL_RESET mode.

When this bit is set and payload overflow occurs, DLC value is stored in a RXMB or FIFO without changing.

4.3.11.7 CFDGCFG.TSP

Timestamp Prescaler

The value configured in these bits defines the period of the clock source used for the Timestamp counter.

Users cannot write to this bit in GL_SLEEP mode.

Users should only write to this bit when RS-CAN-FD module is in GL_RESET mode.

4.3.11.8 CFDGCFG.TSSS

Timestamp Source Select

This bit allows the selection of the clock source for the Timestamp counter.

Users cannot write to this bit in GL_SLEEP mode.

Users should only write to this bit when RS-CAN-FD module is in GL_RESET mode.

Users should not write 1'b1 when CAN-FD communication will be used.

Note bit time clock could be variable depending on the nominal and data rate bit configuration.

4.3.11.9 CFDGCFG.TSBTCS

Timestamp Bit Time Channel Select

These bits allow the selection of the Bit Time clock of a particular channel for the Timestamp counter.

Users cannot write to this bit in GL_SLEEP mode.

Users should only write to this bit when RS-CAN-FD module is in GL_RESET mode.

4.3.11.10 CFDGCFG.ITRCP

Interval Timer Reference Clock Prescaler

These bits allow the definition of a reference clock for the FIFO interval timer source clock.

When the **CFDGCFG.ITRCP[15:0]** bits are 16'h0000, then the timer is disabled.

Users cannot write to this bit in GL_SLEEP mode.

Users should only write to this bit when RS-CAN-FD module is in GL_RESET mode.

4.3.12 CFDGCTR

Global Control Register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
Value after reset	-	-	-	-	-	-	-	-

	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset	-	-	-	-	-	-	TSWR	TSRST

	b15	b14	b13	b12	b11	b10	b9	b8
Value after reset	MOWEIE	QMEIE	-	QOWEIE	CMPOFIE	THLEIE	MEIE	DEIE

	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset	-	-	-	-	-	GSLPR	GMDC[1:0]	

Bit	Symbol	Bit name	Function	R/W
b[31:18]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[17]	TSWR	Timestamp Write	1'b0: Timestamp write disabled 1'b1: Timestamp write enabled	R/W
b[16]	TSRST	Timestamp Reset	1'b0: Timestamp not reset 1'b1: Timestamp reset	R/W
b[15]	MOWEIE	GW FIFO Message overwrite Error Interrupt Enable	1'b0: GW FIFO Message overwrite Error Interrupt Disabled 1'b1: GW FIFO Message overwrite Error Interrupt Enabled	R/W
b[14]	QMEIE	TXQ Message lost Error Interrupt Enable	1'b0: TXQ Message Lost Error Interrupt Disabled 1'b1: TXQ Message Lost Error Interrupt Enabled	R/W
b[13]	-	Reserved	This bit is read as 0b. The write value should be always 0b.	R/W
b[12]	QOWEIE	TXQ Message overwrite Error Interrupt Enable	1'b0: TXQ Message overwrite Error Interrupt Disabled 1'b1: TXQ Message overwrite Error Interrupt Enabled	R/W
b[11]	CMPOFIE	CAN-FD message payload overflow Flag Interrupt enable	1'b0: CAN-FD message payload overflow Flag Interrupt Disabled 1'b1: CAN-FD message payload overflow Flag Interrupt Enabled	R/W
b[10]	THLEIE	TX History List Entry Lost Interrupt Enable	1'b0: TX History List Entry Lost Interrupt Disabled 1'b1: TX History List Entry Lost Interrupt Enabled	R/W
b[9]	MEIE	Message lost Error Interrupt Enable	1'b0: Message Lost Error Interrupt Disabled 1'b1: Message Lost Error Interrupt Enabled	R/W
b[8]	DEIE	DLC check Interrupt Enable	1'b0: DLC check Interrupt Disabled 1'b1: DLC check Interrupt Enabled	R/W
b[7:3]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[2]	GSLPR	Global Sleep Request	1'b0: Global Sleep Request Disabled 1'b1: Global Sleep Request Enabled	R/W
b[1:0]	GMDC	Global Mode Control	2'b00: Global Operation Mode Request 2'b01: Global Reset Mode Request 2'b10: Global Halt Mode Request 2'b11: Keep Current Value	R/W

The Global Control Register is used to control the global mode of the RS-CAN-FD module and to control the timestamp function. It is also used to enable and disable the global error interrupts

4.3.12.1 CFDGCTR.GMDC

Global Mode Control

CFDGCTR.GMDC bits can be used to configure the modes for the RS-CAN-FD module.

Additionally, if **CFDGCTR.GSLPR** is 1'b1 when the RS-CAN-FD module is in Reset Mode, then the RS-CAN-FD module transits to Global Sleep Mode.

Setting **CFDGCTR.GMDC** to 2'b11 has no effect. Mode transition is described in detail later in Section 5.2. Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

4.3.12.2 CFDGCTR.GSLPR

Global Sleep Request

This bit globally selects the Sleep request for RS-CAN-FD module including all CAN channels (Channel Sleep request is set automatically for all channels).

Users can only write to this bit when RS-CAN-FD module is in GL_RESET or GL_SLEEP mode.

4.3.12.3 CFDGCTR.DEIE

DLC check Interrupt Enable

If this bit is 1'b1, then an interrupt will be generated when a DLC error is detected in received frames.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

4.3.12.4 CFDGCTR.MEIE

Message lost Error Interrupt Enable

If this bit is 1'b1, then an interrupt will be generated when a Message Lost condition occurs.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

4.3.12.5 CFDGCTR.THLEIE

TX History List Entry Lost Interrupt Enable

If this bit is 1'b1, then an interrupt will be generated when a TX History List Entry Lost condition occurs.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

4.3.12.6 CFDGCTR.CMPOFIE

CAN-FD message payload overflow Flag Interrupt enable

If this bit is 1'b1, then an interrupt will be generated when a CAN-FD message payload overflow condition occurs.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

4.3.12.7 CFDGCTR.QOWEIE

TXQ Message overwrite Error Interrupt Enable

If this bit is 1'b1, then an interrupt will be generated when TXQ Message overwrite condition occurs.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

4.3.12.8 CFDGCTR.QMEIE

TXQ Message lost Error Interrupt Enable

If this bit is 1'b1, then an interrupt will be generated when a TXQ Message Lost condition occurs.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

4.3.12.9 CFDGCTR.MOWEIE

GW FIFO Message overwrite Error Interrupt Enable

If this bit is 1'b1, then an Interrupt will be generated when GW mode and GW FIFO Message over write condition occurs.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

4.3.12.10 CFDGCTR.TSRST

Timestamp Reset

When this bit is 1'b1, the Global Timestamp Register is reset to 16'h0000.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

Users should not write to this bit when the RS-CAN-FD module is in GL_RESET mode.

Read value is always 0.

This bit is cleared automatically by the RS-CAN-FD logic.

4.3.12.11 CFDGCTR.TSWR

Timestamp Write

If this bit is 1'b1, then data can be written into the Timestamp register.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP or GL_RESET mode.

4.3.13 CFDGSTS

Global Status Register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
Value after reset	-	-	-	-	-	-	-	-
	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset	-	-	-	-	-	-	-	-
	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
Value after reset	-	-	-	-	-	-	-	-
	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset	-	-	-	-	GRAMINIT	GSLPSTS	GHLTSTS	GRSTSTS
	0	0	0	0	1	1	0	1

Bit	Symbol	Bit name	Function	R/W
b[31:4]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[3]	GRAMINIT	Global RAM Initialisation	1'b0: RAM initialisation is finished 1'b1: RAM initialisation ongoing	R
b[2]	GSLPSTS	Global Sleep Status	1'b0: Not in Sleep Mode 1'b1: In Sleep Mode	R
b[1]	GHLTSTS	Global Halt Status	1'b0: Not in Halt Mode 1'b1: In Halt Mode	R
b[0]	GRSTSTS	Global Reset Status	1'b0: Not in Reset Mode 1'b1: In Reset Mode	R

The Global Status Register indicates the global status of the RS-CAN-FD module.

4.3.13.1 CFDGSTS.GRSTSTS

Global Reset Status

Indicates the Global RS-CAN-FD Module Reset mode.

This bit is set automatically when the RS-CAN-FD module enters Global Reset Mode. When the mode is changed from Global Reset Mode to Global Sleep Mode, **CFDGSTS.GRSTSTS** remains set.

This bit is cleared automatically when the RS-CAN-FD module exits the Global Reset Mode.

4.3.13.2 CFDGSTS.GHLTSTS

Global Halt Status

Indicates the Global RS-CAN-FD Module Halt mode.

This bit is set automatically when the RS-CAN-FD module enters Global Halt Mode.

This bit is cleared automatically when the RS-CAN-FD module exits the Halt Mode.

4.3.13.3 CFDGSTS.GSLPSTS

Global Sleep Status

Indicates the Global RS-CAN-FD Module Sleep mode.

This bit is set automatically when the RS-CAN-FD module enters Global Sleep Mode.

This bit is cleared automatically when the RS-CAN-FD module exits the Sleep Mode.

4.3.13.4 CFDGSTS.GRAMINIT

Global RAM Initialisation

Indicates the Global RS-CAN-FD Module RAM initialisation state.

This bit is set automatically when the RS-CAN-FD module enters Global Sleep Mode after HW Reset.

This bit is cleared automatically when the RS-CAN-FD module has finished the RAM initialisation.

This bit is cleared when the test_mode input port is set to 1'b1.

4.3.14 CFDGERFL

Global Error Flag Register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
Value after reset	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
Value after reset	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:24]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[23]	EEF7	ECC Error Flag for Channel 7	1'b0: ECC Error not detected during TX-SCAN 1'b1: ECC Error detected during TX-SCAN	R/W
b[22]	EEF6	ECC Error Flag for Channel 6	1'b0: ECC Error not detected during TX-SCAN 1'b1: ECC Error detected during TX-SCAN	R/W
b[21]	EEF5	ECC Error Flag for Channel 5	1'b0: ECC Error not detected during TX-SCAN 1'b1: ECC Error detected during TX-SCAN	R/W
b[20]	EEF4	ECC Error Flag for Channel 4	1'b0: ECC Error not detected during TX-SCAN 1'b1: ECC Error detected during TX-SCAN	R/W
b[19]	EEF3	ECC Error Flag for Channel 3	1'b0: ECC Error not detected during TX-SCAN 1'b1: ECC Error detected during TX-SCAN	R/W
b[18]	EEF2	ECC Error Flag for Channel 2	1'b0: ECC Error not detected during TX-SCAN 1'b1: ECC Error detected during TX-SCAN	R/W
b[17]	EEF1	ECC Error Flag for Channel 1	1'b0: ECC Error not detected during TX-SCAN 1'b1: ECC Error detected during TX-SCAN	R/W
b[16]	EEF0	ECC Error Flag for Channel 0	1'b0: ECC Error not detected during TX-SCAN 1'b1: ECC Error detected during TX-SCAN	R/W
b[15]	RXSFAIL7	RX Scan Fail of Channel 7	1'b0: RX Scan fail not detected 1'b1: RX Scan fail detected	R/W
b[14]	RXSFAIL6	RX Scan Fail of Channel 6	1'b0: RX Scan fail not detected 1'b1: RX Scan fail detected	R/W
b[13]	RXSFAIL5	RX Scan Fail of Channel 5	1'b0: RX Scan fail not detected 1'b1: RX Scan fail detected	R/W
b[12]	RXSFAIL4	RX Scan Fail of Channel 4	1'b0: RX Scan fail not detected 1'b1: RX Scan fail detected	R/W
b[11]	RXSFAIL3	RX Scan Fail of Channel 3	1'b0: RX Scan fail not detected 1'b1: RX Scan fail detected	R/W
b[10]	RXSFAIL2	RX Scan Fail of Channel 2	1'b0: RX Scan fail not detected 1'b1: RX Scan fail detected	R/W
b[9]	RXSFAIL1	RX Scan Fail of Channel 1	1'b0: RX Scan fail not detected 1'b1: RX Scan fail detected	R/W

b[8]	RXSFAIL0	<i>RX Scan Fail of Channel 0</i>	1'b0: RX Scan fail not detected 1'b1: RX Scan fail detected	R/W
b[7]	MOWES	Message overwrite Error Status	1'b0: Message overwrite Error not detected 1'b1: Message overwrite Error detected	R
b[6]	QMES	TXQ Message Lost Error Status	1'b0: TXQ Message lost Error not detected 1'b1: TXQ Message lost Error detected	R
b[5]	OTBMLTSTS	<i>OTB FIFO Message Lost Status</i>	1'b0: Message lost Error not detected 1'b1: Message lost Error detected	R
b[4]	QOWES	TXQ Message overwrite Error Status	1'b0: TXQ Message overwrite Error not detected 1'b1: TXQ Message overwrite Error detected	R
b[3]	CMPOF	CAN-FD message payload overflow Flag	1'b0: CAN-FD message payload overflow not detected 1'b1: CAN-FD message payload overflow detected	R/W
b[2]	THLES	TX History List Entry Lost Error Status	1'b0: TX History List Entry Lost Error not detected 1'b1: TX History List Entry Lost Error detected	R
b[1]	MES	Message Lost Error Status	1'b0: Message lost Error not detected 1'b1: Message lost Error detected	R
b[0]	DEF	DLC Error Flag	1'b0: DLC Error not detected 1'b1: DLC Error detected	R/W

The Global Error Flag register indicates the detection of global errors.

4.3.14.1 CFDGERFL.DEF

DLC Error Flag

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP or GL_RESET mode. Writing 1'b1 has no influence on the bit values.

Do not use bit clear instruction for clearing this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Then the other bits remain set to 1'b1.

This bit is set automatically when a DLC Error is detected in a received frame.

The bit is cleared by writing 1'b0 to it.

This bit will be cleared automatically in GL_RESET mode.

4.3.14.2 CFDGERFL.MES

Message Lost Error Status

This bit is set automatically when a FIFO Message Lost Error is detected.

This bit is cleared automatically when all FIFO Message Lost flags are cleared.

This bit will be cleared automatically in GL_RESET mode.

This bit is a reserve bit when **CFDGFFIMC.FFIEN** is 1'b1 and **CFDGVMEIS.FMLT** is 1'b1.

When reserve bit, this bit is always read as 0.

4.3.14.3 CFDGERFL.THLES

TX History List Entry Lost Error Status

This bit is set automatically when a TX History List Entry Lost Error is detected.

This bit is cleared automatically when all TX History List Entry Lost flags are cleared.

This bit will be cleared automatically in GL_RESET mode.

This bit is a reserve bit when **CFDGFFIMC.FFIEN** is 1'b1 and **CFDGVMEIS.THLELT** is 1'b1.

When reserve bit, this bit is always read as 0.

4.3.14.4 CFDGERFL.CMPOF

CAN-FD message payload overflow Flag

If this bit is set then a payload overflow has happened on at least one channel.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP or GL_RESET mode.

Writing 1'b1 has no influence on the bit values.

This bit is set automatically when a CAN-FD message payload overflow is detected on at least one channel.

The bit is cleared by writing 1'b0 to it.

Do not use bit clear instruction for clearing this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Then the other bits remain set to 1'b1.

This bit will be cleared automatically in GL_RESET mode.

4.3.14.5 CFDGERFL.QOWES

TXQ Message overwrite Error Status

This bit is set automatically when TXQ Message overwrite Error is detected.

This bit is cleared automatically when all TXQ Message overwrite flags are cleared.

This bit will be cleared automatically in GL_RESET mode.

This bit is a reserve bit when **CFDGFFIMC.FFIEN** is 1'b1 and **CFDGVMEIS.TXQOW** is 1'b1.

When reserve bit, this bit is always read as 0.

4.3.14.6 CFDGERFL.OTBMLTSTS

OTB FIFO Message Lost Status

This bit is set automatically when Message is lost due to attempted storage of a new message when FIFO is already full.

The bit will be cleared when RS-CAN-FD module enters GL_RESET mode, or the OTBMLT bit in the Global OTB FIFO Configuration / Status Register is cleared (CFDGLOTB.OTBMLT = 1'b0).

This bit will be cleared automatically in GL_RESET mode.

4.3.14.7 CFDGERFL.QMES

TXQ Message lost Error Status

This bit is set automatically when TXQ Message lost Error is detected.

This bit is cleared automatically when all TXQ Message lost flags are cleared.

This bit will be cleared automatically in GL_RESET mode.

This bit is a reserve bit when **CFDGFFIMC.FFIEN** is 1'b1 and **CFDGVMEIS.TXQMLT** is 1'b1.

When reserve bit, this bit is always read as 0

4.3.14.8 CFDGERFL.MOWES

Message overwrite Error Status

This bit is set automatically when GW mode and COMFIFO Message overwrite Error is detected.

This bit is cleared automatically when all COMFIFO Message overwrite flags are cleared.

This bit will be cleared automatically in GL_RESET mode.

This bit is a reserve bit when **CFDGFFIMC.FFIEN** is 1'b1 and **CFDGVMEIS.CFMOW** is 1'b1.

When reserve bit, this bit is always read as 0

4.3.14.9 CFDGERFL.RXSFAIL_n

RX Scan Fail of Channel n

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP or GL_RESET mode.

Writing 1'b1 has no influence on the bit values.

Do not use bit clear instruction for clearing this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Then the other bits remain set to 1'b1.

This bit is set automatically when Message is lost due to unprocessed message on the receiving channel.

The bit is cleared by writing 1'b0 to it.

This bit will be cleared automatically in GL_RESET mode.

4.3.14.10 CFDGERFL.EEFn

ECC Error Flag for Channel n

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP or GL_RESET mode.

Writing 1'b1 has no influence on the bit values.

Do not use bit clear instruction for clearing this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Then the other bits remain set to 1'b1.

The bit is cleared by writing 1'b0 to it.

This bit will be cleared automatically in GL_RESET mode.

4.3.15 CFDGTINTSTSv

Global TX Interrupt Status Register v

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
	-	CFOTIF(3 + (v * 4))	TQOEIF(3 + (v * 4))	THIF(3 + (v * 4))	CFTIF(3 + (v * 4))	TQIF(3 + (v * 4))	TAIF(3 + (v * 4))	TSIF(3 + (v * 4))
Value after reset	0	0	0	0	0	0	0	0

	b23	b22	b21	b20	b19	b18	b17	b16
	-	CFOTIF(2 + (v * 4))	TQOEIF(2 + (v * 4))	THIF(2 + (v * 4))	CFTIF(2 + (v * 4))	TQIF(2 + (v * 4))	TAIF(2 + (v * 4))	TSIF(2 + (v * 4))
Value after reset	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8
	-	CFOTIF(1 + (v * 4))	TQOEIF(1 + (v * 4))	THIF(1 + (v * 4))	CFTIF(1 + (v * 4))	TQIF(1 + (v * 4))	TAIF(1 + (v * 4))	TSIF(1 + (v * 4))
Value after reset	0	0	0	0	0	0	0	0

	b7	b6	b5	b4	b3	b2	b1	b0
	-	CFOTIF(0 + (v * 4))	TQOEIF(0 + (v * 4))	THIF(0 + (v * 4))	CFTIF(0 + (v * 4))	TQIF(0 + (v * 4))	TAIF(0 + (v * 4))	TSIF(0 + (v * 4))
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31]	-	Reserved	This bit is read as 0b. The write value should be always 0b.	R/W
b[30]	CFOTIF (3 + (v * 4))	COM FIFO One Frame Transmission Interrupt Flag Channel (3 + (v * 4))	1'b0: Channel n COM FIFO One Frame Transmission Interrupt flag not set 1'b1: Channel n COM FIFO One Frame Transmission Interrupt flag set	R
b[29]	TQOEIF (3 + (v * 4))	TX Queue One Frame Transmission Interrupt Flag Channel (3 + (v * 4))	1'b0: Channel n TX Queue One Frame Transmission Interrupt flag not set 1'b1: Channel n TX Queue One Frame Transmission Interrupt flag set	R
b[28]	THIF (3 + (v * 4))	TX History List Interrupt Channel (3 + (v * 4))	1'b0: Channel n TX History List Interrupt flag not set 1'b1: Channel n TX History List Interrupt flag set	R
b[27]	CFTIF (3 + (v * 4))	COM FIFO TX/GW Mode Interrupt Flag Channel (3 + (v * 4))	1'b0: Channel n COM FIFO TX/GW mode Interrupt flag not set 1'b1: Channel n COM FIFO TX/GW mode Interrupt flag set	R
b[26]	TQIF (3 + (v * 4))	TX Queue Interrupt Flag Channel (3 + (v * 4))	1'b0: Channel n TX Queue Interrupt flag not set 1'b1: Channel n TX Queue Interrupt flag set	R
b[25]	TAIF (3 + (v * 4))	TX Abort Interrupt Flag Channel (3 + (v * 4))	1'b0: Channel n TX abort Interrupt flag not set 1'b1: Channel n TX abort Interrupt flag set	R
b[24]	TSIF (3 + (v * 4))	TX Successful Interrupt Flag Channel (3 + (v * 4))	1'b0: Channel n TX Successful completion Interrupt flag not set 1'b1: Channel n TX Successful completion Interrupt flag set	R
b[23]	-	Reserved	This bit is read as 0b. The write value should be always 0b.	R/W
b[22]	CFOTIF (2 + (v * 4))	COM FIFO One Frame Transmission Interrupt Flag Channel (2 + (v * 4))	1'b0: Channel n COM FIFO One Frame Transmission Interrupt flag not set 1'b1: Channel n COM FIFO One Frame Transmission Interrupt flag set	R
b[21]	TQOEIF (2 + (v * 4))	TX Queue One Frame Transmission Interrupt Flag Channel (2 + (v * 4))	1'b0: Channel n TX Queue One Frame Transmission Interrupt flag not set 1'b1: Channel n TX Queue One Frame Transmission Interrupt flag set	R
b[20]	THIF (2 + (v * 4))	TX History List Interrupt Channel (2 + (v * 4))	1'b0: Channel n TX History List Interrupt flag not set 1'b1: Channel n TX History List Interrupt flag set	R
b[19]	CFTIF (2 + (v * 4))	COM FIFO TX/GW Mode Interrupt Flag Channel (2 + (v * 4))	1'b0: Channel n COM FIFO TX/GW mode Interrupt flag not set 1'b1: Channel n COM FIFO TX/GW mode Interrupt flag set	R
b[18]	TQIF (2 + (v * 4))	TX Queue Interrupt Flag Channel (2 + (v * 4))	1'b0: Channel n TX Queue Interrupt flag not set 1'b1: Channel n TX Queue Interrupt flag set	R
b[17]			1'b0: Channel n TX abort Interrupt flag not set	R

	TAIF (2 + (v * 4))	TX Abort Interrupt Flag Channel (2 + (v * 4))	1'b1: Channel n TX abort Interrupt flag set	
b[16]	TSIF (2 + (v * 4))	TX Successful Interrupt Flag Channel (2 + (v * 4))	1'b0: Channel n TX Successful completion Interrupt flag not set 1'b1: Channel n TX Successful completion Interrupt flag set	R
b[15]	-	Reserved	This bit is read as 0b. The write value should be always 0b.	R/W
b[14]	CFOTIF (1 + (v * 4))	COM FIFO One Frame Transmission Interrupt Flag Channel (1 + (v * 4))	1'b0: Channel n COM FIFO One Frame Transmission Interrupt flag not set 1'b1: Channel n COM FIFO One Frame Transmission Interrupt flag set	R
b[13]	TQOFIF (1 + (v * 4))	TX Queue One Frame Transmission Interrupt Flag Channel (1 + (v * 4))	1'b0: Channel n TX Queue One Frame Transmission Interrupt flag not set 1'b1: Channel n TX Queue One Frame Transmission Interrupt flag set	R
b[12]	THIF (1 + (v * 4))	TX History List Interrupt Channel (1 + (v * 4))	1'b0: Channel n TX History List Interrupt flag not set 1'b1: Channel n TX History List Interrupt flag set	R
b[11]	CFTIF (1 + (v * 4))	COM FIFO TX/GW Mode Interrupt Flag Channel (1 + (v * 4))	1'b0: Channel n COM FIFO TX/GW mode Interrupt flag not set 1'b1: Channel n COM FIFO TX/GW mode Interrupt flag set	R
b[10]	TQIF (1 + (v * 4))	TX Queue Interrupt Flag Channel (1 + (v * 4))	1'b0: Channel n TX Queue Interrupt flag not set 1'b1: Channel n TX Queue Interrupt flag set	R
b[9]	TAIF (1 + (v * 4))	TX Abort Interrupt Flag Channel (1 + (v * 4))	1'b0: Channel n TX abort Interrupt flag not set 1'b1: Channel n TX abort Interrupt flag set	R
b[8]	TSIF (1 + (v * 4))	TX Successful Interrupt Flag Channel (1 + (v * 4))	1'b0: Channel n TX Successful completion Interrupt flag not set 1'b1: Channel n TX Successful completion Interrupt flag set	R
b[7]	-	Reserved	This bit is read as 0b. The write value should be always 0b.	R/W
b[6]	CFOTIF (0 + (v * 4))	COM FIFO One Frame Transmission Interrupt Flag Channel (0 + (v * 4))	1'b0: Channel n COM FIFO One Frame Transmission Interrupt flag not set 1'b1: Channel n COM FIFO One Frame Transmission Interrupt flag set	R
b[5]	TQOFIF (0 + (v * 4))	TX Queue One Frame Transmission Interrupt Flag Channel (0 + (v * 4))	1'b0: Channel n TX Queue One Frame Transmission Interrupt flag not set 1'b1: Channel n TX Queue One Frame Transmission Interrupt flag set	R
b[4]	THIF (0 + (v * 4))	TX History List Interrupt Channel (0 + (v * 4))	1'b0: Channel n TX History List Interrupt flag not set 1'b1: Channel n TX History List Interrupt flag set	R
b[3]	CFTIF (0 + (v * 4))	COM FIFO TX/GW Mode Interrupt Flag Channel (0 + (v * 4))	1'b0: Channel n COM FIFO TX/GW mode Interrupt flag not set 1'b1: Channel n COM FIFO TX/GW mode Interrupt flag set	R
b[2]	TQIF (0 + (v * 4))	TX Queue Interrupt Flag Channel (0 + (v * 4))	1'b0: Channel n TX Queue Interrupt flag not set 1'b1: Channel n TX Queue Interrupt flag set	R
b[1]	TAIF (0 + (v * 4))	TX Abort Interrupt Flag Channel (0 + (v * 4))	1'b0: Channel n TX abort Interrupt flag not set 1'b1: Channel n TX abort Interrupt flag set	R
b[0]	TSIF (0 + (v * 4))	TX Successful Interrupt Flag Channel (0 + (v * 4))	1'b0: Channel n TX Successful completion Interrupt flag not set 1'b1: Channel n TX Successful completion Interrupt flag set	R

The Global TX Interrupt Status register v indicates the detection of transmit specific interrupts.

(no_of_channels = 8)

(no_of_INTFs_per_channel = 7)

(no_of_INTFs = 8 * 7 = 56)

(no_of_INTFs_per_CFDGTINTSTS = 28)

(no_of_CFDGTINTSTSs = ceil(no_of_INTFs / no_of_INTFs_per_CFDGTINTSTS) = ceil(56 / 28) = 2)

v = [0...no_of_CFDGTINTSTSs-1]

n = [0...no_of_channels-1]

(v) can be calculated from the desired Interrupt Channel (n) status flags using the formula **v = floor(n / 4)**

Byte position can be calculated using the formula ($n - (v * 4)$)

e.g. for Interrupt Channel 7 Status Flags we have:

$n = 7, v = \text{floor}(7 / 4) = 1$ and Byte position = $(7 - (1 * 4)) = 3$

Therefore, **Users should read the Byte 3 (4th Byte) of CFDGTINTSTS1.**

4.3.15.1 CFDGTINTSTSv.TSIFn

TX Successful Interrupt Flag Channel ($n - (v * 4)$)

CFDGTINTSTSv.TSIFn Bit (TX Successful Completion Interrupt Flag Channel n)

This bit is set automatically when the Transmission Successful flag of the related channel is set when the Interrupt is enabled.

This bit is cleared automatically when related TX MB Result status bits are cleared or the Interrupt enable is disabled.

This bit will be cleared automatically in GL_RESET mode or CH_RESET mode.

4.3.15.2 CFDGTINTSTSv.TAIFn

TX Abort Interrupt Flag Channel ($n - (v * 4)$)

CFDGTINTSTSv.TAIFn Bit (TX Abort Interrupt Flag Channel n)

This bit is set automatically when abort Successful flag of the related channel is set when the Interrupt is enabled.

This bit is cleared automatically when the related TX MB Result status bits are cleared or the Interrupt enable is disabled.

This bit will be cleared automatically in GL_RESET mode or CH_RESET mode.

4.3.15.3 CFDGTINTSTSv.TQIFn

TX Queue Interrupt Flag Channel ($n - (v * 4)$)

CFDGTINTSTSv.TQIFn Bit (TX Queue Interrupt Flag Channel n)

This bit is set automatically when the related TX Queue Interrupt flag of the related channel is set when the Interrupt is enabled.

This bit is cleared automatically when related TX Queue Interrupt flag is cleared or the Interrupt enable is disabled.

This bit will be cleared automatically in GL_RESET mode or CH_RESET mode.

This bit is a reserve bit when CFDGFFIMC.FFIEN is 1'b1.

When reserve bit, this bit is always read as 0.

4.3.15.4 CFDGTINTSTSv.CFTIFn

COM FIFO TX/GW Mode Interrupt Flag Channel ($n - (v * 4)$)

CFDGTINTSTSv.CFTIFn Bit (COM FIFO in TX/GW mode Interrupt flag Channel n)

This bit is set automatically when the related Common TX/GW FIFO Interrupt Flag (**CFDCFSTSd.CFTXIF**) is set when the Interrupt is enabled.

This bit is cleared automatically when related Common TX/GW FIFO Interrupt Flag is cleared or the Interrupt enable is disabled.

This bit will be cleared automatically in GL_RESET mode or CH_RESET mode.

This bit is a reserve bit when CFDGFFIMC.FFIEN is 1'b1.

When reserve bit, this bit is always read as 0.

4.3.15.5 CFDGTINTSTSv.THIFn

TX History List Interrupt Channel ($n - (v * 4)$)

CFDGTINTSTSv.THIFn Bit (TX History List Interrupt flag Channel n)

This bit is set automatically when the related TX History List Interrupt Flag (**CFDTHLSTS_n.THLIF**) is set when the Interrupt is enabled.

This bit is cleared automatically when related TX History List Interrupt Flag (**CFDTHLSTS_n.THLIF**) is cleared or the Interrupt enable is disabled.

This bit will be cleared automatically in GL_RESET mode or CH_RESET mode.

This bit is a reserve bit when CFDGFFIMC.FFIEN is 1'b1.

When reserve bit, this bit is always read as 0.

4.3.15.6 CFDGTINTSTSv.TQOFIFn

TX Queue One Frame Transmission Interrupt Flag Channel (n - (v * 4))

CFDGTINTSTSv.TQOFIFn Bit (TX Queue One Frame Transmission Interrupt Flag Channel n)

This bit is set automatically when the related TX Queue One Frame TX Interrupt flag of the related channel is set when the Interrupt is enabled.

This bit is cleared automatically when related TX Queue One Frame TX Interrupt flag is cleared or the Interrupt enable is disabled.

This bit will be cleared automatically in GL_RESET mode or CH_RESET mode.

This bit is a reserve bit when CFDGFFIMC.FFIEN is 1'b1.

When reserve bit, this bit is always read as 0.

4.3.15.7 CFDGTINTSTSv.CFOTIFn

COM FIFO One Frame Transmission Interrupt Flag Channel (n - (v * 4))

CFDGTINTSTSv.CFOTIFn Bit (COM FIFO One Frame Transmission Interrupt flag Channel n)

This bit is set automatically when the related Common FIFO One Frame Transmission Interrupt Flag (**CFDCFSTS_d.CFOFTXIF**) is set when the Interrupt is enabled.

This bit is cleared automatically when related Common FIFO One Frame Transmission Interrupt Flag (**CFDCFSTS_d.CFOFTXIF**) is cleared or the Interrupt enable is disabled.

This bit will be cleared automatically in GL_RESET mode or CH_RESET mode.

This bit is a reserve bit when CFDGFFIMC.FFIEN is 1'b1.

When reserve bit, this bit is always read as 0.

4.3.16 CFDGTSC

Global Timestamp Counter Register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24	
Value after reset	-	-	-	-	-	-	-	-	
	0	0	0	0	0	0	0	0	
	b23	b22	b21	b20	b19	b18	b17	b16	
Value after reset	-	-	-	-	-	-	-	-	
	0	0	0	0	0	0	0	0	
	b15	b14	b13	b12	b11	b10	b9	b8	
Value after reset	-	-	-	-	-	-	-	-	
	0	0	0	0	0	0	0	0	
	b7	b6	b5	b4	b3	b2	b1	b0	
Value after reset	-	-	-	-	-	-	-	-	
	0	0	0	0	0	0	0	0	
Bit	Symbol	Bit name	Function						R/W
b[31:16]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[15:0]	TS	Timestamp Value	Timestamp Value						R

The Timestamp counter register stores the Timestamp based on the selected configuration.

4.3.16.1 CFDGTSC.TS

Timestamp Value

The proper incrementing of the Time Stamp counter cannot be guaranteed when moving to Halt state.

The Timestamp value is stored in this register based on the configuration of TSSS, TSBTCS and TSP.

Users cannot write to these bits when the RS-CAN-FD module is in GL_RESET or GL_SLEEP mode.

Users should only write to the bits TS[15:0] when 'Timestamp Write' is enabled (CFDGCTR.TSWR bit is 1'b1) and RS-CAN-FD module is in GL_HALT mode.

These bits will be cleared automatically in GL_RESET mode.

4.3.17 CFDGAFLECTR

Global Acceptance Filter List Entry Control Register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24	
	-	-	-	-	-	-	-	-	
Value after reset	0	0	0	0	0	0	0	0	
	b23	b22	b21	b20	b19	b18	b17	b16	
	-	-	-	-	-	-	-	-	
Value after reset	0	0	0	0	0	0	0	0	
	b15	b14	b13	b12	b11	b10	b9	b8	
	-	-	-	-	-	-	-	AFLDAE	
Value after reset	0	0	0	0	0	0	0	0	
	b7	b6	b5	b4	b3	b2	b1	b0	
	-				AFLPN[6:0]				
Value after reset	0	0	0	0	0	0	0	0	
Bit	Symbol	Bit name	Function						R/W
b[31:9]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[8]	AFLDAE	Acceptance Filter List Data Access Enable	1'b0: Acceptance Filter List Data access disabled 1'b1: Acceptance Filter List Data access enabled						R/W
b[7]	-	Reserved	This bit is read as 0b. The write value should be always 0b.						R/W
b[6:0]	AFLPN	Acceptance Filter List Page Number	Select an Acceptance Filter List Page						R/W

This register is used to select the Global Acceptance Filter List page for reading or writing entries into the Global Acceptance Filter List.

4.3.17.1 CFDGAFLECTR.AFLPN

Acceptance Filter List Page Number

These bits select the Page Number to access the desired RAM area of the Acceptance Filter List. One Acceptance Filter List page consists of 16 Acceptance Filter List entries.

Read/Write accesses to the Acceptance Filter List can only be performed via a fixed window.

Users cannot write to this bit when RS-CAN-FD module is in GL_SLEEP mode.

Users should enter, only the values between 0 and 95 (7'h5F) inclusive.

4.3.17.2 CFDGAFLECTR.AFLDAE

Acceptance Filter List Data Access Enable

This bit prevents Acceptance Filter List write access if cleared after configuration of the Acceptance Filter List.

Users can read data from Acceptance Filter List independent of the status of this bit.

Users cannot write to this bit when RS-CAN-FD module is in GL_SLEEP mode.

This bit should be set to enable write access to Acceptance Filter List.

4.3.18 CFDGAFLCFGw

Global Acceptance Filter List Configuration Register w

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24	
	-	-	-	-	-	-	-	RNC((0)+(w*2))[8]	
Value after reset	0	0	0	0	0	0	0	0	
	b23	b22	b21	b20	b19	b18	b17	b16	
	RNC((0)+(w*2))[7:0]								
Value after reset	0	0	0	0	0	0	0	0	
	b15	b14	b13	b12	b11	b10	b9	b8	
	-	-	-	-	-	-	-	RNC((1)+(w*2))[8]	
Value after reset	0	0	0	0	0	0	0	0	
	b7	b6	b5	b4	b3	b2	b1	b0	
	RNC((1)+(w*2))[7:0]								
Value after reset	0	0	0	0	0	0	0	0	
Bit	Symbol	Bit name	Function						R/W

b[31:25]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[24:16]	RNC((0)+(w*2))	Rule Number for Channel ((0) + (w * 2))	Number of rules dedicated to channel ((0) + (w * 2))	R/W
b[15:9]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[8:0]	RNC((1)+(w*2))	Rule Number for Channel ((1) + (w * 2))	Number of rules dedicated to channel ((1) + (w * 2))	R/W

(no_of_channels = no_of_RNs = 8)

(no_of_RNs_per_CFDGAFLCFG = 2)

(no_of_CFDGAFLCFGs = ceil(no_of_RNs / no_of_INTFs_per_CFDGAFLCFG) = ceil(8 / 2) = 4)

w = [0...no_of_CFDGAFLCFGs-1]

n = [0...no_of_channels-1]

(w) can be calculated from the desired Channel (n) Rule Number using the formula **w = floor(n / 2)**

Word position can be calculated using the formula **((1 - n) + (w * 2))**

e.g. for Channel 7 Rule Number we have:

n = 7, w = floor(7 / 2) = 3 and Word position = ((1 - 7) + (3 * 2)) = 0

Therefore, **Users should read the Word 0 (First Word) of CFDGAFLCFG3.**

This register is used to define the number of Acceptance Filter List Entries (called "Rules") applicable for channels 0 to 7 in the Acceptance Filter List.

The total number of available entries in the Acceptance Filter List is 192 * (n+1) (e.g. 1536 for 8 CAN channels). However, the filters can be allocated flexibly to the different channels depending on requirements as long as both of the following conditions are satisfied:

the maximum number of Acceptance Filter per channel is 384 and

the total number of rules defined for all channels is not exceeding the number of available entries in the Acceptance Filter List.

4.3.18.1 CFDGAFLCFGw.RNC((1-n)+(w*2))

Rule Number for Channel ((1 – n) + (w * 2))

These bits define the number of rules in the Acceptance Filter List for channel n.

Users can only write to this bit when RS-CAN-FD module is in GL_RESET mode.

4.3.19 CFDGAFLIDr

Global Acceptance Filter List ID Registers r = [1...10]h

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24	
	GAFLIDE	GAFLRTR	GAFLLB	GAFLID[28:24]					
Value after reset	0	0	0	0	0	0	0	0	
	b23	b22	b21	b20	b19	b18	b17	b16	
Value after reset	0	0	0	0	0	0	0	0	
	b15	b14	b13	b12	b11	b10	b9	b8	
Value after reset	0	0	0	0	0	0	0	0	
	b7	b6	b5	b4	b3	b2	b1	b0	
Value after reset	0	0	0	0	0	0	0	0	
Bit	Symbol	Bit name	Function						
b[31]	GAFLIDE	Global Acceptance Filter List Entry IDE Field	1'b0: Standard Identifier of Rule entry ID is valid for acceptance filtering 1'b1: Extended Identifier of Rule entry ID is valid for acceptance filtering						
b[30]	GAFLRTR	Global Acceptance Filter List Entry RTR Field	1'b0: Data Frame 1'b1: Remote Frame						
b[29]	GAFLLB	Global Acceptance Filter List Entry Loopback Configuration	1'b0: Global Acceptance Filter List entry ID for acceptance filtering has attribute 'RX' 1'b1: Global Acceptance Filter List entry ID for acceptance filtering has attribute 'TX'						
b[28:0]	GAFLID	Global Acceptance Filter List Entry ID Field	ID part of the Global Acceptance Filter List entry						

(no_of_CFDGAFLIDs = 16)

(r = [1...no_of_CFDGAFLIDs])

These registers are used to configure the ID field of the Rule Entries in the Global Acceptance Filter List.

4.3.19.1 CFDGAFLIDr.GAFLID

Global Acceptance Filter List Entry ID Field

These bits represent the CAN Identifier (ID) field of each of the Global Acceptance Filter List entry. Acceptance filter process compares this field against the ID of a received CAN message.

For alignment of these bits in standard and extended frame format, see Section 4.4

Users cannot write to these bits when **CFDGAFLECTR.AFLDAE** bit is 1'b0.

Users should only write to these bits when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

4.3.19.2 CFDGAFLIDr.GAFLLB

Global Acceptance Filter List Entry Loopback Configuration

This bit selects if the Global Acceptance Filter List entry gets the attribute 'RX' or 'TX'. This attribute decides about the validity of the entry in the mirror mode case, loopback test mode case and during standard (non-loopback) reception. Please see Table 7.1 for detailed description of the validity of the Global Acceptance Filter List entry depending on transmitter/receiver case, type of loopback mode and RX/TX attribute.

Users cannot write to these bits when **CFDGAFLECTR.AFLDAE** bit is 1'b0.

Users should only write to these bits when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

4.3.19.3 CFDGAFLIDr.GAFLRTR

Global Acceptance Filter List Entry RTR Field

This bit allows the configuration of the specified frame format (Data Frame or Remote Frame) for each Global Acceptance Filter List entry. For each Rule entry in a CAN channel the Acceptance filter process compares this bit against the RTR bit of the received CAN message.

Users cannot write to these bits when **CFDGAFLECTR.AFLDAE** bit is 1'b0.

Users should only write to these bits when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

4.3.19.4 CFDGAFLIDr.GAFLIDE

Global Acceptance Filter List Entry IDE Field

This bit allows the configuration of the ID format (Standard ID or Extended ID) for each of the Global Acceptance Filter List entry. For each Rule entry of the related CAN channel the Acceptance filter process compares this bit against the IDE bit of the received CAN message.

Users cannot write to these bits when **CFDGAFLECTR.AFLDAE** bit is 1'b0.

Users should only write to these bits when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

4.3.20 CFDGAFLMr

Global Acceptance Filter List Mask Registers $r = [1...10]h$

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
	GAFLIDEM	GAFLRTRM	GAFLIFL1			GAFLIDM[28:24]		
Value after reset	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
					GAFLIDM[23:16]			
Value after reset	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
					GAFLIDM[15:8]			
Value after reset	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
					GAFLIDM[7:0]			
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31]	GAFLIDEM	Global Acceptance Filter List IDE Mask	1'b0: IDE bit is not considered for ID matching 1'b1: IDE bit is considered for ID matching	R/W
b[30]	GAFLRTRM	Global Acceptance Filter List Entry RTR Mask	1'b0: RTR bit is not considered for ID matching 1'b1: RTR bit is considered for ID matching	R/W
b[29]	GAFLIFL1	Global Acceptance Filter List Information Label 1	Global Acceptance Filter List Information Label bit1	R/W
b[28:0]	GAFLIDM	Global Acceptance Filter List ID Mask Field	Global Acceptance Filter List Mask field bits for ID field bits	R/W

The Global Rule Mask entry registers are used to configure the Mask field of each Rule Entries in the Global Acceptance Filter List.

4.3.20.1 CFDGAFLMr.GAFLIDM

Global Acceptance Filter List ID Mask Field

These bits are the filter mask bits for the related bits in the CAN Identifier field of each Global Acceptance Filter List entry.

1'b0	Corresponding STD-ID / EXT-ID bit is not considered for ID matching
1'b1	Corresponding STD-ID / EXT-ID bit is considered for ID matching

Users cannot write to these bits when **CFDGAFLECTR.AFLDAE** bit is 1'b0.

Users should only write to these bits when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

4.3.20.2 CFDGAFLMr.GAFLIFL1

Global Acceptance Filter List Information Label 1

These bits allow the configuration of a 2-bit Information label that will be attached to a received message accepted by the related Global Acceptance Filter List entry. This bit is a MSB bit of an information label.

Users cannot write to these bits when **CFDGAFLECTR.AFLDAE** bit is 1'b0.

Users should only write to these bits when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

This bit is stored in Information Label Field [1] (**CFDRMFSTS.RMIFL [1]**, **CFDRFFDSTS.RFIFL [1]**, **CFDCFFDCSTS.CFIFL [1]**) of the storage location of an incoming message.

This bit is stored in **CFDTHLACC1n.TIFL** [1] when **CFDTHLCCn.THLDGE=1** is set up using GW function.

4.3.20.3 CFDGAFLMr.GAFLRTRM

Global Acceptance Filter List Entry RTR Mask

This bit allows the configuration of the RTR mask bit for each Global Acceptance Filter List entry.

Users cannot write to these bits when **CFDGAFLECTR.AFLDAE** bit is 1'b0.

Users should only write to these bits when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

4.3.20.4 CFDGAFLMr.GAFLIDEM

Global Acceptance Filter List IDE Mask

This bit allows the configuration of the IDE mask bit for each Global Acceptance Filter List entry.

When IDE mask bit is 1'b0, then the ID comparison depends upon the received IDE bit.

If received IDE bit is 1'b0, then STD-ID comparison takes place.

If received IDE bit is 1'b1, then EXT-ID comparison takes place.

Users cannot write to these bits when **CFDGAFLECTR.AFLDAE** bit is 1'b0.

Users should only write to these bits when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

4.3.21 CFDGAFLP0r

Global Acceptance Filter List Pointer 0 Registers r = [1...10]h

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
GAFLPTR[15:8]								
Value after reset	0	0	0	0	0	0	0	0
GAFLPTR[7:0]								
Value after reset	0	0	0	0	0	0	0	0
GAFLRMDP[6:0]								
Value after reset	0	0	0	0	0	0	0	0
GAFLIFL0 GAFLSRD2 GAFLSRD1 GAFLSRD0 GAFLDLC[3:0]								
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:16]	GAFLPTR	Global Acceptance Filter List Pointer Field	Global Acceptance Filter List Pointer	R/W
b[15]	GAFLRMDV	Global Acceptance Filter List RX Message Buffer Valid	1'b0: Global Acceptance Filter List Single Message Buffer Direction Pointer is invalid 1'b1: Global Acceptance Filter List Single Message Buffer Direction Pointer is valid	R/W
b[14:8]	GAFLRMDP	Global Acceptance Filter List RX Message Buffer Direction Pointer	RX Message Buffer number for storage of received messages	R/W
b[7]	GAFLIFL0	Global Acceptance Filter List Information Label 0	Global Acceptance Filter List Information Label bit0	R/W
b[6]	GAFLSRD2	Global Acceptance Filter List Select Routing destination 2	1'b0: Routing target is CFIFO2 1'b1: Routing target is TX Queue 2 instead of CFIFO2	R/W
b[5]	GAFLSRD1	Global Acceptance Filter List Select Routing destination 1	1'b0: Routing target is CFIFO1 1'b1: Routing target is TX Queue 1 instead of CFIFO1	R/W
b[4]	GAFLSRD0	Global Acceptance Filter List Select Routing destination 0	1'b0: Routing target is CFIFO0 1'b1: Routing target is TX Queue 0 instead of CFIFO0	R/W
b[3:0]	GAFLDLC	Global Acceptance Filter List DLC Field	Minimum no. of Data Bytes in a Data Frame required for its acceptance	R/W

The Global Acceptance Filter List Pointer 0 registers are used to configure the DLC, SW Pointer, Single Message Buffer select and Message Buffer direction pointer for each Rule Entry in the Global Acceptance Filter List.

4.3.21.1 CFDGAFLP0r.GAFLDLC

Global Acceptance Filter List DLC Field

These bits allow the configuration of the minimum DLC (Data Length Code) value for a message to be accepted by the related Global Acceptance Filter List entry (automatic DLC filter function). DLC filter process is only passed if the DLC value of the message accepted by a Global Acceptance Filter List entry is equal or higher than the DLC value configured for this related Global Acceptance Filter List entry. Automatic DLC filter function is disabled for the corresponding Rule Entry when this field is set to 4'h0.

Following binary values can be configured:

Format	DLC[3]	DLC[2]	DLC[1]	DLC[0]	Description
--------	--------	--------	--------	--------	-------------

CAN and CAN-FD	0	0	0	0	DLC of received message = 0 or more (DLC Filter check is disabled)
CAN and CAN-FD	0	0	0	1	DLC of received message = 1 or more
CAN and CAN-FD	0	0	1	0	DLC of received message = 2 or more
CAN and CAN-FD	0	0	1	1	DLC of received message = 3 or more
CAN and CAN-FD	0	1	0	0	DLC of received message = 4 or more
CAN and CAN-FD	0	1	0	1	DLC of received message = 5 or more
CAN and CAN-FD	0	1	1	0	DLC of received message = 6 or more
CAN and CAN-FD	0	1	1	1	DLC of received message = 7 or more
CAN	1	x	x	x	DLC of received message = 8 or more
CAN-FD	1	0	0	0	DLC of received message = 8 or more
CAN-FD	1	0	0	1	DLC of received message = 12 or more
CAN-FD	1	0	1	0	DLC of received message = 16 or more
CAN-FD	1	0	1	1	DLC of received message = 20 or more
CAN-FD	1	1	0	0	DLC of received message = 24 or more
CAN-FD	1	1	0	1	DLC of received message = 32 or more
CAN-FD	1	1	1	0	DLC of received message = 48 or more
CAN-FD	1	1	1	1	DLC of received message = 64

Users cannot write to these bits when **CFDGAFLCTR.AFLDAE** bit is 1'b0.

Users should only write to these bits when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

4.3.21.2 CFDGAFLP0r. GAFLSRD0

Global Acceptance Filter List Select Routing destination 0

This bit changes a copy destination to CFIFO0 or TXQ0 by routing.

If this bit is set as 1, the preset value of **CFDGAFLP1r.GAFLFDP** will choose TX Queue.

If this bit is cleared to 0, the preset value of **CFDGAFLP1r.GAFLFDP** will choose CommonFIFO.

Users cannot write to these bits when **CFDGAFLCTR.AFLDAE** bit is 1'b0.

Users should only write to the bit when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

4.3.21.3 CFDGAFLP0r. GAFLSRD1

Global Acceptance Filter List Select Routing destination 1

This bit changes a copy destination to CFIFO1 or TXQ1 by routing.

If this bit is set as 1, the preset value of **CFDGAFLP1r.GAFLFDP** will choose TX Queue.

If this bit is cleared to 0, the preset value of **CFDGAFLP1r.GAFLFDP** will choose CommonFIFO.

Users cannot write to these bits when **CFDGAFLCTR.AFLDAE** bit is 1'b0.

Users should only write to the bit when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

4.3.21.4 CFDGAFLP0r. GAFLSRD2

Global Acceptance Filter List Select Routing destination 2

This bit changes a copy destination to CFIFO2 or TXQ2 by routing.

If this bit is set as 1, the preset value of **CFDGAFLP1r.GAFLFDP** will choose TX Queue.

If this bit is cleared to 0, the preset value of **CFDGAFLP1r.GAFLFDP** will choose CommonFIFO.

Users cannot write to these bits when **CFDGAFLECTR.AFLDAE** bit is 1'b0.

Users should only write to the bit when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

4.3.21.5 CFDGAFLP0r.GAFLIFL0

Global Acceptance Filter List Information Label 0

These bits allow the configuration of a 2-bit Information label that will be attached to a received message accepted by the related Global Acceptance Filter List entry. This bit is a LSB bit of an information label.

Users cannot write to these bits when **CFDGAFLECTR.AFLDAE** bit is 1'b0.

Users should only write to the bit when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

This bit is stored in Information Label Field[0] (**CFDRMFDSTS.RMIFL[0]**, **CFDRFFDSTS.RFIFL[0]**, **CFDCFFDCSTS.CFIFL[0]**) of the storage location of an incoming message.

This bit is stored in **CFDTHLACC1n.TIFL[0]** when **CFDTHLCCn.THLDGE=1** is set up using GW function.

4.3.21.6 CFDGAFLP0r.GAFLRMDP

Global Acceptance Filter List RX Message Buffer Direction Pointer

These bits allow the configuration of a single reception Message Buffer as the destination target for a received message that is passing the acceptance check of the related Global Acceptance Filter List entry. The value entered is the single destination Message Buffer number.

Users cannot write to these bits when **CFDGAFLECTR.AFLDAE** bit is 1'b0.

Users should only write to these bits when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

CFDRMNB.NRXMB[7:0] is the value entered in the RX Message Buffer Number Register to configure the number of RX Message Buffers. The value to be entered in **CFDGAFLP0r.GAFLRMDP[6:0]** bits should only be between 7'h0 and (**CFDRMNB.NRXMB[7:0]** - 8'h1).

If **CFDRMNB.NRXMB[7:0]** = 8'h0, then the **GAFLRMV** bit should be configured as 1'b0.

4.3.21.7 CFDGAFLP0r.GAFLRMV

Global Acceptance Filter List RX Message Buffer Valid

This bit allows the enabling/disabling of a single reception Message Buffer as the target for a received message that is passing the acceptance check of the related Global Acceptance Filter List entry.

Users cannot write to these bits when **CFDGAFLECTR.AFLDAE** bit is 1'b0.

Users should only write to these bits when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

4.3.21.8 CFDGAFLP0r.GAFLPTR

Global Acceptance Filter List Pointer Field

These bits allow the configuration of a 16-bit pointer that will be attached to a received message accepted by the related Global Acceptance Filter List entry. The Pointer will be added during message storage in the Message Buffer area and can be used by the application as support function. The pointer information could be used for example to support PDU Identifier allocation for the received message in AUTOSAR systems.

Users cannot write to these bits when **CFDGAFLECTR.AFLDAE** bit is 1'b0.

Users should only write to these bits when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

4.3.22 CFDGAFLP1r

Global Acceptance Filter List Pointer 1 Registers r = [1...10]h

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
	GAFLFDP[31:24]							
Value after reset	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
	GAFLFDP[23:16]							
Value after reset	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
	GAFLFDP[15:8]							
Value after reset	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
	GAFLFDP[7:0]							
Value after reset	0	0	0	0	0	0	0	0
Bit	Symbol	Bit name	Function					R/W
b[31:0]	GAFLFDP	Global Acceptance Filter List FIFO Direction Pointer (GAFLFDP[((n+1)*3+7):0])	FIFO direction pointer bits for received message storage					R/W

The Global Acceptance Filter List Pointer 1 registers are used to configure the FIFO direction pointer fields in each Rule Entry of the Global Acceptance Filter List.

4.3.22.1 CFDGAFLP1r.GAFLFDP

Global Acceptance Filter List FIFO Direction Pointer (GAFLFDP[((n+1)*3+7):0])

These bits allow the configuration of FIFO Buffers as the target for a received message passing the acceptance check of the related Global Acceptance Filter List entry. Each bit of the **CFDGAFLP1r.GAFLFDP[31:0]** is configuring a dedicated FIFO:

Bit	Value (Binary)	Function
b0	0	Disable RX FIFO 0 as target for reception
	1	Enable RX FIFO 0 as target for reception
b1	0	Disable RX FIFO 1 as target for reception
	1	Enable RX FIFO 1 as target for reception
b2	0	Disable RX FIFO 2 as target for reception
	1	Enable RX FIFO 2 as target for reception
b3	0	Disable RX FIFO 3 as target for reception
	1	Enable RX FIFO 3 as target for reception
b4	0	Disable RX FIFO 4 as target for reception
	1	Enable RX FIFO 4 as target for reception
b5	0	Disable RX FIFO 5 as target for reception
	1	Enable RX FIFO 5 as target for reception
b6	0	Disable RX FIFO 6 as target for reception
	1	Enable RX FIFO 6 as target for reception
b7	0	Disable RX FIFO 7 as target for reception
	1	Enable RX FIFO 7 as target for reception
b8	0	Disable Common FIFO 0 and Channel 0 TX Queue 0 as target for reception
	1	GAFLSRD0=0: Enable Common FIFO 0 as target for reception GAFLSRD0=1: Enable Channel 0 TX Queue 0 as target for reception

b9	0	Disable Common FIFO 1 and Channel 0 TX Queue 1 as target for reception
	1	GAFLSRD1=0: Enable Common FIFO 1 as target for reception GAFLSRD1=1: Enable Channel 0 TX Queue 1 as target for reception
b10	0	Disable Common FIFO 2 and Channel 0 TX Queue 2 as target for reception
	1	GAFLSRD2=0: Enable Common FIFO 2 as target for reception GAFLSRD2=1: Enable Channel 0 TX Queue 2 as target for reception
b11	0	Disable Common FIFO 3 and Channel 1 TX Queue 0 as target for reception
	1	GAFLSRD0=0: Enable Common FIFO 3 as target for reception GAFLSRD0=1: Enable Channel 1 TX Queue 0 as target for reception
b12	0	Disable Common FIFO 4 and Channel 1 TX Queue 1 as target for reception
	1	GAFLSRD1=0: Enable Common FIFO 4 as target for reception GAFLSRD1=1: Enable Channel 1 TX Queue 1 as target for reception
b13	0	Disable Common FIFO 5 and Channel 1 TX Queue 2 as target for reception
	1	GAFLSRD2=0: Enable Common FIFO 5 as target for reception GAFLSRD2=1: Enable Channel 1 TX Queue 2 as target for reception
b14	0	Disable Common FIFO 6 and Channel 2 TX Queue 0 as target for reception
	1	GAFLSRD0=0: Enable Common FIFO 6 as target for reception GAFLSRD0=1: Enable Channel 2 TX Queue 0 as target for reception
b15	0	Disable Common FIFO 7 and Channel 2 TX Queue 1 as target for reception
	1	GAFLSRD1=0: Enable Common FIFO 7 as target for reception GAFLSRD1=1: Enable Channel 2 TX Queue 1 as target for reception
b16	0	Disable Common FIFO 8 and Channel 2 TX Queue 2 as target for reception
	1	GAFLSRD2=0: Enable Common FIFO 8 as target for reception GAFLSRD2=1: Enable Channel 2 TX Queue 2 as target for reception
b17	0	Disable Common FIFO 9 and Channel 3 TX Queue 0 as target for reception
	1	GAFLSRD0=0: Enable Common FIFO 9 as target for reception GAFLSRD0=1: Enable Channel 3 TX Queue 0 as target for reception
b18	0	Disable Common FIFO 10 and Channel 3 TX Queue 1 as target for reception

	1	GAFLSRD1 =0:Enable Common FIFO 10 as target for reception GAFLSRD1 =1:Enable Channel 3 TX Queue 1 as target for reception
b19	0	Disable Common FIFO 11 and Channel 3 TX Queue 2 as target for reception
	1	GAFLSRD2 =0:Enable Common FIFO 11 as target for reception GAFLSRD2 =1:Enable Channel 3 TX Queue 2 as target for reception
b20	0	Disable Common FIFO 12 and Channel 4 TX Queue 0 as target for reception
	1	GAFLSRD0 =0:Enable Common FIFO 12 as target for reception GAFLSRD0 =1:Enable Channel 4 TX Queue 0 as target for reception
b21	0	Disable Common FIFO 13 and Channel 4 TX Queue 1 as target for reception
	1	GAFLSRD1 =0:Enable Common FIFO 13 as target for reception GAFLSRD1 =1:Enable Channel 4 TX Queue 1 as target for reception
b22	0	Disable Common FIFO 14 and Channel 4 TX Queue 2 as target for reception
	1	GAFLSRD2 =0:Enable Common FIFO 14 as target for reception GAFLSRD2 =1:Enable Channel 4 TX Queue 2 as target for reception
b23	0	Disable Common FIFO 15 and Channel 5 TX Queue 0 as target for reception
	1	GAFLSRD0 =0:Enable Common FIFO 15 as target for reception GAFLSRD0 =1:Enable Channel 5 TX Queue 0 as target for reception
b24	0	Disable Common FIFO 16 and Channel 5 TX Queue 1 as target for reception
	1	GAFLSRD1 =0:Enable Common FIFO 16 as target for reception GAFLSRD1 =1:Enable Channel 5 TX Queue 1 as target for reception
b25	0	Disable Common FIFO 17 and Channel 5 TX Queue 2 as target for reception
	1	GAFLSRD2 =0:Enable Common FIFO 17 as target for reception GAFLSRD2 =1:Enable Channel 5 TX Queue 2 as target for reception
b26	0	Disable Common FIFO 18 and Channel 6 TX Queue 0 as target for reception
	1	GAFLSRD0 =0:Enable Common FIFO 18 as target for reception GAFLSRD0 =1:Enable Channel 6 TX Queue 0 as target for reception
b27	0	Disable Common FIFO 19 and Channel 6 TX Queue 1 as target for reception
	1	GAFLSRD1 =0:Enable Common FIFO 19 as target for reception

		GAFLSRD1=1: Enable Channel 6 TX Queue 1 as target for reception
b28	0	Disable Common FIFO 20 and Channel 6 TX Queue 2 as target for reception
	1	GAFLSRD2=0: Enable Common FIFO 20 as target for reception GAFLSRD2=1: Enable Channel 6 TX Queue 2 as target for reception
b29	0	Disable Common FIFO 21 and Channel 7 TX Queue 0 as target for reception
	1	GAFLSRD0=0: Enable Common FIFO 21 as target for reception GAFLSRD0=1: Enable Channel 7 TX Queue 0 as target for reception
b30	0	Disable Common FIFO 22 and Channel 7 TX Queue 1 as target for reception
	1	GAFLSRD1=0: Enable Common FIFO 22 as target for reception GAFLSRD1=1: Enable Channel 7 TX Queue 1 as target for reception
b31	0	Disable Common FIFO 23 and Channel 7 TX Queue 2 as target for reception
	1	GAFLSRD2=0: Enable Common FIFO 23 as target for reception GAFLSRD2=1: Enable Channel 7 TX Queue 2 as target for reception

Users cannot write to these bits when **CFDGAFLECTR.AFLDAE** bit is 1'b0.

For storage in Common FIFO, target for reception can only be those Common FIFO Buffers that are configured as RX FIFO or GW FIFO.

For storage in TX Queue, when these TX Queue buffers of a target are in GW mode, it can do.

Users should only write to these bits when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

Users should only configure up to 8 destination FIFO Buffers or 7 destination FIFO Buffers plus one RX Message Buffer or 8 destination TX Queue Buffers or 7 destination TX Queue Buffers plus one RX Message Buffer.

Or a setup of a maximum of 8 destinations in all is possible at FIFO buffer and TX Queue buffer.

4.3.23 CFDRMNB

RX Message Buffer Number Register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24	
Value after reset	-	-	-	-	-	-	-	-	
	0	0	0	0	0	0	0	0	
	b23	b22	b21	b20	b19	b18	b17	b16	
Value after reset	-	-	-	-	-	-	-	-	
	0	0	0	0	0	0	0	0	
	b15	b14	b13	b12	b11	b10	b9	b8	
Value after reset	-	-	-	-	-	0	0	0	
	0	0	0	0	0	0	0	0	
	b7	b6	b5	b4	b3	b2	b1	b0	
	NRXMB[7:0]								
Value after reset	0	0	0	0	0	0	0	0	
Bit	Symbol	Bit name	Function						R/W
b[31:11]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[10:8]	RMPLS	Reception Message Buffer Payload Data Size	3'b000: 8 Bytes 3'b001: 12 Bytes 3'b010: 16 Bytes 3'b011: 20 Bytes 3'b100: 24 Bytes 3'b101: 32 Bytes 3'b110: 48 Bytes 3'b111: 64 Bytes						R/W
b[7:0]	NRXMB	Number of RX Message Buffers	Used to define the number of RX Message Buffer						R/W

The RX Message Buffer Number register is used to configure the total number of RX Message Buffers allocated to all channels.

4.3.23.1 CFDRMNB.NRXMB

Number of RX Message Buffers

These bits are used to configure the number of RX Message Buffers.

Users can only write to these bits when RS-CAN-FD module is in GL_RESET modes.

Users should enter, only the values between 0 and $((n+1)*16)$ inclusive (where n is the number of channels), with 8'h0 indicating that, no RX Message Buffer is allocated.

4.3.23.2 CFDRMNB.RMPLS

Reception Message Buffer Payload Data Size

These bits are used to configure the message buffer payload data size.

Users can only write to these bits when RS-CAN-FD module is in GL_RESET modes.

4.3.24 CFDRMNDt

RX Message Buffer New Data Register t

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
	RMNSu[31:24]							
Value after reset	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
Value after reset	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset	0	0	0	0	0	0	0	0
Bit	Symbol	Bit name	Function					R/W
b[31:0]	RMNSu	RX Message Buffer New Data Status	1'b0: New Data not stored in corresponding RX Message Buffer 1'b1: New Data stored in corresponding RX Message Buffer					R/W

The RX Message Buffer New Data status register bits show the New Data storage status of the RX Message Buffers.

4.3.24.1 CFDRMNDt.RMNSu

RX Message Buffer New Data Status

These bits show the NewData Flag status for the corresponding RX Message Buffer. RMNS bit 0 corresponds to RX Message Buffer 0 and so on.

(no_of_channels = 8)
 (no_of_CFDRMBCPs_per_channel = No. of RX Message Buffer Components per Channel = 16)
 (no_of_CFDRMBCPs = No. of RX Message Buffer Components =
 no_of_channels * no_of_CFDRMBCPs_per_channel = 8 * 16 = 128)
 (no_of_bits_per_register = 32)
 (no_of_CFDRMNDs = No. of CFDRMND Registers = no_of_CFDRMBCPs / no_of_bits_per_register = 128 / 32 = 4)
 (t = [0...no_of_CFDRMNDs-1])

(u = [t*32...(no_of_CFDRMBCPs - ((no_of_CFDRMNDs - 1 - t) * 32) - 1)])

(t) can be calculated from the desired NewData status flag (u) using the formula $t = \text{floor}(u / 32)$

Bit position can be calculated using the formula $(u - (t * 32))$

e.g. for NewData Status Flag 47 we have:

$u = 47, t = \text{floor}(47 / 32) = 1$ and bit position = $(47 - (1 * 32)) = 15$

Therefore, **Users should read the bit 15 of CFDRMND1.**

Users cannot write to these bits when RS-CAN-FD module is in GL_RESET or GL_SLEEP mode.

Writing 1'b1 has no influence on the bit values.

This bit cannot be cleared when message storage in the corresponding RX Message Buffer is in progress.

Do not use bit clear instruction for clearing these bits. Use the MOV instruction to ensure that only the specified bit is cleared. Then the other bits remain set to 1'b1.

The field is set automatically when storage of a new message starts in the corresponding RX Message Buffer.

The duration of message storage time is 6 peripheral clock (pclk) cycles for **CFDRMNB.RMPLS** = 3'b000

(max 8 byte payload).

For **CFDRMNB.RMPLS** > 3'b000 it is 6 + 1 for each four byte (max 20 peripheral clock (pclk) cycles for 64 byte).

This bit is cleared by writing a 1'b0 to it.

This bit is cleared automatically when RS-CAN-FD module enters GL_RESET mode.

4.3.25 CFDRFCCa

RX FIFO Configuration / Control Registers a

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
	-	-	-	DMAE	-	-	-	MEIE
Value after reset	0	0	0	0	0	0	0	0

	b23	b22	b21	b20	b19	b18	b17	b16
	-	-	-	-	-	-	-	RFFIE
Value after reset	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8
	RFIGCV[2:0]			RFIM	-	RFDC[2:0]		
Value after reset	0	0	0	0	0	0	0	0

	b7	b6	b5	b4	b3	b2	b1	b0
	-	RFPLS[2:0]			-	-	RFIE	RFE
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:29]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[28]	DMAE	DMA Transfer Enable for RXFIFO for FFI Mode	1'b0: DMA Transfer Request disabled for RXFIFO a 1'b1: DMA Transfer Request enabled for RXFIFO a	R/W
b[27:25]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[24]	MEIE	Message lost Error Interrupt Enable for FFI Mode	1'b0: Message Lost Error Interrupt Disabled 1'b1: Message Lost Error Interrupt Enabled	R/W
b[23:17]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[16]	RFFIE	RX FIFO Full interrupt Enable	1'b0: FIFO Interrupt generation disabled 1'b1: FIFO Interrupt generation enabled	R/W
b[15:13]	RFIGCV	RX FIFO Interrupt Generation Counter Value	3'b000: Interrupt generated when FIFO is 1/8 th Full 3'b001: Interrupt generated when FIFO is 1/4 th Full 3'b010: Interrupt generated when FIFO is 3/8 th Full 3'b011: Interrupt generated when FIFO is 1/2 Full 3'b100: Interrupt generated when FIFO is 5/8 th Full 3'b101: Interrupt generated when FIFO is 3/4 th Full 3'b110: Interrupt generated when FIFO is 7/8 th Full 3'b111: Interrupt generated when FIFO is Full	R/W
b[12]	RFIM	RX FIFO Interrupt Mode	1'b0: Interrupt generated when RX FIFO counter reaches RFIGCV value from values smaller than RFIGCV 1'b1: Interrupt generated at the end of every received message storage	R/W
b[11]	-	Reserved	This bit is read as 0b. The write value should be always 0b.	R/W
b[10:8]	RFDC	RX FIFO Depth Configuration	3'b000: FIFO Depth = 0 Messages 3'b001: FIFO Depth = 4 Messages 3'b010: FIFO Depth = 8 Messages 3'b011: FIFO Depth = 16 Messages 3'b100: FIFO Depth = 32 Messages 3'b101: FIFO Depth = 48 Messages 3'b110: FIFO Depth = 64 Messages 3'b111: FIFO Depth = 128 Messages	R/W
b[7]	-	Reserved	This bit is read as 0b. The write value should be always 0b.	R/W
b[6:4]	RFPLS	Rx FIFO Payload Data Size configuration	3'b000: 8 Bytes 3'b001: 12 Bytes 3'b010: 16 Bytes	R/W

3'b011: 20 Bytes
 3'b100: 24 Bytes
 3'b101: 32 Bytes
 3'b110: 48 Bytes
 3'b111: 64 Bytes

b[3:2]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[1]	RFIE	RX FIFO Interrupt Enable	1'b0: FIFO Interrupt generation disabled 1'b1: FIFO Interrupt generation enabled	R/W
b[0]	RFE	RX FIFO Enable	1'b0: FIFO disabled 1'b1: FIFO enabled	R/W

The RX FIFO Configuration / Control registers are used to configure and control the 8 RX FIFOs.

(a = RX FIFO index = [0...no_of_RFIFOs-1])

(no_of_RFIFOs = No. of RX FIFOs = 8)

4.3.25.1 CFDRFCCa.RFE

RX FIFO Enable

This bit enables the FIFO when it is set. If this bit is cleared, the RX FIFO will be cleared and is empty.

Users can only write to this bit when RS-CAN-FD module is in GL_HALT or GL_OPERATION modes.
This bit can only be set if the configured FIFO depth is greater than 3'h0 (**CFDRFCCa.RFDC** > 3'b000).

The **CFDRFCCa.RFE** bit should be set by a separate write access to the **CFDRFCCa** register, after all the other bits of the **CFDRFCCa** register have been set.

This bit is cleared automatically when RS-CAN-FD module enters GL_RESET mode.

4.3.25.2 CFDRFCCa.RFIE

RX FIFO Interrupt Enable

This bit enables generation of the FIFO Interrupt when it is set.

Users cannot write to this bit when RS-CAN-FD module is in GL_SLEEP mode.

4.3.25.3 CFDRFCCa.RFPLS

Rx FIFO Payload Data Size configuration

These bits define the message data payload allocation in the RAM.
This is the max. number of Bytes which can be received by this FIFO.
Please refer to Section 8.2.1.4 for details.

Users can only write to these bits when RS-CAN-FD module is in GL_RESET mode.

4.3.25.4 CFDRFCCa.RFDC

RX FIFO Depth Configuration

These bits select the depth of the FIFO in terms of number of Messages. If the FIFO depth is configured to 0 Messages then the FIFO cannot be used.

Users can only write to these bits when RS-CAN-FD module is in GL_RESET mode.

4.3.25.5 CFDRFCCa.RFIM

RX FIFO Interrupt Mode

This bit selects the Interrupt generation condition for the FIFO.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

Users should only write to this bit when RS-CAN-FD module is in GL_RESET mode.

4.3.25.6 CFDRFCCa.RFIGCV

RX FIFO Interrupt Generation Counter Value

D012250_111_S01 VERSION 1.021.01

©RENESAS SYSTEM DESIGN

LEVEL 3 – CONFIDENTIAL

STATUS: Issued

FORM EM-PRJ3A (APRIL 2017) VERSION 4.1A

PAGE 126 OF 648

These bits select the counter value of the FIFO for generation of FIFO Interrupt. These values represent fractions of the FIFO depth for which Interrupt is generated.

Users cannot write to these bits when the RS-CAN-FD module is in GL_SLEEP mode.

The setting of these bits should be synchronised with the **CFDRFCCa.RFDC** bits. Refer to Section 8.2.1.5 for detailed information.

Users should only write to these bits when RS-CAN-FD module is in GL_RESET mode.

4.3.25.7 CFDRFCCa.RFFIE

RX FIFO Full Interrupt Enable

This bit enables generation of the RXFIFO Full Interrupt when it is set.

Users cannot write to the bit when the RS-CAN-FD module is in GL_SLEEP mode.

The following contents are imagined as how to use interruption.

1. Interruption output in number of arbitrary stages (**CFDRFCCa.RFIGCV**)

2. Interruption output in FIFO full state

Management of the receiving data of FIFO can be performed by these notices of interruption.

4.3.25.8 CFDRFCCa.MEIE

Message lost Error Interrupt Enable for FFI Mode

This bit is used instead of **CFDGCTR.MEIE** when **CFDGFFIMC.FFIEN** is 1'b1 and **CFDGVMEIS.FMLT** is 1'b1.

Users can only access to this bit when **CFDGFFIMC.FFIEN** is 1'b1.

Users cannot write to this bit when RS-CAN-FD module is in GL_SLEEP mode.

4.3.25.9 CFDRFCCa.DMAE

DMA Transfer Enable for RXFIFO for FFI Mode

Users can only access to this bit when **CFDGFFIMC.FFIEN** is 1'b1.

This bit is cleared automatically when the RS-CAN-FD module is in GL_RESET.

4.3.26 CFDRFSTS_a

RX FIFO Status Registers a

Address: Refer to section 4.1									
Value after reset									
b31	b30	b29	b28	b27	b26	b25	b24		
-	-	-	RFDMASTS	-	-	-	-		
Value after reset	0	0	0	0	0	0	0		
b23	b22	b21	b20	b19	b18	b17	b16		
-	-	-	-	-	-	-	-	RFFF	
Value after reset	0	0	0	0	0	0	0		
b15	b14	b13	b12	b11	b10	b9	b8		
RFMC[7:0]									
Value after reset	0	0	0	0	0	0	0		
b7	b6	b5	b4	b3	b2	b1	b0		
-	-	-	-	RFIF	RFMLT	RFFLL	RFEMP		
Value after reset	0	0	0	0	0	0	0		
Bit	Symbol	Bit name	Function						R/W
b[31:29]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[28]	RFDMASTS	DMA Transfer Status for RX FIFO for FFI Mode	1'b0: DMA transfer stopped 1'b1: DMA transfer ongoing						R
b[27:17]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[16]	RFFIF	RX FIFO Full Interrupt Flag	1'b0: FIFO Full interrupt condition not satisfied 1'b1: FIFO Full interrupt condition satisfied						R/W
b[15:8]	RFMC	RX FIFO Message Count	Number of Messages stored in FIFO						R
b[7:4]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[3]	RFIF	RX FIFO Interrupt Flag	1'b0: FIFO Interrupt condition not satisfied 1'b1: FIFO Interrupt condition satisfied						R/W
b[2]	RFMLT	RX FIFO Message Lost	1'b0: No Message Lost in FIFO 1'b1: FIFO Message Lost						R/W
b[1]	RFFLL	RX FIFO Full	1'b0: FIFO Not Full 1'b1: FIFO Full						R
b[0]	RFEMP	RX FIFO Empty	1'b0: FIFO Not Empty 1'b1: FIFO Empty						R

The FIFO status registers show the status of the messages stored in corresponding FIFO Buffers.

(a = RX FIFO index = [0...no_of_RFIFOs-1])

(no_of_RFIFOs = No. of RX FIFOs = 8)

4.3.26.1 CFDRFSTS_a.RFEMP

RX FIFO Empty

This bit is set automatically when **CFDRFSTS_a.RFMC** is 8'h0.

This bit is set automatically when RX FIFO is disabled by setting the **CFDRFCCa.RFE** bit to 1'b0.

This bit is set automatically when RS-CAN-FD module enters GL_RESET mode.

This bit is cleared automatically when the first message is stored in the RX FIFO Buffer.

4.3.26.2 CFDRFSTS_a.RFFLL

RX FIFO Full

This bit is set automatically when number of CAN messages stored in the FIFO matches the configured FIFO depth.

This bit is cleared automatically when the number of CAN messages stored in the FIFO is less than the configured FIFO depth.

This bit is cleared automatically when RX FIFO is disabled by setting the **CFDRFCCa.RFE** bit to 1'b0.
This bit is cleared automatically when RS-CAN-FD module enters GL_RESET mode.

4.3.26.3 CFDRFSTSa.RFMLT

RX FIFO Message Lost

Users can only write to this bit when RS-CAN-FD module is in GL_HALT or GL_OPERATION modes.

Writing 1'b1 has no influence on the bit values.

Do not use bit clear instruction for clearing this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Then the other bits remain set to 1'b1.

This bit is set automatically whenever a message is lost due to attempted storage when the FIFO is already full.

If the set from the CAN channel occurs simultaneously with the clear by the write access, then the bit is set.

The bit is cleared by writing 1'b0 to it.

The bit is cleared when RS-CAN-FD module enters GL_RESET mode.

4.3.26.4 CFDRFSTSa.RFIF

RX FIFO Interrupt Flag

This bit will not be cleared automatically if the RX FIFO Buffer is disabled.

Users can only write to this bit when RS-CAN-FD module is in GL_HALT or GL_OPERATION modes.
Writing 1'b1 has no influence on the bit values.

Do not use bit clear instruction for clearing this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Then the other bits remain set to 1'b1.

This bit is set automatically when the configured Interrupt condition is satisfied.

If the set from the CAN channel occurs simultaneously with the clear by the write access, then the bit is set.

The bit is cleared by writing 1'b0 to it.

The bit is cleared when RS-CAN-FD module enters GL_RESET mode.

4.3.26.5 CFDRFSTSa.RFMC

RX FIFO Message Count

These bits indicate the number of CAN messages stored in the RX FIFO that can be read by the CPU.

These bits are cleared automatically when the FIFO is disabled.

These bits are cleared automatically when RS-CAN-FD module enters GL_RESET mode.

4.3.26.6 CFDRFSTSa.RFFIF

RX FIFO Full Interrupt Flag

This bit will not be cleared automatically if the RX FIFO Buffer is disabled.

Users can only write to this bit when RS-CAN-FD module is in GL_HALT or GL_OPERATION modes.

Writing 1'b1 has no influence on the bit values.

Do not use bit clear instruction for clearing this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Then the other bits remain set to 1'b1.

This bit is set automatically when the FIFO full Interrupt condition is satisfied.

If the set from the CAN channel occurs simultaneously with the clear by the write access, then the bit is set.

The bit is cleared by writing 1'b0 to it.

The bit is cleared when RS-CAN-FD module enters GL_RESET mode.

4.3.26.7 CFDRFSTS_a.RFDMASTS

DMA Transfer Status for RX FIFO for FFI Mode

When **CFDGFFIMC.FFIEN** is 1 'b 0, the read value of this bit is 0.

4.3.27 CFDRFPCTR_a

RX FIFO Pointer Control Registers

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24	
Value after reset	-	-	-	-	-	-	-	-	
	0	0	0	0	0	0	0	0	
	b23	b22	b21	b20	b19	b18	b17	b16	
Value after reset	-	-	-	-	-	-	-	-	
	0	0	0	0	0	0	0	0	
	b15	b14	b13	b12	b11	b10	b9	b8	
Value after reset	-	-	-	-	-	-	-	-	
	0	0	0	0	0	0	0	0	
	b7	b6	b5	b4	b3	b2	b1	b0	
	RFPC[7:0]								
Value after reset	0	0	0	0	0	0	0	0	
Bit	Symbol	Bit name	Function						R/W
b[31:8]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[7:0]	RFPC	RX FIFO Pointer Control	Increments read pointer of the corresponding RX FIFO Buffers						W

These registers can be used to increment the Read Pointer of the corresponding RX FIFO Buffers.

(a = RX FIFO index = [0...no_of_RFIFOs-1])

(no_of_RFIFOs = No. of RX FIFOs = 8)

4.3.27.1 CFDRFPCTR_a.RFPC

RX FIFO Pointer Control

When the value 8'hFF is written to these bits, then the Pointer of the corresponding RX FIFO Buffer is moved to the next FIFO entry.

Read value from these bits is always 8'h0.

Users can only write to these bits when RS-CAN-FD module is in GL_HALT or GL_OPERATION modes.

Users should only write 8'hFF to these registers when the corresponding RX FIFO is enabled and not empty.

Users should not write to the FIFO control registers when DMA is enabled.

4.3.28 CFDCFCCd

Common FIFO Configuration / Control Registers d

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
CFITT[7:0]								
Value after reset	0	0	0	0	0	0	0	0

	b23	b22	b21	b20	b19	b18	b17	b16
CFDC[2:0]								
Value after reset	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8
CFIGCV[2:0]								
Value after reset	0	0	0	0	0	0	0	0

	b7	b6	b5	b4	b3	b2	b1	b0
CFPLS[2:0]								
Value after reset	-	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:24]	CFITT	Common FIFO Interval Transmission Time	Delay the start of transmission from the FIFO if configured in TX or GW mode, delay is a multiple of basic Interval Timer Clock Source unit	R/W
b[23:21]	CFDC	Common FIFO Depth Configuration	3'b000: FIFO Depth = 0 Messages 3'b001: FIFO Depth = 4 Messages 3'b010: FIFO Depth = 8 Messages 3'b011: FIFO Depth = 16 Messages 3'b100: FIFO Depth = 32 Messages 3'b101: FIFO Depth = 48 Messages 3'b110: FIFO Depth = 64 Messages 3'b111: FIFO Depth = 128 Messages	R/W
b[20:16]	CFTML	Common FIFO TX Message Buffer Link	Transmission scan link position of the corresponding channel	R/W
b[15:13]	CFIGCV	Common FIFO Interrupt Generation Counter Value	3'b000: Interrupt generated when FIFO is 1/8 th Full 3'b001: Interrupt generated when FIFO is 1/4 th Full 3'b010: Interrupt generated when FIFO is 3/8 th Full 3'b011: Interrupt generated when FIFO is 1/2 Full 3'b100: Interrupt generated when FIFO is 5/8 th Full 3'b101: Interrupt generated when FIFO is 3/4 th Full 3'b110: Interrupt generated when FIFO is 7/8 th Full 3'b111: Interrupt generated when FIFO is Full	R/W
b[12]	CFIM	Common FIFO Interrupt Mode	1'b0: RX FIFO Mode: RX Interrupt generated when Common FIFO counter reaches CFIGCV value from a lower value; TX FIFO Mode: TX Interrupt generated when Common FIFO transmits the last message successfully; GW FIFO Mode: For RX interrupt flag:	R/W

Interrupt generated when FIFO counter increments and reaches the value configured in CFIGCV;

For TX interrupt flag:

Interrupt generated when FIFO transmits the last message successfully;

1'b1:

RX FIFO Mode: RX Interrupt generated at the end of every received message storage;

TX FIFO Mode: Interrupt generated for every successfully transmitted message;

GW FIFO Mode:

For RX interrupt flag:

Interrupt generated when a message is stored in the FIFO;

For TX interrupt flag:

Interrupt generated when a message is successfully transmitted from the FIFO;

b[11]	CFITR	Common FIFO Interval Timer Resolution	1'b0: Reference Clock Period x1 1'b1: Reference Clock Period x10	R/W
b[10]	CFITSS	Common FIFO Interval Timer Source Select	1'b0: Reference Clock (x1 / x10 period) 1'b1: Bit Time Clock of related channel (FIFO is linked to fixed channel)	R/W
b[9:8]	CFM	Common FIFO Mode	2'b00: RX FIFO Mode 2'b01: TX FIFO Mode 2'b10: CAN – CAN GW FIFO Mode 2'b11: Reserved	R/W
b[7]	-	Reserved	This bit is read as 0b. The write value should be always 0b.	R/W
b[6:4]	CFPLS	Common FIFO Payload Data size configuration	3'b000: 8 Bytes 3'b001: 12 Bytes 3'b010: 16 Bytes 3'b011: 20 Bytes 3'b100: 24 Bytes 3'b101: 32 Bytes 3'b110: 48 Bytes 3'b111: 64 Bytes	R/W
b[3]	-	Reserved	This bit is read as 0b. The write value should be always 0b.	R/W
b[2]	CFTXIE	Common FIFO TX Interrupt Enable	1'b0: FIFO Interrupt generation disabled for Frame TX 1'b1: FIFO Interrupt generation enabled for Frame TX	R/W
b[1]	CFRXIE	Common FIFO RX Interrupt Enable	1'b0: FIFO Interrupt generation disabled for Frame RX 1'b1: FIFO Interrupt generation enabled for Frame RX	R/W
b[0]	CFE	Common FIFO Enable	1'b0: FIFO disabled 1'b1: FIFO enabled	R/W

The Common FIFO Configuration / Control registers are used to configure the Common FIFOs.
(no_of_channels = 8)

(no_of_CFIFOs_per_channel = No. of Common FIFOs per Channel = 3)

Where the total number of CFIFOs = no_of_CFIFOs = no_of_CFIFOs_per_channel * no_of_channels = 3 * 8 = 24 as shown in Figure 8.1

(d = Common FIFO index = [0 .. no_of_CFIFOs -1])

4.3.28.1 CFDCFCCd.CFE

Common FIFO Enable

This bit enables the FIFO when it is set.

FIFO is disabled when this bit is cleared.

This bit can also be used, by clearing it, to abort transmission from Common FIFO when configured in TX Mode or GW Mode.

This bit can also be used, by clearing it, to stop reception into the Common FIFO in RX mode.

Users can only write to this bit when RS-CAN-FD module is in GL_HALT or GL_OPERATION modes and the related RS-CAN-FD channel is not in CH_RESET mode for FIFOs configured as TX or GW FIFO.

This bit can only be set if the configured FIFO depth is greater than 3'h0 (**CFDCFCCd.CFDC** > 3'b000).

The **CFDCFCCd.CFE** bit should be set by a separate write access to the **CFDCFCCd** register, after all the other bits of the **CFDCFCCd** register have been set.

This bit is cleared automatically when RS-CAN-FD module enters GL_RESET mode.

This bit is cleared automatically when the related channel enters CH_RESET mode if the FIFO is configured in TX or GW mode.

4.3.28.2 CFDCFCCd.CFRXIE

Common FIFO RX Interrupt Enable

This bit enables generation of the FIFO Interrupt when the Interrupt flag is set after reception of a frame in the corresponding FIFO.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

4.3.28.3 CFDCFCCd.CFTXIE

Common FIFO TX Interrupt Enable

This bit enables the generation of the Common FIFO Interrupt when the Interrupt flag is set after transmission of a frame from the corresponding FIFO.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

4.3.28.4 CFDCFCCd.CFPLS

Common FIFO Payload Data size configuration

These bits define the message data payload allocation in the RAM.

This is the max. number of Bytes which can be received or transmitted by this FIFO.

Please refer to Section 8 for details.

Users can only write to this bit when RS-CAN-FD module is in GL_RESET mode.

4.3.28.5 CFDCFCCd.CFM

Common FIFO Mode

These bits select the Mode of the FIFO. When HW Reset is active, all the common FIFO Buffers will be configured in RX FIFO mode.

Users cannot write to these bits in GL_OPERATION & GL_SLEEP modes.

Users should not configure these bits to 2'b11.

Users should only write to this bit when RS-CAN-FD module is in GL_RESET mode.

4.3.28.6 CFDCFCCd.CFITSS

Common FIFO Interval Timer Source Select

This bit selects the basic clock source for the Interval Transmission Timer.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

Users should not write to this bit when the **CFDCFCCd.CFE** bit is set to 1'b1.

Users should not write 1'b1 when CAN-FD communication will be used.

Note bit time clock could be variable depending on the nominal and data rate bit configuration.

4.3.28.7 CFDCFCCd.CFTR

Common FIFO Interval Timer Resolution

This bit selects the resolution of the Reference Clock for the Interval Transmission Timer (Peripheral Clock is the source for the Reference Clock).

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

Users should not write to this bit when the **CFDCFCCd.CFE** bit is set to 1'b1

4.3.28.8 CFDCFCCd.CFIM

Common FIFO Interrupt Mode

This bit selects the Interrupt generation condition for the FIFO.

Users cannot write to this bit in GL_SLEEP mode.

Users should only write to this bit when RS-CAN-FD module is in GL_RESET mode.

4.3.28.9 CFDCFCCd.CFIGCV

Common FIFO Interrupt Generation Counter Value

These bits select the message counter value for the generation of the FIFO Interrupt. These values represent fractions of the FIFO depth at which Interrupt is to be generated.

Users cannot write to these bits when the RS-CAN-FD module is in GL_SLEEP mode.

The setting of these bits should be synchronised with the **CFDCFCCd.CFDC** bits. Refer to Section 8.2.1.5 for detailed information.

Users should only write to this bit when RS-CAN-FD module is in GL_RESET mode.

4.3.28.10 CFDCFCCd.CFTML

Common FIFO TX Message Buffer Link

These bits select the normal transmit Message Buffer position where the TX or GW FIFO is linked to, for transmission scanning.

Users cannot write to these bits in GL_OPERATION & GL_SLEEP modes.

Users should only write to this bit when RS-CAN-FD module is in GL_RESET mode.

4.3.28.11 CFDCFCCd.CFDC

Common FIFO Depth Configuration

These bits select the depth of the Common FIFO in terms of number of Messages. If the FIFO depth is configured to 0 Messages then the FIFO cannot be used.

Users can only write to this bit when RS-CAN-FD module is in GL_RESET mode.

4.3.28.12 CFDCFCCd.CFITT

Common FIFO Interval Transmission Time

These bits select the delay in the start of transmission for all messages transmitted from this FIFO when configured in TX or GW mode. The delay is a multiple of the basic Interval Timer Clock Source period (Reference Clock x1, Reference Clock x10 or Bit Time Clock of the related CAN channel).

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

Users should not write to this bit when the **CFDCFCCd.CFE** bit is set to 1'b1.

For **CFDGCFG.ITRCP[15:0]** = 16'h0 these bits should only be set to 16'h0.

4.3.29 CFDCFCCEd

Common FIFO Configuration / Control Enhancement Register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
	-	-	-	DMAE	-	-	MOWIE	MEIE
Value after reset	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
	-	-	-	-	-	-	-	CFBME
Value after reset	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
	-	-	-	-	-	-	-	CFMOWM
Value after reset	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
	-	-	-	-	-	CFOFTXIE	CFOFRXIE	CFFIE
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:29]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[28]	DMAE	DMA Transfer Enable for Common FIFO for FFI Mode	1'b0: DMA Transfer Request disabled for channel n 1'b1: DMA Transfer Request enabled for channel n	R/W
b[27:26]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[25]	MOWIE	GW FIFO Message overwrite Error Interrupt Enable for FFI Mode	1'b0: GW FIFO Message overwrite Error Interrupt Disabled 1'b1: GW FIFO Message overwrite Error Interrupt Enabled	R/W
b[24]	MEIE	Message lost Error Interrupt Enable for FFI Mode	1'b0: Message Lost Error Interrupt Disabled 1'b1: Message Lost Error Interrupt Enabled	R/W
b[23:17]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[16]	CFBME	Common FIFO Buffering Mode Enable	1'b0: Transmission from Common FIFO 1'b1: Transmission halt from Common FIFO	R/W
b[15:9]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[8]	CFMOWM	Common FIFO message overwrite mode	1'b0: Message discarded mode 1'b1: Message overwrite mode	R/W
b[7:3]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[2]	CFOFTXIE	Common FIFO One Frame Transmission Interrupt Enable	1'b0: One Frame TX Interrupt generation disabled 1'b1: One Frame TX Interrupt generation enabled	R/W
b[1]	CFOFRXIE	Common FIFO One Frame Reception Interrupt Enable	1'b0: One Frame RX Interrupt generation disabled 1'b1: One Frame RX Interrupt generation enabled	R/W
b[0]	CFFIE	Common FIFO Full interrupt Enable	1'b0: FIFO Interrupt generation disabled 1'b1: FIFO Interrupt generation enabled	R/W

The Common FIFO Configuration / Control Enhancement registers are used to configure the Common FIFOs.

(no_of_channels = 8)

(no_of_CFIFOs_per_channel = No. of Common FIFOs per Channel = 3)

Where the total number of CFIFOs = no_of_CFIFOs = no_of_CFIFOs_per_channel * no_of_channels = 3 * 8 = 24 as shown in Figure 8.1

(d = Common FIFO index = [0 .. no_of_CFIFOs -1])

4.3.29.1 CFDCFCCEd.CFFIE

Common FIFO Full interrupt Enable

This bit enables generation of the FIFO full Interrupt when the interrupt flag is set after reception of a frame

in the corresponding FIFO.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

The following contents are imagined as how to use interruption.

1. Interruption output in number of arbitrary stages (**CFDCFCCd.CFIGCV**)

2. Interruption output in FIFO full state

Management of the receiving data of FIFO can be performed by these notices of interruption.

4.3.29.2 CFDCFCCEd.CFOFRXIE

Common FIFO One Frame Reception Interrupt Enable

This bit enables generation of the One Frame Reception Interrupt when the Interrupt flag is set after reception of a frame in the corresponding FIFO.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

4.3.29.3 CFDCFCCEd.CFOFTXIE

Common FIFO One Frame Transmission Interrupt Enable

This bit enables generation of the One Frame Transmission Interrupt when the Interrupt flag is set after transmission of a frame in the corresponding FIFO.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

4.3.29.4 CFDCFCCEd.CFMOWM

Common FIFO message overwrite mode

A receiving message is discarded, when this bit is 0 and FIFO is full.

A receiving message is overwritten, when this bit is 1 and FIFO is full.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

Users should only write 1 to this bit when the Common FIFO is in GW mode.

Users should not write change for this bit when the **CFDCFCCd.CFE** bit is 1'b1.

4.3.29.5 CFDCFCCEd.CFBME

Common FIFO Buffering Mode Enable

When this bit is 0, messages are transmitted from FIFO.

When this bit is 1, messages are not transmitted from FIFO.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

Users should not write 1 from 0 for this bit when the **CFDCFCCd.CFE** bit is 1'b1

4.3.29.6 CFDCFCCEd.MEIE

Message lost Error Interrupt Enable for FFI Mode

This bit is used instead of **CFDGCTR.MEIE** when **CFDGFFIMC.FFIEN** is 1'b1 and **CFDGVMEIS.FMLT** is 1'b1.

Users can only access to this bit when **CFDGFFIMC.FFIEN** is 1'b1.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

4.3.29.7 CFDCFCCEd.MOWEIE

GW FIFO Message overwrite Error Interrupt Enable for FFI Mode

This bit is used instead of **CFDGCTR.MOWEIE** when **CFDGFFIMC.FFIEN** is 1'b1 and **CFDGVMEIS.CFMOW** is 1'b1.

Users can only access to this bit when **CFDGFFIMC.FFIEN** is 1'b1

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

4.3.29.8 CFDCFCCEd.DMAE

DMA Transfer Enable for Common FIFO a for FFI Mode

User can only access to this bit when **CFDGFFIMC.FFIEN** is 1'b1.

This bit is cleared when RS-CAN-FD module enters GL_RESET mode.

4.3.30 CFDCFSTSd

Common FIFO Status Registers d

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
Value after reset	-	-	-	CFDMASTS	-	-	-	CFMOW
	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset	-	-	-	-	-	CFOFTXIF	CFOFRXIF	CFFIF
	b15	b14	b13	b12	b11	b10	b9	b8
Value after reset	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset	-	-	-	CFTXIF	CFRXIF	CFMLT	CFFLL	CFEMP
Bit	Symbol	Bit name	Function					R/W
b[31:29]	-	Reserved	These bits are read as 0. The write value should be always 0.					R/W
b[28]	CFDMASTS	DMA Transfer Status for Common FIFO in FFI Mode	1'b0: DMA Transfer Request disabled for channel n 1'b1: DMA Transfer Request enabled for channel n					R
b[27:25]	-	Reserved	These bits are read as 0. The write value should be always 0.					R/W
b[24]	CFMOW	Common FIFO message overwrite	1'b0: No Message overwrite occurred in FIFO 1'b1: Message overwrite occurred in FIFO					R/W
b[23:19]	-	Reserved	These bits are read as 0. The write value should be always 0.					R/W
b[18]	CFOFTXIF	Common FIFO One Frame Transmission	Every FIFO transmits a frame, a corresponding interrupt is set.					R/W
b[17]	CFOFRXIF	Common FIFO One Frame Reception	Every FIFO receives a frame, a corresponding interrupt is set.					R/W
b[16]	CFFIF	Common FIFO Full Interrupt Flag	1'b0: Interrupt condition not satisfied for FIFO Full interrupt 1'b1: Interrupt condition satisfied for FIFO Full interrupt					R/W
b[15:8]	CFMC	Common FIFO Message Count	No. of Messages stored in FIFO					R
b[7:5]	-	Reserved	These bits are read as 0. The write value should be always 0.					R/W
b[4]	CFTXIF	Common TX FIFO Interrupt Flag	1'b0: FIFO Interrupt condition not satisfied after Frame Transmission 1'b1: FIFO Interrupt condition satisfied after Frame Transmission					R/W
b[3]	CFRXIF	Common RX FIFO Interrupt Flag	1'b0: FIFO Interrupt condition not satisfied after Frame Reception 1'b1: FIFO Interrupt condition satisfied after Frame Reception					R/W
b[2]	CFMLT	Common FIFO Message Lost	1'b0: No Message Lost in FIFO 1'b1: FIFO Message Lost					R/W
b[1]	CFFLL	Common FIFO Full	1'b0: FIFO Not Full 1'b1: FIFO Full					R
b[0]	CFEMP	Common FIFO Empty	1'b0: FIFO Not Empty 1'b1: FIFO Empty					R

The FIFO status registers show the status of the messages stored in corresponding FIFO Buffers.

(no_of_channels = 8)

(no_of_CFIFOs_per_channel = No. of Common FIFOs per Channel = 3)

Where the total number of CFIFOs = no_of_CFIFOs = no_of_CFIFOs_per_channel * no_of_channels = 3 * 8 = 24 as shown in Figure 8.1

(d = Common FIFO index = [0 .. no_of_CFIFOs -1])

4.3.30.1 CFDCFSTSd.CFEMP

Common FIFO Empty

This bit is set automatically when the CPU has read all messages from the FIFO configured in RX mode.

D012250_111_S01 VERSION 1.021.01

STATUS: Issued

©RENESAS SYSTEM DESIGN

FORM EM-PRJ3A (APRIL 2017) VERSION 4.1A

LEVEL 3 – CONFIDENTIAL

PAGE 139 OF 648

This bit is set automatically when all messages have been transmitted from the FIFO configured in TX or GW Mode.

This bit is set automatically when FIFO is disabled by setting the CFE bit to 1'b0.

This bit is set automatically when RS-CAN-FD module enters GL_RESET mode.

This bit is set automatically when RS-CAN-FD module enters CH_RESET when FIFO configured in TX Mode or GW Mode.

This bit is cleared automatically when the first reception message is stored in the FIFO when configured in RX Mode.

This bit is cleared automatically when the first message to be transmitted is stored in the FIFO when configured in TX or GW Mode.

4.3.30.2 CFDCFSTSd.CFFLL

Common FIFO Full

This bit is set automatically when the number of CAN messages stored in the FIFO matches the configured FIFO depth.

This bit is cleared automatically when the number of CAN messages stored in the FIFO is less than the configured FIFO depth.

This bit is cleared automatically when the FIFO is disabled by setting the CFE bit to 1'b0.

This bit is cleared automatically when RS-CAN-FD module enters GL_RESET.

This bit is cleared automatically when RS-CAN-FD module enters CH_RESET when FIFO is configured in TX or GW Mode.

4.3.30.3 CFDCFSTSd.CFMLT

Common FIFO Message Lost

Users can only write to this bit when RS-CAN-FD module is in GL_HALT or GL_OPERATION mode and the related RS-CAN-FD channel is not in CH_RESET mode for FIFOs configured as TX or GW FIFO.

Writing 1'b1 has no influence on the bit values.

Do not use bit clear instruction for clearing this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Then the other bits remain set to 1'b1.

This bit is set automatically whenever a message is lost due to attempted storage of a new message when FIFO is already full in RX or GW mode.

If the set from the CAN channel occurs simultaneously with the clear by the write access, then the bit is set.

The bit is cleared by writing 1'b0 to it.

The bit is cleared when RS-CAN-FD module enters GL_RESET mode.

The bit is cleared when the related channel enters CH_RESET mode if the FIFO is configured in TX or GW mode.

4.3.30.4 CFDCFSTSd.CFRXIF

Common RX FIFO Interrupt Flag

This bit will not be cleared automatically if the Common FIFO Buffer is disabled.

Users can only write to this bit when RS-CAN-FD module is in GL_HALT or GL_OPERATION mode and the related RS-CAN-FD channel is not in CH_RESET mode for FIFOs configured as TX or GW FIFO.

Writing 1'b1 has no influence on the bit values.

Do not use bit clear instruction for clearing this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Then the other bits remain set to 1'b1.

This bit is set automatically when the configured Interrupt condition is satisfied for Common FIFO Buffers when configured in GW mode or RX mode.

The bit is cleared by writing 1'b0 to it.

The bit is cleared when RS-CAN-FD module enters GL_RESET mode.

The bit is cleared when the related channel enters CH_RESET mode if the FIFO is configured in GW mode.

4.3.30.5 CFDCFSTSd.CFTXIF

Common TX FIFO Interrupt Flag

This bit will not be cleared automatically if the Common FIFO Buffer is disabled.

Users can only write to this bit when RS-CAN-FD module is in GL_HALT or GL_OPERATION mode and the related RS-CAN-FD channel is not in CH_RESET mode for FIFOs configured as TX or GW FIFO.

Writing 1'b1 has no influence on the bit values.

Do not use bit clear instruction for clearing this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Then the other bits remain set to 1'b1.

This bit is set automatically when the configured Interrupt condition is satisfied for Common FIFO Buffers configured in GW mode or TX mode.

The bit is cleared by writing 1'b0 to it.

The bit is cleared when RS-CAN-FD module enters GL_RESET mode.

The bit is cleared when the related channel enters CH_RESET mode if the FIFO is configured in TX or GW mode.

4.3.30.6 CFDCFSTSd.CFMC

Common FIFO Message Count

These bits indicate the following:

Number of CAN messages stored by the CPU in the FIFO configured in TX Mode pending for transmission.

Number of CAN messages stored in the FIFO Buffer configured in RX Mode by RS-CAN-FD to be read by the CPU.

Number of CAN messages stored by the RS-CAN-FD in the GW FIFO pending for transmission.

These bits are cleared automatically when the FIFO is disabled.

These bits are cleared automatically when RS-CAN-FD module enters GL_RESET mode.

These bits are cleared automatically when the related channel enters CH_RESET mode if the FIFO is configured in TX or GW mode.

4.3.30.7 CFDCFSTSd.CFFIF

Common FIFO Full Interrupt Flag

This bit will not be cleared automatically if the Common FIFO Buffer is disabled.

Users can only write to this bit when RS-CAN-FD module is in GL_HALT or GL_OPERATION mode and the related RS-CAN-FD channel is not in CH_RESET mode for FIFOs configured as TX or GW FIFO.

Writing 1'b1 has no influence on the bit values.

Do not use bit clear instruction for clearing this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Then the other bits remain set to 1'b1.

This bit is set automatically when the FIFO full Interrupt condition is satisfied for Common FIFO Buffers when configured in GW mode or RX mode.

If the set from the CAN channel occurs simultaneously with the clear by the write access, then the bit is set.

The bit is cleared by writing 1'b0 to it.

The bit is cleared when RS-CAN-FD module enters GL_RESET mode.

The bit is cleared when the related channel enters CH_RESET mode if the FIFO is configured in TX or GW mode.

4.3.30.8 CFDCFSTSd.CFOFRXIF

Common FIFO One Frame Reception Interrupt Flag

This bit will not be cleared automatically if the Common FIFO Buffer is disabled.

Users can only write to this bit when RS-CAN-FD module is in GL_HALT or GL_OPERATION mode and the related RS-CAN-FD channel is not in CH_RESET mode for FIFOs configured as TX or GW FIFO.

Writing 1'b1 has no influence on the bit values.

Do not use bit clear instruction for clearing this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Then the other bits remain set to 1'b1.

This bit is set automatically when the One Frame Reception Interrupt condition is satisfied for Common FIFO Buffers when configured in GW mode or RX mode.

If the set from the CAN channel occurs simultaneously with the clear by the write access, then the bit is set. The bit is cleared by writing 1'b0 to it.

The bit is cleared when RS-CAN-FD module enters GL_RESET mode.

The bit is cleared when the related channel enters CH_RESET mode if the FIFO is configured in TX or GW mode.

This bit is not influenced by the value of **CFDCFCCd.CFIM**.

4.3.30.9 CFDCFSTSd.CFOFTXIF

Common FIFO One Frame Transmission Interrupt Flag

This bit will not be cleared automatically if the Common FIFO Buffer is disabled.

Users can only write to this bit when RS-CAN-FD module is in GL_HALT or GL_OPERATION mode and the related RS-CAN-FD channel is not in CH_RESET mode for FIFOs configured as TX or GW FIFO.

Writing 1'b1 has no influence on the bit values.

Do not use bit clear instruction for clearing this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Then the other bits remain set to 1'b1.

This bit is set automatically when the One Frame Transmission Interrupt condition is satisfied for Common FIFO Buffers configured in GW mode or TX mode.

If the set from the CAN channel occurs simultaneously with the clear by the write access, then the bit is set. The bit is cleared by writing 1'b0 to it.

The bit is cleared when RS-CAN-FD module enters GL_RESET mode.

The bit is cleared when the related channel enters CH_RESET mode if the FIFO is configured in TX or GW mode.

This bit is not influenced by the value of **CFDCFCCd.CFIM**.

4.3.30.10 CFDCFSTSd.CFMOW

Common FIFO Message overwrite

Users can only write to this bit when RS-CAN-FD module is in GL_HALT or GL_OPERATION mode and the related RS-CAN-FD channel is not in CH_RESET mode for FIFOs configured as TX or GW FIFO.

Writing 1'b1 has no influence on the bit values.

Do not use bit clear instruction for clearing this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Then the other bits remain set to 1'b1.

This bit is set automatically whenever a message is overwrite storage of a new message when

CFDCFCCEd.CFMOWM=1 and FIFO is already full in GW mode.

If the set from the CAN channel occurs simultaneously with the clear by the write access, then the bit is set.

The bit is cleared by writing 1'b0 to it.

The bit is cleared when RS-CAN-FD module enters GL_RESET mode.

The bit is cleared when the related channel enters CH_RESET mode if the FIFO is configured in TX or GW mode.

4.3.30.11 CFDCFSTSd.CFDMASTS

DMA Transfer Status for Common FIFO in FFI Mode

User can only access to this bit when **CFDGFFIMC.FFIEN** is 1'b1.

When **CFDGFFIMC.FFIEN** is 1'b0, the read value of this bit is 0

If Common FIFO 0 : This bit is DMA Transfer Status only.

If Common FIFO 1 : This bit is reserved.

If Common FIFO 2 : This bit is DMA TX Transfer Status only.

This bit is cleared when RS-CAN-FD module enters GL_RESET mode.

4.3.31 CFDCFPCTRd

Common FIFO Pointer Control Registers d

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
	-	-	-	-	-	-	-	-
Value after reset	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
	-	-	-	-	-	-	-	-
Value after reset	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
	-	-	-	-	-	-	-	-
Value after reset	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
	CFPC[7:0]							
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:8]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[7:0]	CFPC	Common FIFO Pointer Control	Increments read or write pointer of the corresponding Common FIFO Buffers depending upon the mode configuration	W

These registers can be used to increment the Read or Write Pointer of the corresponding Common FIFO.
(no_of_channels = 8)

(no_of_CFIFOs_per_channel = No. of Common FIFOs per Channel = 3)

Where the total number of CFIFOs = no_of_CFIFOs = no_of_CFIFOs_per_channel * no_of_channels = 3 * 8 = 24 as shown in Figure 8.1

(d = Common FIFO index = [0 .. no_of_CFIFOs -1])

4.3.31.1 CFDCFPCTRd.CFPC

Common FIFO Pointer Control

When the value 8'hFF is written into these bits, then the Read Pointer of the corresponding Common FIFO Buffer, when configured in RX mode, or the Write Pointer of the corresponding Common FIFO Buffer, when configured in TX mode, moves to the next FIFO entry.

Read value from these bits is always 8'h0.

Users can only write to these bits when RS-CAN-FD module is in GL_HALT or GL_OPERATION modes.

Users should only write 8'hFF to this register when the Common FIFO is enabled and is not empty if configured in RX mode.

Users should only write 8'hFF to this register when the Common FIFO is enabled and is not full if configured in TX mode.

Users should only write 8'hFF to this register when the Common FIFO is enabled and is not configured in GW mode.

Users should not write to the FIFO control registers when DMA is enabled.

4.3.32 CFDFESTS

FIFO Empty Status Register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
CFxEMP[23:16]								
Value after reset	1	1	1	1	1	1	1	1
CFxEMP[15:8]								
Value after reset	1	1	1	1	1	1	1	1
CFxEMP[7:0]								
Value after reset	1	1	1	1	1	1	1	1
RFxEMP[7:0]								
Value after reset	1	1	1	1	1	1	1	1

Bit	Symbol	Bit name	Function	R/W
b[31:8]	CFxEMP	Common FIFO Empty Status	1'b0: Corresponding FIFO not Empty 1'b1: Corresponding FIFO Empty	R
b[7:0]	RFxEMP	RX FIFO Empty Status	1'b0: Corresponding FIFO not Empty 1'b1: Corresponding FIFO Empty	R

The FIFO Empty status register bits show the status of the Empty bits of the FIFO Buffers.

4.3.32.1 CFDFESTS.RFxEMP (x = FIFO index = [0...a-1]) (a = No. of RX FIFOs = 8)

RX FIFO Empty Status

This bit is set when the RS-CAN-FD module enters GL_RESET mode.

Each bit is set automatically when the corresponding bit is set in the RX FIFO Status Registers.

Each bit is cleared automatically when the corresponding bit is cleared in the RX FIFO Status Registers.

4.3.32.2 CFDFESTS.CFxEMP (x = FIFO index = 0 to ((n+1)*3 - 1))

Common FIFO Empty Status

This bit is set when the RS-CAN-FD module enters GL_RESET mode.

Each bit is set automatically when the corresponding bit is set in the Common FIFO Status Registers.

Each bit is cleared automatically when the corresponding bit is cleared in the Common FIFO Status Registers.

4.3.33 CFDFSTS

FIFO Full Status Register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
CFxFLL[23:16]								
Value after reset	0	0	0	0	0	0	0	0
CFxFLL[15:8]								
Value after reset	0	0	0	0	0	0	0	0
CFxFLL[7:0]								
Value after reset	0	0	0	0	0	0	0	0
RFxFLL[7:0]								
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:8]	CFxFLL	Common FIFO Full Status	1'b0: Corresponding FIFO not Full 1'b1: Corresponding FIFO Full	R
b[7:0]	RFxFLL	RX FIFO Full Status	1'b0: Corresponding FIFO not Full 1'b1: Corresponding FIFO Full	R

The FIFO Full status register bits show the status of the Full bits of the FIFO Buffers.

4.3.33.1 CFDFSTS.RFxFLL (x = FIFO index = [0...a-1]) (a = No. of RX FIFOs = 8)

RX FIFO Full Status

Each bit is set automatically when the corresponding bit is set in the RX FIFO Status Registers.

Each bit is cleared automatically when the corresponding bit is cleared in the RX FIFO Status Registers.
This bit is cleared when RS-CAN-FD module enters GL_RESET mode.

4.3.33.2 CFDFSTS.CFxFLL (x = FIFO index = 0 to ((n+1)*3 -1))

Common FIFO Full Status

Each bit is set automatically when the corresponding bit is set in the Common FIFO Status Registers.

Each bit is cleared automatically when the corresponding bit is cleared in the Common FIFO Status Registers.

This bit is cleared when RS-CAN-FD module enters GL_RESET mode.

4.3.34 CFDFMSTS

FIFO Message Lost Status Register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
CFxMLT[23:16]								
Value after reset	0	0	0	0	0	0	0	0
CFxMLT[15:8]								
Value after reset	0	0	0	0	0	0	0	0
CFxMLT[7:0]								
Value after reset	0	0	0	0	0	0	0	0
RFxMLT[7:0]								
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:8]	CFxMLT	Common FIFO Msg Lost Status	1'b0: Corresponding FIFO Msg Lost flag not set 1'b1: Corresponding FIFO Msg Lost flag set	R
b[7:0]	RFxMLT	RX FIFO Msg Lost Status	1'b0: Corresponding FIFO Msg Lost flag not set 1'b1: Corresponding FIFO Msg Lost flag set	R

The FIFO Msg Lost status register bits show the status of the Msg Lost bits of the FIFO Buffers.

4.3.34.1 CFDFMSTS.RFxMLT (x = FIFO index = [0...a-1]) (a = No. of RX FIFOs = 8)

RX FIFO Msg Lost Status

This bit is cleared when RS-CAN-FD module enters GL_RESET mode.

Each bit is set automatically when the corresponding bit is set in the RX FIFO Status Registers.

Each bit is cleared automatically when the corresponding bit is cleared in the RX FIFO Status Registers.

4.3.34.2 CFDFMSTS.CFxMLT (x = FIFO index = 0 to ((n+1)*3 -1))

Common FIFO Msg Lost Status

This bit is cleared when RS-CAN-FD module enters GL_RESET mode.

Each bit is set automatically when the corresponding bit is set in the Common FIFO Status Registers.

Each bit is cleared automatically when the corresponding bit is cleared in the Common FIFO Status Registers.

4.3.35 CFDRFISTS

RX FIFO Interrupt Flag Status Register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
Value after reset	-	-	-	-	-	-	-	-
	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset	RFxFFLL[7:0]							
	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
Value after reset	-	-	-	-	-	-	-	-
	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset	RFxIF[7:0]							
	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:24]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[23:16]	RFxFFLL	RX FIFO[x] Interrupt Full Flag Status	1'b0: Corresponding RX FIFO interrupt Full flag not set 1'b1: Corresponding RX FIFO interrupt Full flag set	R
b[15:8]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[7:0]	RFxIF	RX FIFO[x] Interrupt Flag Status	1'b0: Corresponding RX FIFO interrupt flag not set 1'b1: Corresponding RX FIFO interrupt flag set	R

The FIFO Interrupt Flag status register bits show the status of the Interrupt Flag bits of the RX FIFO Buffers.

4.3.35.1 CFDRFISTS.RFxIF (x = FIFO index = [0...a-1]) (a = No. of RX FIFOs = 8)

RX FIFO[x] Interrupt Flag Status

Each bit is set automatically when the corresponding interrupt flag bit is set in the RX FIFO Status Registers.

This bit is cleared when RS-CAN-FD module enters GL_RESET mode.

Each bit is cleared automatically when the corresponding interrupt flag bit is cleared in the RX FIFO Status Registers.

4.3.35.2 CFDRFISTS.RFxFFLL (x = FIFO index = [0...a-1]) (a = No. of RX FIFOs = 8)

RX FIFO[x] Interrupt Full Flag Status

Each bit is set automatically when the corresponding interrupt Full flag bit is set in the RX FIFO Status Registers.

This bit is cleared when RS-CAN-FD module enters GL_RESET mode.

Each bit is cleared automatically when the corresponding interrupt Full flag bit is cleared in the RX FIFO Status Registers.

4.3.36 CFDCFRISTS

Common FIFO RX Interrupt Flag Status Register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
Value after reset	-	-	-	-	-	-	-	-
	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset	CFxRXIF[23:16]							
	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
Value after reset	CFxRXIF[15:8]							
	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset	CFxRXIF[7:0]							
	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:24]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[23:0]	CFxRXIF	Common FIFO [x] RX Interrupt Flag Status	1'b0: Corresponding Common FIFO RX interrupt flag is not set 1'b1: Corresponding Common FIFO RX interrupt flag is set	R

The FIFO Interrupt Flag status register bits show the status of the Interrupt Flag bits of the Common FIFO Buffers.

4.3.36.1 CFDCFRISTS.CFxRXIF (x = FIFO index = 0 to ((n+1)*3 -1))

Common FIFO [x] RX Interrupt Flag Status

Each bit is set automatically when the corresponding RX interrupt flag bit is set in the Common FIFO Status Registers.

This bit is cleared when RS-CAN-FD module enters GL_RESET mode.

Each bit is cleared automatically when the corresponding RX interrupt flag bit is cleared in the Common FIFO Status Registers.

4.3.37 CFDCFTISTS

Common FIFO TX Interrupt Flag Status Register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
Value after reset	-	-	-	-	-	-	-	-
	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset	0	0	0	0	0	0	0	0
	CFxTXIF[23:16]							
	b15	b14	b13	b12	b11	b10	b9	b8
Value after reset	0	0	0	0	0	0	0	0
	CFxTXIF[15:8]							
	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset	0	0	0	0	0	0	0	0
	CFxTXIF[7:0]							

Bit	Symbol	Bit name	Function	R/W
b[31:24]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[23:0]	CFxTXIF	Common FIFO [x] TX Interrupt Flag Status	1'b0: Corresponding Common FIFO TX interrupt flag is not set 1'b1: Corresponding Common FIFO TX interrupt flag is set	R

The FIFO Interrupt Flag status register bits show the status of the Interrupt Flag bits of the Common FIFO Buffers.

4.3.37.1 CFDCFTISTS.CFxTXIF (x = FIFO index = 0 to ((n+1)*3 - 1))

Common FIFO [x] TX Interrupt Flag Status

Each bit is set automatically when the corresponding TX interrupt flag bit is set in the Common FIFO Status Registers.

This bit is cleared when RS-CAN-FD module enters GL_RESET mode.

Each bit is cleared automatically when the corresponding TX interrupt flag bit is cleared in the Common FIFO Status Registers.

4.3.38 CFDFFFSTS

FIFO FDC level Full Status Register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
CFxFFLL[23:16]								
Value after reset	0	0	0	0	0	0	0	0
CFxFFLL[15:8]								
Value after reset	0	0	0	0	0	0	0	0
CFxFFLL[7:0]								
Value after reset	0	0	0	0	0	0	0	0
RFxFFLL [7:0]								
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:8]	CFxFFLL	COMMON FIFO FDC level full Status	1'b0: Corresponding FIFO Full interrupt not set 1'b1: Corresponding FIFO Full interrupt is set	R
b[7:0]	RFxFFLL	RX FIFO FDC level full Status	1'b0: Corresponding FIFO Full interrupt not set 1'b1: Corresponding FIFO Full interrupt is set	R

The FIFO Full status register bits show the status of the Full Interrupt Flag bits of the FIFO Buffers.

4.3.38.1 CFDFFFSTS.RFxFFLL (x = FIFO index = [0...a-1]) (a = No. of RX FIFOs = 8)

RX FIFO FDC level Full Status

Each bit is set automatically when the corresponding bit is set in the RX FIFO Status Registers.

Each bit is cleared automatically when the corresponding bit is cleared in the RX FIFO Status Registers.
This bit is cleared when RS-CAN-FD module enters GL_RESET mode.

4.3.38.2 CFDFFFSTS.CFxFFLL (x = FIFO index = 0 to ((n+1)*3 - 1))

Common FIFO FDC level Full Status

Each bit is set automatically when the corresponding bit is set in the Common FIFO Status Registers.

Each bit is cleared automatically when the corresponding bit is cleared in the Common FIFO Status Registers.

This bit is cleared when RS-CAN-FD module enters GL_RESET mode.

4.3.39 CFDCFMOWSTS

Common FIFO Message OverWrite Status

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
Value after reset	-	-	-	-	-	-	-	-
	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset	0	0	0	0	0	0	0	0
	CFxMOW [23:16]							
	b15	b14	b13	b12	b11	b10	b9	b8
Value after reset	0	0	0	0	0	0	0	0
	CFxMOW [15:8]							
	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset	0	0	0	0	0	0	0	0
	CFxMOW [7:0]							

Bit	Symbol	Bit name	Function	R/W
b[31:24]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[23:0]	CFxMOW	Common FIFO [x] Message overwrite status	1'b0: Corresponding FIFO overwrite flag is not set 1'b1: Corresponding FIFO overwrite flag is set	R

4.3.39.1 CFDCFMOWSTS.CFxMOW (x = FIFO index = 0 to ((n+1)*3 - 1))

Common FIFO Message OverWrite Status

This bit is cleared when RS-CAN-FD module enters GL_RESET mode.

Each bit is set automatically when the corresponding bit is set in the Common FIFO Status Registers.

Each bit is cleared automatically when the corresponding bit is cleared in the Common FIFO Status Registers.

This register is effective only in GW mode.

4.3.40 CFDCFOFRISTS

Common FIFO One Frame RX Interrupt Flag Status

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
Value after reset	-	-	-	-	-	-	-	-
	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset	0	0	0	0	0	0	0	0
	CFxOFRXIF[23:16]							
	b15	b14	b13	b12	b11	b10	b9	b8
Value after reset	0	0	0	0	0	0	0	0
	CFxOFRXIF[15:8]							
	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset	0	0	0	0	0	0	0	0
	CFxOFRXIF[7:0]							

Bit	Symbol	Bit name	Function	R/W
b[31:24]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[23:0]	CFxOFRXIF	Common FIFO [x] One Frame RX Interrupt Flag Status	1'b0: Corresponding Common FIFO One Frame RX interrupt flag is not set 1'b1: Corresponding Common FIFO One Frame RX interrupt flag is set	R

The FIFO One Frame RX Interrupt Flag status register bits show the status of the Interrupt Flag bits of the Common FIFO Buffers.

4.3.40.1 CFDCFOFRISTS.CFxOFRXIF (x = FIFO index = 0 to ((n+1)*3 -1))

Common FIFO [x] One Frame RX Interrupt Flag Status

Each bit is set automatically when the corresponding One Frame RX interrupt flag bit is set in the Common FIFO Status Registers.

This bit is cleared when RS-CAN-FD module enters GL_RESET mode.

Each bit is cleared automatically when the corresponding One Frame RX interrupt flag bit is cleared in the Common FIFO Status Registers.

4.3.41 CFDCFOFTISTS

Common FIFO One Frame TX Interrupt Flag Status

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
Value after reset	-	-	-	-	-	-	-	-
	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset	0	0	0	0	0	0	0	0
	CFxOFTXIF[23:16]							
	b15	b14	b13	b12	b11	b10	b9	b8
Value after reset	0	0	0	0	0	0	0	0
	CFxOFTXIF[15:8]							
	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset	0	0	0	0	0	0	0	0
	CFxOFTXIF[7:0]							

Bit	Symbol	Bit name	Function	R/W
b[31:24]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[23:0]	CFxOFTXIF	Common FIFO [x] One Frame TX Interrupt Flag Status	1'b0: Corresponding Common FIFO One Frame TX interrupt flag is not set 1'b1: Corresponding Common FIFO One Frame TX interrupt flag is set	R

The FIFO One Frame TX Interrupt Flag status register bits show the status of the Interrupt Flag bits of the Common FIFO Buffers.

4.3.41.1 CFDCFOFTISTS.CFxOFTXIF (x = FIFO index = 0 to ((n+1)*3 - 1))

Common FIFO [x] One Frame TX Interrupt Flag Status

Each bit is set automatically when the corresponding One Frame TX interrupt flag bit is set in the Common FIFO Status Registers.

This bit is cleared when RS-CAN-FD module enters GL_RESET mode.

Each bit is cleared automatically when the corresponding One Frame TX interrupt flag bit is cleared in the Common FIFO Status Registers.

4.3.42 CFDCDTCT

DMA Transfer Control

Register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24	
Value after reset	-	-	-	-	-	-	-	-	
	b23	b22	b21	b20	b19	b18	b17	b16	
Value after reset	-	-	-	-	-	-	-	-	
	b15	b14	b13	b12	b11	b10	b9	b8	
Value after reset	CFDMAE7	CFDMAE6	CFDMAE5	CFDMAE4	CFDMAE3	CFDMAE2	CFDMAE1	CFDMAE0	
	0	0	0	0	0	0	0	0	
	b7	b6	b5	b4	b3	b2	b1	b0	
Value after reset	RFDMAE7	RFDMAE6	RFDMAE5	RFDMAE4	RFDMAE3	RFDMAE2	RFDMAE1	RFDMAE0	
Bit	Symbol	Bit name	Function						R/W
b[31:16]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[15]	CFDMAE7	DMA Transfer Enable for Common FIFO 0 of channel 7	1'b0: DMA Transfer Request disabled for channel n 1'b1: DMA Transfer Request enabled for channel n						R/W
b[14]	CFDMAE6	DMA Transfer Enable for Common FIFO 0 of channel 6	1'b0: DMA Transfer Request disabled for channel n 1'b1: DMA Transfer Request enabled for channel n						R/W
b[13]	CFDMAE5	DMA Transfer Enable for Common FIFO 0 of channel 5	1'b0: DMA Transfer Request disabled for channel n 1'b1: DMA Transfer Request enabled for channel n						R/W
b[12]	CFDMAE4	DMA Transfer Enable for Common FIFO 0 of channel 4	1'b0: DMA Transfer Request disabled for channel n 1'b1: DMA Transfer Request enabled for channel n						R/W
b[11]	CFDMAE3	DMA Transfer Enable for Common FIFO 0 of channel 3	1'b0: DMA Transfer Request disabled for channel n 1'b1: DMA Transfer Request enabled for channel n						R/W
b[10]	CFDMAE2	DMA Transfer Enable for Common FIFO 0 of channel 2	1'b0: DMA Transfer Request disabled for channel n 1'b1: DMA Transfer Request enabled for channel n						R/W
b[9]	CFDMAE1	DMA Transfer Enable for Common FIFO 0 of channel 1	1'b0: DMA Transfer Request disabled for channel n 1'b1: DMA Transfer Request enabled for channel n						R/W
b[8]	CFDMAE0	DMA Transfer Enable for Common FIFO 0 of channel 0	1'b0: DMA Transfer Request disabled for channel n 1'b1: DMA Transfer Request enabled for channel n						R/W
b[7]	RFDMAE7	DMA Transfer Enable for RXFIFO 7	1'b0: DMA Transfer Request disabled 1'b1: DMA Transfer Request enabled						R/W
b[6]	RFDMAE6	DMA Transfer Enable for RXFIFO 6	1'b0: DMA Transfer Request disabled 1'b1: DMA Transfer Request enabled						R/W
b[5]	RFDMAE5	DMA Transfer Enable for RXFIFO 5	1'b0: DMA Transfer Request disabled 1'b1: DMA Transfer Request enabled						R/W
b[4]	RFDMAE4	DMA Transfer Enable for RXFIFO 4	1'b0: DMA Transfer Request disabled 1'b1: DMA Transfer Request enabled						R/W
b[3]	RFDMAE3	DMA Transfer Enable for RXFIFO 3	1'b0: DMA Transfer Request disabled 1'b1: DMA Transfer Request enabled						R/W
b[2]	RFDMAE2	DMA Transfer Enable for RXFIFO 2	1'b0: DMA Transfer Request disabled 1'b1: DMA Transfer Request enabled						R/W
b[1]	RFDMAE1	DMA Transfer Enable for RXFIFO 1	1'b0: DMA Transfer Request disabled 1'b1: DMA Transfer Request enabled						R/W
b[0]	RFDMAE0		1'b0: DMA Transfer Request disabled						R/W

DMA Transfer Enable for
RXFIFO 0

1'b1: DMA Transfer Request enabled

The DMA Transfer Control Register bits control the start and stop of the DMA transfer operation.

4.3.42.1 CFDCDTCT.RFDMAEe

DMA Transfer Enable for RXFIFO e

(e = DMA Transfer Enable FIFO index = [0... No. of RX FIFOs-1]) (No. of RX FIFOs = 8)

This bit cannot be set in GL_SLEEP or GL_RESET mode.

This bit is cleared when RS-CAN-FD module enters GL_RESET mode.

This bit is a reserve bit when **CFDGFFIMC.FFIEN** is 1'b1.

When reserve bit, this bit is always read as 0.

4.3.42.2 CFDCDTCT.CFDMAEn

DMA Transfer Enable for Common FIFO 0 of channel n

By this bit only Common FIFO0 can be linked to a DMA channel, to link Common FIFO2 see **CFDCDTTCT.CFDMAEn**.

Common FIFO1 cannot be linked to a DMA channel.

This bit cannot be set in GL_SLEEP or GL_RESET mode.

Users should not enable a DMA transfer for a common FIFO that is configured as TX or GW FIFO.

This bit is cleared when RS-CAN-FD module enters GL_RESET mode.

This bit is a reserve bit when **CFDGFFIMC.FFIEN** is 1'b1.

When reserve bit, this bit is always read as 0.

4.3.43 CFDCDTSTS

DMA Transfer Status Register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24	
Value after reset	-	-	-	-	-	-	-	-	
Value after reset	0	0	0	0	0	0	0	0	
	b23	b22	b21	b20	b19	b18	b17	b16	
Value after reset	-	-	-	-	-	-	-	-	
Value after reset	0	0	0	0	0	0	0	0	
	b15	b14	b13	b12	b11	b10	b9	b8	
Value after reset	CFDMASTS 7	CFDMAST S6	CFDMAST S5	CFDMAST S4	CFDMAST S3	CFDMAST S2	CFDMAST S1	CFDMAST S0	
Value after reset	0	0	0	0	0	0	0	0	
	b7	b6	b5	b4	b3	b2	b1	b0	
Value after reset	RFDMASTS 7	RFDMAST S6	RFDMAST S5	RFDMAST S4	RFDMAST S3	RFDMAST S2	RFDMAST S1	RFDMAST S0	
Value after reset	0	0	0	0	0	0	0	0	
Bit	Symbol	Bit name	Function						R/W
b[31:16]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[15]	CFDMA STS7	DMA Transfer Status only for Common FIFO 0 of channel 7	1'b0: DMA transfer stopped 1'b1: DMA transfer ongoing						R
b[14]	CFDMA STS6	DMA Transfer Status only for Common FIFO 0 of channel 6	1'b0: DMA transfer stopped 1'b1: DMA transfer ongoing						R
b[13]	CFDMA STS5	DMA Transfer Status only for Common FIFO 0 of channel 5	1'b0: DMA transfer stopped 1'b1: DMA transfer ongoing						R
b[12]	CFDMA STS4	DMA Transfer Status only for Common FIFO 0 of channel 4	1'b0: DMA transfer stopped 1'b1: DMA transfer ongoing						R
b[11]	CFDMA STS3	DMA Transfer Status only for Common FIFO 0 of channel 3	1'b0: DMA transfer stopped 1'b1: DMA transfer ongoing						R
b[10]	CFDMA STS2	DMA Transfer Status only for Common FIFO 0 of channel 2	1'b0: DMA transfer stopped 1'b1: DMA transfer ongoing						R
b[9]	CFDMA STS1	DMA Transfer Status only for Common FIFO 0 of channel 1	1'b0: DMA transfer stopped 1'b1: DMA transfer ongoing						R
b[8]	CFDMA STS0	DMA Transfer Status only for Common FIFO 0 of channel 0	1'b0: DMA transfer stopped 1'b1: DMA transfer ongoing						R
b[7]	RFDMA STS7	DMA Transfer Status for RX FIFO 7	1'b0: DMA transfer stopped 1'b1: DMA transfer ongoing						R
b[6]	RFDMA STS6	DMA Transfer Status for RX FIFO 6	1'b0: DMA transfer stopped 1'b1: DMA transfer ongoing						R
b[5]	RFDMA STS5	DMA Transfer Status for RX FIFO 5	1'b0: DMA transfer stopped 1'b1: DMA transfer ongoing						R

b[4]	RFDMA STS4	DMA Transfer Status for RX FIFO 4	1'b0: DMA transfer stopped 1'b1: DMA transfer ongoing	R
b[3]	RFDMA STS3	DMA Transfer Status for RX FIFO 3	1'b0: DMA transfer stopped 1'b1: DMA transfer ongoing	R
b[2]	RFDMA STS2	DMA Transfer Status for RX FIFO 2	1'b0: DMA transfer stopped 1'b1: DMA transfer ongoing	R
b[1]	RFDMA STS1	DMA Transfer Status for RX FIFO 1	1'b0: DMA transfer stopped 1'b1: DMA transfer ongoing	R
b[0]	RFDMA STS0	DMA Transfer Status for RX FIFO 0	1'b0: DMA transfer stopped 1'b1: DMA transfer ongoing	R

The DMA Transfer Status Register bits show the status of the DMA transfer.

4.3.43.1 CFDCDTSTS.RFDMASTSe

DMA Transfer Status for RX FIFO e

(e = DMA Transfer Enable FIFO index = [0... No. of RX FIFOs-1]) (No. of RX FIFOs = 8)

Each bit is set automatically when the corresponding DMA enable bit is set and the corresponding DMA FIFO is not empty.

Each bit is cleared automatically when the DMA transfer stops either because the DMA is disabled or the DMA FIFO is empty.

When **CFDCDTCT.RFDMAEe** is set to 1'b0 while DMA transfer for the corresponding FIFO is on going, **CFDCDTSTS.RFDMASTSe** becomes 1'b0 when the DMA transfer is completed.

This bit is cleared when RS-CAN-FD module enters GL_RESET mode.

4.3.43.2 CFDCDTSTS.CFDMASTSn

DMA Transfer Status only for Common FIFO 0 of channel n

Each bit is set automatically when the corresponding DMA enable bit is set and the corresponding DMA FIFO is not empty.

Each bit is cleared automatically when the DMA transfer stops either because the DMA is disabled or the DMA FIFO is empty.

When **CFDCDTCT.CFDMAEn** is set to 1'b0 while DMA transfer for the corresponding FIFO is on going, **CFDCDTSTS.CFDMASTSn** becomes 1'b0 when the DMA transfer is completed.

This bit is cleared when RS-CAN-FD module enters GL_RESET mode.

4.3.44 CFDCDTTCT

DMA TX Transfer Control Register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
Value after reset	-	-	-	-	-	-	-	-
	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset	CFDMA E7	CFDMA E6	CFDMA E5	CFDMA E4	CFDMA E3	CFDMA E2	CFDMA E1	CFDMA E0
	b15	b14	b13	b12	b11	b10	b9	b8
Value after reset	TQ3DMA E7	TQ3DMA E6	TQ3DMA E5	TQ3DMA E4	TQ3DMA E3	TQ3DMA E2	TQ3DMA E1	TQ3DMA E0
	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset	TQ0DMA E7	TQ0DMA E6	TQ0DMA E5	TQ0DMA E4	TQ0DMA E3	TQ0DMA E2	TQ0DMA E1	TQ0DMA E0
Bit	Symbol	Bit name	Function					
b[31:24]	-	Reserved	These bits are read as 0. The write value should be always 0.					
b[23]	CFDMAE7	DMA TX Transfer Enable for Common FIFO 2 of channel 7	1'b0: DMA TX Transfer Request disabled for channel n 1'b1: DMA TX Transfer Request enabled for channel n					
b[22]	CFDMAE6	DMA TX Transfer Enable for Common FIFO 2 of channel 6	1'b0: DMA TX Transfer Request disabled for channel n 1'b1: DMA TX Transfer Request enabled for channel n					
b[21]	CFDMAE5	DMA TX Transfer Enable for Common FIFO 2 of channel 5	1'b0: DMA TX Transfer Request disabled for channel n 1'b1: DMA TX Transfer Request enabled for channel n					
b[20]	CFDMAE4	DMA TX Transfer Enable for Common FIFO 2 of channel 4	1'b0: DMA TX Transfer Request disabled for channel n 1'b1: DMA TX Transfer Request enabled for channel n					
b[19]	CFDMAE3	DMA TX Transfer Enable for Common FIFO 2 of channel 3	1'b0: DMA TX Transfer Request disabled for channel n 1'b1: DMA TX Transfer Request enabled for channel n					
b[18]	CFDMAE2	DMA TX Transfer Enable for Common FIFO 2 of channel 2	1'b0: DMA TX Transfer Request disabled for channel n 1'b1: DMA TX Transfer Request enabled for channel n					
b[17]	CFDMAE1	DMA TX Transfer Enable for Common FIFO 2 of channel 1	1'b0: DMA TX Transfer Request disabled for channel n 1'b1: DMA TX Transfer Request enabled for channel n					
b[16]	CFDMAE0	DMA TX Transfer Enable for Common FIFO 2 of channel 0	1'b0: DMA TX Transfer Request disabled for channel n 1'b1: DMA TX Transfer Request enabled for channel n					
b[15]	TQ3DMAE7	DMA TX Transfer Enable for TXQ 3 of channel 7	1'b0: DMA TX Transfer Request disabled 1'b1: DMA TX Transfer Request enabled					
b[14]	TQ3DMAE6	DMA TX Transfer Enable for TXQ 3 of channel 6	1'b0: DMA TX Transfer Request disabled 1'b1: DMA TX Transfer Request enabled					
b[13]	TQ3DMAE5	DMA TX Transfer Enable for TXQ 3 of channel 5	1'b0: DMA TX Transfer Request disabled 1'b1: DMA TX Transfer Request enabled					
b[12]	TQ3DMAE4	DMA TX Transfer Enable for TXQ 3 of channel 4	1'b0: DMA TX Transfer Request disabled 1'b1: DMA TX Transfer Request enabled					
b[11]	TQ3DMAE3	DMA TX Transfer Enable for TXQ 3 of channel 3	1'b0: DMA TX Transfer Request disabled 1'b1: DMA TX Transfer Request enabled					
b[10]	TQ3DMAE2	DMA TX Transfer Enable for TXQ 3 of channel 2	1'b0: DMA TX Transfer Request disabled 1'b1: DMA TX Transfer Request enabled					
b[9]	TQ3DMAE1	DMA TX Transfer Enable for TXQ 3 of channel 1	1'b0: DMA TX Transfer Request disabled 1'b1: DMA TX Transfer Request enabled					

b[8]	TQ3DMAE0	DMA TX Transfer Enable for TXQ 3 of channel 0	1'b0: DMA TX Transfer Request disabled 1'b1: DMA TX Transfer Request enabled	R/W
b[7]	TQ0DMAE7	DMA TX Transfer Enable for TXQ 0 of channel 7	1'b0: DMA TX Transfer Request disabled 1'b1: DMA TX Transfer Request enabled	R/W
b[6]	TQ0DMAE6	DMA TX Transfer Enable for TXQ 0 of channel 6	1'b0: DMA TX Transfer Request disabled 1'b1: DMA TX Transfer Request enabled	R/W
b[5]	TQ0DMAE5	DMA TX Transfer Enable for TXQ 0 of channel 5	1'b0: DMA TX Transfer Request disabled 1'b1: DMA TX Transfer Request enabled	R/W
b[4]	TQ0DMAE4	DMA TX Transfer Enable for TXQ 0 of channel 4	1'b0: DMA TX Transfer Request disabled 1'b1: DMA TX Transfer Request enabled	R/W
b[3]	TQ0DMAE3	DMA TX Transfer Enable for TXQ 0 of channel 3	1'b0: DMA TX Transfer Request disabled 1'b1: DMA TX Transfer Request enabled	R/W
b[2]	TQ0DMAE2	DMA TX Transfer Enable for TXQ 0 of channel 2	1'b0: DMA TX Transfer Request disabled 1'b1: DMA TX Transfer Request enabled	R/W
b[1]	TQ0DMAE1	DMA TX Transfer Enable for TXQ 0 of channel 1	1'b0: DMA TX Transfer Request disabled 1'b1: DMA TX Transfer Request enabled	R/W
b[0]	TQ0DMAE0	DMA TX Transfer Enable for TXQ 0 of channel 0	1'b0: DMA TX Transfer Request disabled 1'b1: DMA TX Transfer Request enabled	R/W

The DMA TX Transfer Control Register bits control the start and stop of the DMA transfer operation.

4.3.44.1 CFDCDTTCT.TQ0DMAEn

DMA TX Transfer Enable for TXQ 0 of channel n

This bit cannot be set in GL_SLEEP or GL_RESET mode.

This bit is cleared when RS-CAN-FD module enters GL_RESET mode.

This bit is a reserve bit when **CFDGFFIMC.FFIEN** is 1'b1.

When reserve bit, this bit is always read as 0.

4.3.44.2 CFDCDTTCT.TQ3DMAEn

DMA TX Transfer Enable for TXQ 3 of channel n

This bit cannot be set in GL_SLEEP or GL_RESET mode.

This bit is cleared when RS-CAN-FD module enters GL_RESET mode.

This bit is a reserve bit when **CFDGFFIMC.FFIEN** is 1'b1.

When reserve bit, this bit is always read as 0.

4.3.44.3 CFDCDTTCT.CFDMAEn

DMA TX Transfer Enable for Common FIFO 2 of channel n

By this bit only Common FIFO2 can be linked to a DMA channel, to link Common FIFO0 see **CFDCDTCT.CFDMAEn**.

Common FIFO1 cannot be linked to a DMA channel.

This bit cannot be set in GL_SLEEP or GL_RESET mode.

Users should not enable a DMA transfer for a common FIFO that is configured as RX or GW FIFO.

This bit is cleared when RS-CAN-FD module enters GL_RESET mode.

This bit is a reserve bit when **CFDGFFIMC.FFIEN** is 1'b1.

When reserve bit, this bit is always read as 0.

4.3.45 CFDCDTTSTS

DMA TX Transfer Status Register

Address: Refer to section
4.1

	b31	b30	b29	b28	b27	b26	b25	b24	
Value after reset	-	-	-	-	-	-	-	-	
	b23	b22	b21	b20	b19	b18	b17	b16	
Value after reset	CFDMASTS7	CFDMASTS6	CFDMASTS5	CFDMASTS4	CFDMASTS3	CFDMASTS2	CFDMASTS1	CFDMASTS0	
	b15	b14	b13	b12	b11	b10	b9	b8	
Value after reset	TQ3DMASTS7	TQ3DMAS TS6	TQ3DMAS TS5	TQ3DMAS TS4	TQ3DMAS TS3	TQ3DMAS TS2	TQ3DMAS TS1	TQ3DMAS TS0	
	b7	b6	b5	b4	b3	b2	b1	b0	
Value after reset	TQ0DMASTS7	TQ0DMAS TS6	TQ0DMAS TS5	TQ0DMAS TS4	TQ0DMAS TS3	TQ0DMAS TS2	TQ0DMAS TS1	TQ0DMAS TS0	
Bit	Symbol	Bit name	Function						R/W
b[31:24]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[23]	CFDMASTS7	DMA TX Transfer Status only for Common FIFO 2 of channel 7	1'b0: DMA transfer stopped 1'b1: DMA transfer enable						R
b[22]	CFDMASTS6	DMA TX Transfer Status only for Common FIFO 2 of channel 6	1'b0: DMA transfer stopped 1'b1: DMA transfer enable						R
b[21]	CFDMASTS5	DMA TX Transfer Status only for Common FIFO 2 of channel 5	1'b0: DMA transfer stopped 1'b1: DMA transfer enable						R
b[20]	CFDMASTS4	DMA TX Transfer Status only for Common FIFO 2 of channel 4	1'b0: DMA transfer stopped 1'b1: DMA transfer enable						R
b[19]	CFDMASTS3	DMA TX Transfer Status only for Common FIFO 2 of channel 3	1'b0: DMA transfer stopped 1'b1: DMA transfer enable						R
b[18]	CFDMASTS2	DMA TX Transfer Status only for Common FIFO 2 of channel 2	1'b0: DMA transfer stopped 1'b1: DMA transfer enable						R
b[17]	CFDMASTS1	DMA TX Transfer Status only for Common FIFO 2 of channel 1	1'b0: DMA transfer stopped 1'b1: DMA transfer enable						R
b[16]	CFDMASTS0	DMA TX Transfer Status only for Common FIFO 2 of channel 0	1'b0: DMA transfer stopped 1'b1: DMA transfer enable						R
b[15]	TQ3DMASTS7	DMA TX Transfer Status for TXQ3 of channel 7	1'b0: DMA transfer stopped 1'b1: DMA transfer enable						R
b[14]	TQ3DMASTS6	DMA TX Transfer Status for TXQ3 of channel 6	1'b0: DMA transfer stopped 1'b1: DMA transfer enable						R
b[13]			1'b0: DMA transfer stopped						R

	TQ3DM ASTS5	DMA TX Transfer Status for TXQ3 of channel 5	1'b1: DMA transfer enable	
b[12]	TQ3DM ASTS4	DMA TX Transfer Status for TXQ3 of channel 4	1'b0: DMA transfer stopped 1'b1: DMA transfer enable	R
b[11]	TQ3DM ASTS3	DMA TX Transfer Status for TXQ3 of channel 3	1'b0: DMA transfer stopped 1'b1: DMA transfer enable	R
b[10]	TQ3DM ASTS2	DMA TX Transfer Status for TXQ3 of channel 2	1'b0: DMA transfer stopped 1'b1: DMA transfer enable	R
b[9]	TQ3DM ASTS1	DMA TX Transfer Status for TXQ3 of channel 1	1'b0: DMA transfer stopped 1'b1: DMA transfer enable	R
b[8]	TQ3DM ASTS0	DMA TX Transfer Status for TXQ3 of channel 0	1'b0: DMA transfer stopped 1'b1: DMA transfer enable	R
b[7]	TQ0DM ASTS7	DMA TX Transfer Status for TXQ0 of channel 7	1'b0: DMA transfer stopped 1'b1: DMA transfer enable	R
b[6]	TQ0DM ASTS6	DMA TX Transfer Status for TXQ0 of channel 6	1'b0: DMA transfer stopped 1'b1: DMA transfer enable	R
b[5]	TQ0DM ASTS5	DMA TX Transfer Status for TXQ0 of channel 5	1'b0: DMA transfer stopped 1'b1: DMA transfer enable	R
b[4]	TQ0DM ASTS4	DMA TX Transfer Status for TXQ0 of channel 4	1'b0: DMA transfer stopped 1'b1: DMA transfer enable	R
b[3]	TQ0DM ASTS3	DMA TX Transfer Status for TXQ0 of channel 3	1'b0: DMA transfer stopped 1'b1: DMA transfer enable	R
b[2]	TQ0DM ASTS2	DMA TX Transfer Status for TXQ0 of channel 2	1'b0: DMA transfer stopped 1'b1: DMA transfer enable	R
b[1]	TQ0DM ASTS1	DMA TX Transfer Status for TXQ0 of channel 1	1'b0: DMA transfer stopped 1'b1: DMA transfer enable	R
b[0]	TQ0DM ASTS0	DMA TX Transfer Status for TXQ0 of channel 0	1'b0: DMA transfer stopped 1'b1: DMA transfer enable	R

The DMA Transfer Status Register bits show the status of the DMA transfer.

4.3.45.1 CFDCDTTSTS.TQ0DMASTS_n

DMA TX Transfer Status for TXQ0 of channel n

This bit is set when the **CFDCDTTCT.TQ0DMAEn** bit in the corresponding **CFDCDTTCT** register is set.

This bit is cleared when the **CFDCDTTCT.TQ0DMAEn** bit in the corresponding **CFDCDTTCT** register is cleared.

This bit is cleared when RS-CAN-FD module enters GL_RESET mode.

4.3.45.2 CFDCDTTSTS.TQ3DMASTS_n

DMA TX Transfer Status for TXQ3 of channel n

This bit is set when the **CFDCDTTCT.TQ3DMAEn** bit in the corresponding **CFDCDTTCT** register is set.

This bit is cleared when the **CFDCDTTCT.TQ3DMAEn** bit in the corresponding **CFDCDTTCT** register is cleared.

This bit is cleared when RS-CAN-FD module enters GL_RESET mode.

4.3.45.3 CFDCDTTSTS.CFDMASTS_n

DMA TX Transfer Status only for Common FIFO 2 of channel n

This bit is set when the **CFDCDTTCT.CFDMAEn** bit in the corresponding **CFDCDTTCT** register is set.

This bit is cleared when the **CFDCDTTCT.CFDMAEn** bit in the corresponding **CFDCDTTCT** register is cleared.

This bit is cleared when RS-CAN-FD module enters GL_RESET mode.

4.3.46 CFDGRINTSTS_n

Global RX Interrupt Status Register

Address: Refer to section 4.1

		b31	b30	b29	b28	b27	b26	b25	b24
		CFOFRIF[2:0]				CFRFIF[2:0]			
Value after reset		0	0	0	0	0	0	0	0
Value after reset		0	0	0	0	0	0	0	0
Value after reset		0	0	0	0	0	0	0	0
Value after reset		0	0	0	0	0	0	0	0
Bit	Symbol	Bit name	Function						R/W
b[31]	-	Reserved	This bit is read as 0b. The write value should be always 0b.						R/W
b[30:28]	CFOFRIF	Common FIFO One Frame RX Interrupt Flag Channel n	1'b0: Corresponding Common FIFO One Frame RX interrupt flag is not set 1'b1: Corresponding Common FIFO One Frame RX interrupt flag is set						R
b[27]	-	Reserved	This bit is read as 0b. The write value should be always 0b.						R/W
b[26:24]	CFRFIF	Common FIFO FDC level Full Interrupt Flag Channel n	1'b0: Corresponding Common FIFO Full interrupt flag is not set 1'b1: Corresponding Common FIFO Full interrupt flag is set						R
b[23:19]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[18:16]	CFRIF	Common FIFO RX Interrupt Flag Channel n	1'b0: Corresponding Common FIFO RX interrupt flag is not set 1'b1: Corresponding Common FIFO RX interrupt flag is set						R
b[15:14]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[13:12]	BQOFRIF	Borrowed TXQ One Frame RX Interrupt Flag Channel n	1'b0: Corresponding TXQ One Frame RX interrupt flag is not set 1'b1: Corresponding TXQ One Frame RX interrupt flag is set						R
b[11]	-	Reserved	This bit is read as 0b. The write value should be always 0b.						R/W
b[10:8]	QOFRIF	TXQ One Frame RX Interrupt Flag Channel n	1'b0: Corresponding TXQ One Frame RX interrupt flag is not set 1'b1: Corresponding TXQ One Frame RX interrupt flag is set						R
b[7:6]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[5:4]	BQFIF	Borrowed TXQ Full Interrupt Flag Channel n	1'b0: Corresponding TXQ Full interrupt flag is not set 1'b1: Corresponding TXQ Full interrupt flag is set						R

b[3]	-	Reserved	This bit is read as 0b. The write value should be always 0b.	R/W
b[2:0]	QFIF	TXQ Full Interrupt Flag Channel n	1'b0: Corresponding TXQ Full interrupt flag is not set	R
			1'b1: Corresponding TXQ Full interrupt flag is set	

4.3.46.1 CFDGRINTSTS_n.QFIF

TXQ Full Interrupt Flag Channel n

This bit is set automatically when the TXQ Full Interrupt flag of the related channel is set when the Interrupt is enabled.

This bit is cleared automatically when related TXQ Result status bits are cleared or the Interrupt enable is disabled.

This bit will be cleared automatically in GL_RESET mode or CH_RESET mode.

This bit is a reserve bit when **CFDGFFIMC.FFIEN** is 1'b1.

When reserve bit, this bit is always read as 0.

4.3.46.2 CFDGRINTSTS_n.BQFIF

Borrowed TXQ Full Interrupt Flag Channel n

This bit is set, when a Flexible transmission buffer assignment function is used and borrowed TXQ is in full status. Operation is the same as **CFDGRINTSTS_n.QFIF**.

This bit of the channel which lends TXMB is a reserve bit.

This bit is a reserve bit when **CFDGFFIMC.FFIEN** is 1'b1.

When reserve bit, this bit is always read as 0.

4.3.46.3 CFDGRINTSTS_n.QOFRIF

TXQ One Frame RX Interrupt Flag Channel n

This bit is set automatically when the TXQ One Frame RX Interrupt flag of the related channel is set when the Interrupt is enabled.

This bit is cleared automatically when related TXQ Result status bits are cleared or the Interrupt enable is disabled.

This bit will be cleared automatically in GL_RESET mode or CH_RESET mode.

This bit is a reserve bit when **CFDGFFIMC.FFIEN** is 1'b1.

When reserve bit, this bit is always read as 0.

4.3.46.4 CFDGRINTSTS_n.BQOFRIF

Borrowed TXQ One Frame RX Interrupt Flag Channel n

This bit is set, when a Flexible transmission buffer assignment function is used and borrowed TXQ receives one frame. Operation is the same as **CFDGRINTSTS_n.QOFRIF**.

This bit of the channel which lends TXMB is a reserve bit.

This bit is a reserve bit when **CFDGFFIMC.FFIEN** is 1'b1.

When reserve bit, this bit is always read as 0.

4.3.46.5 CFDGRINTSTS_n.CFRIF

Common FIFO RX Interrupt Flag Channel n

This bit is set automatically when the Common FIFO RX Interrupt flag of the related channel is set when the Interrupt is enabled.

This bit is cleared automatically when related Common FIFO RX Result status bits are cleared or the Interrupt enable is disabled.

This bit will be cleared automatically in GL_RESET mode or CH_RESET mode.
This bit is a reserve bit when **CFDGFFIMC.FFIEN** is 1'b1.

When reserve bit, this bit is always read as 0.

4.3.46.6 CFDGRINTSTS_n.CFRFIF

Common FIFO FDC level full Interrupt Flag Channel n

This bit is set automatically when the Common FIFO Full Interrupt flag of the related channel is set when the Interrupt is enabled.

This bit is cleared automatically when related Common FIFO RX Result status bits are cleared or the Interrupt enable is disabled.

This bit will be cleared automatically in GL_RESET mode or CH_RESET mode.

This bit is a reserve bit when **CFDGFFIMC.FFIEN** is 1'b1.

When reserve bit, this bit is always read as 0.

4.3.46.7 CFDGRINTSTS_n.CFOFRIF

Common FIFO One Frame RX Interrupt Flag Channel n

This bit is set automatically when the Common FIFO One Frame RX Interrupt flag of the related channel is set when the Interrupt is enabled.

This bit is cleared automatically when related Common FIFO RX Result status bits are cleared or the Interrupt enable is disabled.

This bit will be cleared automatically in GL_RESET mode or CH_RESET mode.

This bit is a reserve bit when **CFDGFFIMC.FFIEN** is 1'b1.

When reserve bit, this bit is always read as 0.

4.3.47 CFDTMCi

TX Message Buffer Control Registers i

Address: Refer to section 4.1

	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset	-	-	-	-	-	TMOM	TMTAR	TMTR

0	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---

Bit	Symbol	Bit name	Function	R/W
b[7:3]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[2]	TMOM	TX Message Buffer One-shot Mode	1'b0: TX Message Buffer not configured in one-shot mode 1'b1: TX Message Buffer configured in one-shot mode	R/W
b[1]	TMTAR	TX Message Buffer Transmission abort Request	1'b0: TX Message Buffer transmission request abort not requested 1'b1: TX Message Buffer transmission request abort requested	R/W
b[0]	TMTR	TX Message Buffer Transmission Request	1'b0: TX Message Buffer Transmission not requested 1'b1: TX Message Buffer Transmission requested	R/W

The TX Message Buffer Control register configures the TX Message Buffer functions.
(no_of_channels = 8)

(no_of_TMBCTRs_per_channel = No. of TX Message Buffer Control Registers per Channel = 64)
Where the total number of TMBCTRs = no_of_TMBCTRs = no_of_TMBCTRs_per_channel * no_of_channels = 64 * 8 = 512
(i = TX Message Buffer Control Register index = [0...no_of_TMBCTRs-1])

4.3.47.1 CFDTMCi.TMTR

TX Message Buffer Transmission Request

If this bit is set, then the RS-CAN-FD module logic will try to transmit the message stored in the corresponding Message Buffer.

Users can only write to this bit when the related RS-CAN-FD channel is in CH_HALT or CH_OPERATION mode.

Users cannot set this bit if the corresponding TX Message Buffer is linked to a COM FIFO in TX or GW mode or is a part of TX Queue.

This bit cannot be directly cleared by a CPU write access.

This bit can only be set when Transmission Result Flag bits (**CFDTMSTSj.TMTRF**) in the **CFDTMSTSj** register corresponding to the MB are cleared to 2'b00.

This bit is automatically cleared by the RS-CAN-FD module logic at the end of successful transmission.

This bit is automatically cleared by the RS-CAN-FD module logic at the end of transmission abort, requested by the corresponding **CFDTMCi.TMTAR** bit.

This bit is automatically cleared by the RS-CAN-FD module logic when there is the detection of CAN bus error or arbitration loss if **CFDTMCi.TMOM** bit is set for the Message Buffer.

This bit is automatically cleared by the RS-CAN-FD module logic when the RS-CAN-FD module enters GL_RESET mode or the related channel enters CH_RESET mode.

4.3.47.2 CFDTMCi.TMTAR

TX Message Buffer Transmission abort Request

If this bit is set, then the RS-CAN-FD module logic will try to abort the transmission of the frame stored in the corresponding Message Buffer.

In most cases, transmission cannot be aborted if the internal scan for transmission is completed and the Message Buffer has already been selected for Transmission. In this case, frame may be transmitted successfully from the Message Buffer. The Message Buffer selection will be released by entering CH_HALT mode.

However, MB selected for transmission can be aborted by Abort request when the CAN node detects a new message on the bus (RX pin) before it can start transmission from the selected MB.

Users can only write to this bit when the related RS-CAN-FD channel is in CH_HALT or CH_OPERATION mode.

This bit can only be set when the related transmit request (**CFDTMCi.TMTR**) bit is set.

This bit cannot be cleared by a CPU write access.

Clearing of this bit by RS-CAN-FD has priority over setting by CPU write access.

This bit is automatically cleared by the RS-CAN-FD module logic at the end of successful transmission

This bit is automatically cleared by the RS-CAN-FD module logic at the end of transmission abort

This bit is automatically cleared by the RS-CAN-FD module logic when there is the detection of CAN bus error or arbitration loss

This bit is automatically cleared by the RS-CAN-FD module logic when the RS-CAN-FD module enters GL_RESET mode or the related channel enters CH_RESET mode.

4.3.47.3 CFDTMCi.TMOM

TX Message Buffer One-shot Mode

If this bit is set, then the RS-CAN-FD module logic will attempt transmission of the message only once.

If the transmission is successful, the **CFDTMSTSj.TMTRF** bits are set to 2'b10 or 2'b11. If it is not successful due to bus error or bus arbitration lost, the transmission is automatically aborted and **CFDTMSTSj.TMTRF** bits are set to 2'b01.

The **CFDTMCi.TMOM** bit will remain set if the transmission is completed successfully or aborted due to error or loss of arbitration.

Users can only write to this bit when the related RS-CAN-FD channel is in CH_HALT or CH_OPERATION mode.

Users should set this bit at the same time as **CFDTMCi.TMTR** bit.

Users should clear this bit by a write access.

If a message has been already requested for transmission, then users should not write to this bit until the message has been successfully transmitted or transmission has been aborted.

This bit will be automatically cleared by the RS-CAN-FD module logic when the RS-CAN-FD module enters GL_RESET mode or the related channel enters CH_RESET mode.

4.3.48 CFDTMSTSj

TX Message Buffer Status Registers j

Address: Refer to section 4.1

	b7	b6	b5	b4	b3	b2	b1	b0
	-	-	-	TMTARM	TMTRM	TMTRF[1:0]	TMTSTS	
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[7:5]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[4]	TMTARM	TX Message Buffer Transmission abort Request Mirrored	1'b0: TX Message Buffer transmission request abort not requested 1'b1: TX Message Buffer transmission request abort requested	R
b[3]	TMTRM	TX Message Buffer Transmission Request Mirrored	1'b0: TX Message Buffer Transmission not requested 1'b1: TX Message Buffer Transmission requested	R
b[2:1]	TMTRF	TX Message Buffer Transmission Result Flag	2'b00: No Result 2'b01: Transmission aborted from the TX MB 2'b10: Transmission successful from the TX MB & Transmission abort was not requested 2'b11: Transmission successful from the TX MB & Transmission abort was requested	R/W
b[0]	TMTSTS	TX Message Buffer Transmission Status	1'b0: No transmission ongoing 1'b1: Transmission ongoing	R

The TX Message Buffer Status Registers show the Transmission and Transmission abort status for the corresponding Message Buffers.

No_of_TMBSTRs_per_channel = No. of TX Message Buffer Status Registers per Channel = 64)

Where the total number of TMBSTRs = no_of_TMBSTRs = no_of_TMBSTRs_per_channel * no_of_channels = 64 * 8 = 512

(j = TX Message Buffer Status Register index = [0...no_of_TMBSTRs-1])

4.3.48.1 CFDTMSTSj.TMTSTS

TX Message Buffer Transmission Status

This bit is set automatically at the start of the transmission from the corresponding TX Message Buffer.

This bit is cleared automatically when the transmission stops.

This bit is cleared automatically when the RS-CAN-FD module enters GL_RESET mode.

This bit is cleared automatically when the related channel enters CH_RESET mode.

4.3.48.2 CFDTMSTSj.TMTRF

TX Message Buffer Transmission Result Flag

These bits show the result for the corresponding TX MB. The status is as follows:

00: Transmission in progress or has not been requested.

01: Transmission has been aborted from the corresponding TX MB.

10: Transmission was successful from the corresponding TX MB and TMTAR bit was not set for this TX MB.

11: Transmission was successful from the corresponding TX MB. But, TMTAR bit was set for this TX MB.

Users can only write to this bit when the related RS-CAN-FD channel is in CH_HALT or CH_OPERATION mode.

These bits will be cleared automatically when the RS-CAN-FD module enters GL_RESET or the related channel enters CH_RESET mode.

4.3.48.3 CFDTMSTSj.TMTRM

TX Message Buffer Transmission Request Mirrored

This bit is set when the **CFDTMCi.TMTR** bit in the corresponding **CFDTMCi** register is set.

This bit is cleared when the **CFDTMCi.TMTR** bit in the corresponding **CFDTMCi** register is cleared.

4.3.48.4 CFDTMSTSj.TMTARM

TX Message Buffer Transmission abort Request Mirrored

This bit is set when the **CFDTMCi.TMTAR** bit in the corresponding **CFDTMCi** register is set.

This bit is cleared when the **CFDTMCi.TMTAR** bit in the corresponding **CFDTMCi** register is cleared.

4.3.49 CFDTMTRSTS_f

TX Message Buffer Transmission Request Status Register f

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
CFDTMTRSTS _g [31:24]								
Value after reset	0	0	0	0	0	0	0	0
CFDTMTRSTS _g [23:16]								
Value after reset	0	0	0	0	0	0	0	0
CFDTMTRSTS _g [15:8]								
Value after reset	0	0	0	0	0	0	0	0
CFDTMTRSTS _g [7:0]								
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:0]	CFDTMTRSTS _g	TX Message Buffer Transmission Request Status	1'b0: Transmission not requested for corresponding TX Message Buffer 1'b1: Transmission requested for corresponding TX Message Buffer	R

(no_of_channels = 8)

(no_of_TXMBs_per_channel = 64)

(no_of_bits_per_register = 32)

(no_of_TXMBs = No. of TX Message Buffers = no_of_channels*no_of_TXMBs_per_channel = 512)

(no_of_CFDTMTRSTS = No. of CFDTMTRSTS = no_of_TXMBs/no_of_bits_per_register = 512 / 32 = 16)

(f = **CFDTMTRSTS** index = [0... . no_of_CFDTMTRSTS -1])

f = [0...15]

When n=0, f=[0, 1] fmin=0, fmax=1

When n=1, f=[2, 3] fmin=2, fmax=3

When n=2, f=[4, 5] fmin=4, fmax=5

When n=3, f=[6, 7] fmin=6, fmax=7

When n=4, f=[8, 9] fmin=8, fmax=9

When n=5, f=[10, 11] fmin=10, fmax=11

When n=6, f=[12, 13] fmin=12, fmax=13

When n=7, f=[14, 15] fmin=14, fmax=15

4.3.49.1 CFDTMTRSTS_f.CFDTMTRSTS_g

TX Message Buffer Transmission Request Status

These bits show the status of the **CFDTMCi.TMTR** bits of the TX Message Buffer Control Registers.

Alignment of the bits is as shown in Table 4.2.

Bit position	g = TX Message Buffer Number
n*64-fmin*32	n*64+0
n*64+1-fmin*32	n*64+1
.	.
.	.
n*64+31-fmin*32	n*64+31
n*64+32-fmax*32	n*64+32
n*64+33-fmax*32	n*64+33
.	.
.	.
n*64+62-fmax*32	n*64+62
n*64+63-fmax*32	n*64+63

Table 4.2 Alignment of CFDTMTRSTS mirror bits

Each bit is set automatically when the corresponding bit is set in the TX Message Buffer Control Registers, only when the Message Buffer is not belonging to a TX Queue.

Each bit is cleared automatically when the corresponding bit is cleared in the TX Message Buffer Control Registers.

Each bit is cleared automatically when the RS-CAN-FD module enters GL_RESET or CH_RESET mode.

4.3.50 CFDTMTARSTSf

TX Message Buffer Transmission Abort Request Status Register f

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
CFDTMTARSTSg[31:24]								
Value after reset	0	0	0	0	0	0	0	0
CFDTMTARSTSg[23:16]								
Value after reset	0	0	0	0	0	0	0	0
CFDTMTARSTSg[15:8]								
Value after reset	0	0	0	0	0	0	0	0
CFDTMTARSTSg[7:0]								
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:0]	CFDTMTARSTSg	TX Message Buffer Transmission abort Request Status	1'b0: Transmission abort not requested for corresponding TX Message Buffer 1'b1: Transmission abort requested for corresponding TX Message Buffer	R

(no_of_channels = 8)

(no_of_TXMBs_per_channel = 64)

(no_of_bits_per_register = 32)

(no_of_TXMBs = No. of TX Message Buffers = no_of_channels*no_of_TXMBs_per_channel = 512)

(no_of_TMTARSTS = No. of TMTARSTS = no_of_TXMBs/no_of_bits_per_register = 512 / 32 = 16)

(f = **CFDTMTARSTS** index = [0...no_of_TMTARSTS-1])

f = [0..15]

When n=0, f=[0, 1] fmin=0, fmax=1

When n=1, f=[2, 3] fmin=2, fmax=3

When n=2, f=[4, 5] fmin=4, fmax=5

When n=3, f=[6, 7] fmin=6, fmax=7

When n=4, f=[8, 9] fmin=8, fmax=9

When n=5, f=[10, 11] fmin=10, fmax=11

When n=6, f=[12, 13] fmin=12, fmax=13

When n=7, f=[14, 15] fmin=14, fmax=15

4.3.50.1 CFDTMTARSTSf.CFDTMTARSTSg

TX Message Buffer Transmission abort Request Status

These bits show the status of the **CFDTMCi.TMTAR** bits of the TX Message Buffer Control Registers.

Alignment of the bits is as shown in Table 4.3.

Bit position	$g = \text{TX Message Buffer Number}$
$n*64-f\min*32$	$n*64+0$
$n*64+1-f\min*32$	$n*64+1$
.	.
.	.
$n*64+31-f\min*32$	$n*64+31$
$n*64+32-f\max*32$	$n*64+32$
$n*64+33-f\max*32$	$n*64+33$
.	.
.	.
$n*64+62-f\max*32$	$n*64+62$
$n*64+63-f\max*32$	$n*64+63$

Table 4.3 Alignment of CFDTMTARSTS mirror bits

Each bit is set automatically when the corresponding bit is set in the TX Message Buffer Control Registers, also when the Message Buffer is belonging to a TX Queue.

Each bit is cleared automatically when the corresponding bit is cleared in the TX Message Buffer Control Registers.

Each bit is cleared automatically when the RS-CAN-FD module enters GL_RESET or CH_RESET mode. If a CAN channel enters CH_RESET mode, then the bits related to that channel will be cleared.

4.3.51 CFDTMTCSTSf

TX Message Buffer Transmission Completion Status Register f

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
CFDTMTCSTSg[31:24]								
Value after reset	0	0	0	0	0	0	0	0
CFDTMTCSTSg[23:16]								
Value after reset	0	0	0	0	0	0	0	0
CFDTMTCSTSg[15:8]								
Value after reset	0	0	0	0	0	0	0	0
CFDTMTCSTSg[7:0]								
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:0]	CFDTMTCSTSg	TX Message Buffer Transmission Completion Status	1'b0: Transmission not complete for corresponding TX Message Buffer 1'b1: Transmission completed for corresponding TX Message Buffer	R

(no_of_channels = 8)

(no_of_TXMBs_per_channel = 64)

(no_of_bits_per_register = 32)

(no_of_TXMBs = No. of TX Message Buffers = no_of_channels*no_of_TXMBs_per_channel = 512)

(no_of_CFDTMTCSTS = No. of CFDTMTCSTS = no_of_TXMBs/no_of_bits_per_register = 512 / 32 = 16)

(f = CFDTMTCSTS index = [0...no_of_CFDTMTCSTS-1])

f = [0...15]

When n=0, f=[0, 1] fmin=0, fmax=1

When n=1, f=[2, 3] fmin=2, fmax=3

When n=2, f=[4, 5] fmin=4, fmax=5

When n=3, f=[6, 7] fmin=6, fmax=7

When n=4, f=[8, 9] fmin=8, fmax=9

When n=5, f=[10, 11] fmin=10, fmax=11

When n=6, f=[12, 13] fmin=12, fmax=13

When n=7, f=[14, 15] fmin=14, fmax=15

4.3.51.1 CFDTMTCSTSf.CFDTMTCSTSg

TX Message Buffer Transmission Completion Status

These bits show the status of successful completion of the TX Message Buffer Status Registers.

Alignment of the bits is as shown in Table 4.4.

Bit position	g = TX Message Buffer Number
n*64-fmin*32	n*64+0
n*64+1-fmin*32	n*64+1
.	.
.	.
n*64+31-fmin*32	n*64+31
n*64+32-fmax*32	n*64+32
n*64+33-fmax*32	n*64+33
.	.
.	.
n*64+62-fmax*32	n*64+62
n*64+63-fmax*32	n*64+63

Table 4.4 Alignment of CFDTMTCSTS mirror bits

Each bit is set automatically when the corresponding bit is set in the TX Message Buffer Status Registers.

Each bit is cleared automatically when the corresponding bit is cleared in the TX Message Buffer Status Registers.

Each bit is cleared automatically when the RS-CAN-FD module enters GL_RESET or CH_RESET mode. If a CAN channel enters CH_RESET mode, then the bits related to that channel will be cleared.

4.3.52 CFDTMTASTS_f

TX Message Buffer Transmission Abort Status Register f

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
CFDTMTASTS _g [31:24]								
Value after reset	0	0	0	0	0	0	0	0
CFDTMTASTS _g [23:16]								
Value after reset	0	0	0	0	0	0	0	0
CFDTMTASTS _g [15:8]								
Value after reset	0	0	0	0	0	0	0	0
CFDTMTASTS _g [7:0]								
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:0]	CFDTMTASTS _g	TX Message Buffer Transmission abort Status	1'b0: Transmission not aborted for corresponding TX Message Buffer 1'b1: Transmission aborted for corresponding TX Message Buffer	R

(no_of_channels = 8)

(no_of_TXMBs_per_channel = 64)

(no_of_bits_per_register = 32)

(no_of_TXMBs = No. of TX Message Buffers = no_of_channels*no_of_TXMBs_per_channel = 512)

(no_of_CFDTMTASTS = No. of CFDTMTASTS = no_of_TXMBs/no_of_bits_per_register = 512 / 32 = 16)

(f = CFDTMTASTS index = [0...no_of_CFDTMTASTS-1])

f = [0..15]

When n=0, f=[0, 1] fmin=0, fmax=1

When n=1, f=[2, 3] fmin=2, fmax=3

When n=2, f=[4, 5] fmin=4, fmax=5

When n=3, f=[6, 7] fmin=6, fmax=7

When n=4, f=[8, 9] fmin=8, fmax=9

When n=5, f=[10, 11] fmin=10, fmax=11

When n=6, f=[12, 13] fmin=12, fmax=13

When n=7, f=[14, 15] fmin=14, fmax=15

4.3.52.1 CFDTMTASTS_f.CFDTMTASTS_g

TX Message Buffer Transmission abort Status

These bits show the status of the successful transmission abort of the corresponding TX Message Buffer.

Alignment of the bits is as shown in Table 4.5.

Bit position	g = TX Message Buffer Number
n*64-fmin*32	n*64+0
n*64+1-fmin*32	n*64+1
.	.
.	.
n*64+31-fmin*32	n*64+31
n*64+32-fmax*32	n*64+32
n*64+33-fmax*32	n*64+33
.	.
.	.
n*64+62-fmax*32	n*64+62
n*64+63-fmax*32	n*64+63

Table 4.5 Alignment of CFDTMTASTS mirror bits

Each bit is set automatically when the **CFDTMSTSj.TMTRF** bits are set to 2'b01 in the corresponding TX Message Buffer Status Register.

Each bit is cleared automatically when the **CFDTMSTSj.TMTRF** bits are cleared in the corresponding TX Message Buffer Status Register.

Each bit is cleared automatically when the RS-CAN-FD module enters GL_RESET or CH_RESET mode. If a CAN channel enters CH_RESET mode, then the bits related to that channel will be cleared.

4.3.53 CFDTMIECf

TX Message Buffer Interrupt Enable Configuration Register f

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
	TMIEg[31:24]							
Value after reset	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
Value after reset	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset	0	0	0	0	0	0	0	0
	TMIEg[7:0]							
Bit	Symbol	Bit name	Function					R/W
b[31:0]	TMIEg	TX Message Buffer Interrupt Enable	1'b0: TX Message Buffer Interrupt disabled for corresponding TX message buffer 1'b1: TX Message Buffer Interrupt enabled for corresponding TX message buffer					R/W

(no_of_channels = 8)

(no_of_TXMBs_per_channel = 64)

(no_of_bits_per_register = 32)

(no_of_TXMBs = No. of TX Message Buffers = no_of_channels*no_of_TXMBs_per_channel = 512)

(no_of_CFDTMIEC = No. of CFDTMIEC = no_of_TXMBs/no_of_bits_per_register = 512 / 32 = 16)

(f = CFDTMIEC index = [0...no_of_CFDTMIEC-1])

f = [0..15]

When n=0, f=[0, 1] fmin=0, fmax=1

When n=1, f=[2, 3] fmin=2, fmax=3

When n=2, f=[4, 5] fmin=4, fmax=5

When n=3, f=[6, 7] fmin=6, fmax=7

When n=4, f=[8, 9] fmin=8, fmax=9

When n=5, f=[10, 11] fmin=10, fmax=11

When n=6, f=[12, 13] fmin=12, fmax=13

When n=7, f=[14, 15] fmin=14, fmax=15

4.3.53.1 CFDTMIECf.TMIEg

TX Message Buffer Interrupt Enable

If this bit is set, then an interrupt will be generated at the end of a successful transmission from the corresponding Message Buffer.

Please refer to Section 9 for TX Message Buffer Interrupt specification.

Alignment of the bits is as shown in Table 4.6

Bit position	$g = \text{TX Message Buffer Number}$
$n*64-f\min*32$	$n*64+0$
$n*64+1-f\min*32$	$n*64+1$
.	.
.	.
$n*64+31-f\min*32$	$n*64+31$
$n*64+32-f\max*32$	$n*64+32$
$n*64+33-f\max*32$	$n*64+33$
.	.
.	.
$n*64+62-f\max*32$	$n*64+62$
$n*64+63-f\max*32$	$n*64+63$

Table 4.6 Alignment of TMIE bits

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

Users should not write to these bits when the related CAN_channel is in CH_SLEEP mode.

Users should not write to these bits if the corresponding TX Message Buffer is part of a TX Queue.

Users should not write to these bits if the corresponding TX Message Buffer is linked to a Common FIFO (via **CFDCFCCd.CFTML** bits)

4.3.54 CFDTXQCC0n

TX Queue Configuration / Control Registers0 n

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
Value after reset	-	-	-	DMAE	-	-	QOWIE	QMEIE
Value after reset	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset	-	-	-	-	-	TXQOFTXIE	TXQOFRXIE	TXQFIE
Value after reset	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
Value after reset	-	-	-			TXQDC[4:0]		
Value after reset	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset	TXQIM	-	TXQTXIE	-	-	TXQOWE	TXQGWE	TXQE
Bit	Symbol	Bit name	Function					R/W
b[31:29]	-	Reserved	These bits are read as 0. The write value should be always 0.					R/W
b[28]	DMAE	DMA TX Transfer Enable for TXQ for FFI Mode	1'b0: DMA Transfer Request disabled for channel n 1'b1: DMA Transfer Request enabled for channel n					R/W
b[27:26]	-	Reserved	These bits are read as 0. The write value should be always 0.					R/W
b[25]	QOWIE	TXQ Message overwrite Error Interrupt Enable for FFI Mode	1'b0: TXQ Message overwrite Error Interrupt Disabled 1'b1: TXQ Message overwrite Error Interrupt Enabled					R/W
b[24]	QMEIE	TXQ Message lost Error Interrupt Enable for FFI Mode	1'b0: TXQ Message Lost Error Interrupt Disabled 1'b1: TXQ Message Lost Error Interrupt Enabled					R/W
b[23:19]	-	Reserved	These bits are read as 0. The write value should be always 0.					R/W
b[18]	TXQOFTXIE	TXQ One Frame Transmission Interrupt Enable	1'b0: One Frame TX Interrupt generation disabled 1'b1: One Frame TX Interrupt generation enabled					R/W
b[17]	TXQOFRXIE	TXQ One Frame Reception Interrupt Enable	1'b0: One Frame RX Interrupt generation disabled 1'b1: One Frame RX Interrupt generation enabled					R/W
b[16]	TXQFIE	TXQ Full interrupt Enable	1'b0: TX Queue Full Interrupt generation disabled 1'b1: TX Queue Full Interrupt generation enabled					R/W
b[15:13]	-	Reserved	These bits are read as 0. The write value should be always 0.					R/W
b[12:8]	TXQDC	TX Queue Depth Configuration	5'b00000: 0 messages 5'b00001: Reserved 5'b00010: 3 messages 5'b00011: 4 messages : 5'b11110: 31 messages 5'b11111: 32 messages					R/W
b[7]	TXQIM	TX Queue Interrupt Mode	1'b0: when the last message is successfully transmitted 1'b1: At every successful transmission					R/W
b[6]	-	Reserved	This bit is read as 0b. The write value should be always 0b.					R/W
b[5]	TXQTXIE	TX Queue TX Interrupt Enable	1'b0: TX Queue TX Interrupt disabled 1'b1: TX Queue TX Interrupt enabled					R/W
b[4:3]	-	Reserved	These bits are read as 0. The write value should be always 0.					R/W
b[2]	TXQOWE	TX Queue Overwrite Mode Enable	1'b0: TX Queue OW mode disabled 1'b1: TX Queue OW mode enabled					R/W
b[1]	TXQGWE		1'b0: TX Queue GW mode disabled					R/W

		TX Queue Gateway Mode Enable	1'b1: TX Queue GW mode enabled	
b[0]	TXQE	TX Queue Enable	1'b0: TX Queue disabled 1'b1: TX Queue enabled	R/W

The TX Queue Configuration / Control Registers are used to configure the TX Queue transmission.

TXQ0 is composed of TXMB0 to TXMB31 (at the maximum) when **TXQE** is enabled.

4.3.54.1 CFDTXQCC0n.TXQE

TX Queue Enable

Users cannot write to this bit when the related RS-CAN-FD channel is in CH_RESET mode.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

This bit cannot be set if the configured TX Queue depth is 5'h0 (**CFDTXQCC0n.TXQDC == 5'h0**).

Users cannot write to this bit when the related channel is in CH_SLEEP mode.

This bit is cleared automatically when the related RS-CAN-FD channel is in CH_RESET mode.

4.3.54.2 CFDTXQCC0n.TXQGWE

TX Queue Gateway Mode Enable

When this bit is set, the TX queue is in TX queue GW mode.

Users cannot write to the bit when the RS-CAN-FD module is in GL_SLEEP mode.

Users cannot write to the bit when the related RS-CAN-FD channel is in CH_HALT or CH_OPERATION mode.

Users should not write to the bit when the related RS-CAN-FD channel is in CH_SLEEP.

When this bit is set, CPU must not access the TX queue.

4.3.54.3 CFDTXQCC0n.TXQOWE

TX Queue Overwrite Mode Enable

When this bit is set, the TX queue is in TX queue overwrite mode.

An overwrite function is valid when the same ID as ID of the data written in from the gateway or CPU is in TX Queue, e.g. when a frame is received and is stored into the TX queue, if a message with the same ID has been stored in the TX queue, the old message will be overwritten by the new message.

Users cannot write to the bit when the RS-CAN-FD module is in GL_SLEEP mode.

Users cannot write to the bit when the related RS-CAN-FD channel is in CH_HALT or CH_OPERATION mode.

Users should not write to the bit when the related RS-CAN-FD channel is in CH_SLEEP.

When users use the function in GW mode, the depth of TXQ (**CFDTXQCC0n.TXQDC**) should be configured to the value which is the various number of ID which is used in the TX queue plus 3.

Then the function is valid for the standard ID frame and is invalid for the extended ID frame.

Users should not write change for this bit when the **CFDTXQCC0n.TXQE** bit is 1'b1.

4.3.54.4 CFDTXQCC0n.TXQTXIE

TX Queue TX Interrupt Enable

If this bit is set, then an interrupt will be generated based on the setting of the TXQIM bit.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

Users should not write to this bit when the RS-CAN-FD module is in CH_SLEEP mode.

4.3.54.5 CFDTXQCC0n.TXQIM

TX Queue Interrupt Mode

This bit selects the Interrupt generation condition for the TX Queue.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

Users should not write to this bit when the RS-CAN-FD module is in CH_SLEEP mode.

Users should not write to this bit when the RS-CAN-FD module is in CH_HALT or CH_OPERATION mode.

4.3.54.6 CFDTXQCC0n.TXQDC

TX Queue Depth Configuration

These bits select the depth of the transmission queue. The Message Buffer selection starts from MB[0] up to MB[31] depending upon the configured depth.

When users use TXQ1 and TXQ0 simultaneously, the total depth of TXQ1 and TXQ0 should be 32 or less.

Users cannot write to these bits when the RS-CAN-FD module is in GL_SLEEP mode.

Users cannot write to these bits when the related RS-CAN-FD channel is in CH_HALT or CH_OPERATION mode.

Users should not write to these bits when the related RS-CAN-FD channel is in CH_SLEEP.

4.3.54.7 CFDTXQCC0n.TXQFIE

TXQ Full Interrupt Enable

If this bit is set, then an interrupt will be generated based on the setting of the **CFDTXQSTS0n.TXQFIF** bit.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

Users should not write to this bit when the RS-CAN-FD module is in CH_SLEEP mode.

Users should write 1 to this bit only when the Gateway mode (**CFDTXQCC0n.TXQGWE=1**).

4.3.54.8 CFDTXQCC0n.TXQOFRXIE

TXQ One Frame Reception Interrupt Enable

If this bit is set, then an interrupt will be generated based on the setting of the **CFDTXQSTS0n.TXQOFRXIF** bit.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

Users should not write to this bit when the RS-CAN-FD module is in CH_SLEEP mode.

Users should write 1 to this bit only when the Gateway mode (**CFDTXQCC0n.TXQGWE=1**).

4.3.54.9 CFDTXQCC0n.TXQOFTXIE

TXQ One Frame Transmission Interrupt Enable

If this bit is set, then an interrupt will be generated based on the setting of the **CFDTXQSTS0n.TXQOFTXIF** bit.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

Users should not write to this bit when the RS-CAN-FD module is in CH_SLEEP mode.

4.3.54.10 CFDTXQCC0n.QMEIE

TXQ Message lost Error Interrupt Enable for FFI Mode

This bit is used instead of **CFDGCTR.QMEIE** when **CFDGFFIMC.FFIEN** is 1'b1 and **CFDGVMEM.TXQMLT** is 1.

Users can only access to this bit when **CFDGFFIMC.FFIEN** is 1'b1.

Users cannot write to this bit when RS-CAN-FD module is in GL_SLEEP mode.

4.3.54.11 CFDTXQCC0n. QOWEIE

TXQ Message overwrite Error Interrupt Enable for FFI Mode

This bit is used instead of **CFDGCTR.QOWEIE** when **CFDGFFIMC.FFIEN** is 1'b1 and **CFDGVMIS.TXQOW** is 1.

Users can only access to this bit when **CFDGFFIMC.FFIEN** is 1'b1.

Users cannot write to this bit when RS-CAN-FD module is in GL_SLEEP mode.

4.3.54.12 CFDTXQCC0n. DMAE

DMA TX Transfer Enable for TXQ for FFI Mode

Users can only access to this bit when **CFDGFFIMC.FFIEN** is 1'b1.

This bit will be cleared automatically when the RS-CAN-FD module enters GL_RESET.

Users cannot write to this bit when RS-CAN-FD module is in GL_SLEEP mode.

4.3.55 CFDTXQCC1n

TX Queue Configuration / Control Registers1 n

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24	
Value after reset	-	-	-	-	-	-	QOWEIE	QMEIE	
	b23	b22	b21	b20	b19	b18	b17	b16	
Value after reset	-	-	-	-	-	TXQOFTXIE	TXQOFRXIE	TXQFIE	
	b15	b14	b13	b12	b11	b10	b9	b8	
Value after reset	-	-	-			TXQDC[4:0]	0	0	
	b7	b6	b5	b4	b3	b2	b1	b0	
Value after reset	TXQIM	-	TXQTXIE	-	-	TXQOWE	TXQGWE	TXQE	
Bit	Symbol	Bit name	Function						R/W
b[31:26]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[25]	QOWEIE	TXQ Message overwrite Error Interrupt Enable for FFI Mode	1'b0: TXQ Message overwrite Error Interrupt Disabled 1'b1: TXQ Message overwrite Error Interrupt Enabled						R/W
b[24]	QMEIE	TXQ Message lost Error Interrupt Enable for FFI Mode	1'b0: TXQ Message Lost Error Interrupt Disabled 1'b1: TXQ Message Lost Error Interrupt Enabled						R/W
b[23:19]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[18]	TXQOFTXIE	TXQ One Frame Transmission Interrupt Enable	1'b0: One Frame TX Interrupt generation disabled 1'b1: One Frame TX Interrupt generation enabled						R/W
b[17]	TXQOFRXIE	TXQ One Frame Reception Interrupt Enable	1'b0: One Frame RX Interrupt generation disabled 1'b1: One Frame RX Interrupt generation enabled						R/W
b[16]	TXQFIE	TXQ Full Interrupt Enable	1'b0: TX Queue Full Interrupt generation disabled 1'b1: TX Queue Full Interrupt generation enabled						R/W
b[15:13]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[12:8]	TXQDC	TX Queue Depth Configuration	5'b00000: 0 messages 5'b00001: Reserved 5'b00010: 3 messages 5'b00011: 4 messages : 5'b11110: 31 messages 5'b11111: 32 messages						R/W
b[7]	TXQIM	TX Queue Interrupt Mode	1'b0: when the last message is successfully transmitted 1'b1: At every successful transmission						R/W
b[6]	-	Reserved	This bit is read as 0b. The write value should be always 0b.						R/W
b[5]	TXQTXIE	TX Queue TX Interrupt Enable	1'b0: TX Queue TX Interrupt disabled 1'b1: TX Queue TX Interrupt enabled						R/W
b[4:3]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[2]	TXQOWE	TX Queue Overwrite Mode Enable	1'b0: TX Queue OW mode disabled 1'b1: TX Queue OW mode enabled						R/W
b[1]	TXQGWE	TX Queue Gateway Mode Enable	1'b0: TX Queue GW mode disabled 1'b1: TX Queue GW mode enabled						R/W
b[0]	TXQE	TX Queue Enable	1'b0: TX Queue disabled						R/W

1'b1: TX Queue enabled

The TX Queue Configuration / Control Registers are used to configure the TX Queue transmission.

TXQ1 is composed of TXMB31 to TXMB0 (at the maximum) when **TXQE** is enabled.

4.3.55.1 CFDTXQCC1n.TXQE

TX Queue Enable

Users cannot write to this bit when the related RS-CAN-FD channel is in CH_RESET mode.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

This bit cannot be set if the configured TX Queue depth is 5'h0 (**CFDTXQCC1n.TXQDC == 5'h0**).

Users cannot write to this bit when the related channel is in CH_SLEEP mode.

This bit is cleared automatically when the related RS-CAN-FD channel is in CH_RESET mode.

4.3.55.2 CFDTXQCC1n.TXQGWE

TX Queue Gateway Mode Enable

When this bit is set, the TX queue is in TX queue GW mode.

Users cannot write to the bit when the RS-CAN-FD module is in GL_SLEEP mode.

Users cannot write to the bit when the related RS-CAN-FD channel is in CH_HALT or CH_OPERATION mode.

Users should not write to the bit when the related RS-CAN-FD channel is in CH_SLEEP.

When this bit is set, CPU must not access the TX queue.

4.3.55.3 CFDTXQCC1n.TXQOWE

TX Queue Overwrite Mode Enable

When this bit is set, the TX queue is in TX queue overwrite mode.

An overwrite function is valid when the same ID as ID of the data written in from the gateway or CPU is in TX Queue, e.g. when a frame is received and is stored into the TX queue, if a message with the same ID has been stored in the TX queue, the old message will be overwritten by the new message.

Users cannot write to the bit when the RS-CAN-FD module is in GL_SLEEP mode.

Users cannot write to the bit when the related RS-CAN-FD channel is in CH_HALT or CH_OPERATION mode.

Users should not write to the bit when the related RS-CAN-FD channel is in CH_SLEEP.

When users use the function in GW mode, the depth of TXQ (**CFDTXQCC1n.TXQDC**) should be configured to the value which is the various number of ID which is used in the TX queue plus 3.

Then the function is valid for the standard ID frame and is invalid for the extended ID frame.

Users should not write change for this bit when the **CFDTXQCC1n.TXQE** bit is 1'b1.

4.3.55.4 CFDTXQCC1n.TXQTXIE

TX Queue TX Interrupt Enable

If this bit is set, then an interrupt will be generated based on the setting of the TXQIM bit.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

Users should not write to this bit when the RS-CAN-FD module is in CH_SLEEP mode.

4.3.55.5 CFDTXQCC1n.TXQIM

TX Queue Interrupt Mode

This bit selects the Interrupt generation condition for the TX Queue.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

Users should not write to this bit when the RS-CAN-FD module is in CH_SLEEP mode.

Users should not write to this bit when the RS-CAN-FD module is in CH_HALT or CH_OPERATION mode.

4.3.55.6 CFDTXQCC1n.TXQDC

TX Queue Depth Configuration

These bits select the depth of the transmission queue. The Message Buffer selection starts from MB[31] down to MB[0] depending upon the configured depth.

When users use TXQ1 and TXQ0 simultaneously, the total depth of TXQ1 and TXQ0 should be 32 or less.

Users cannot write to these bits when the RS-CAN-FD module is in GL_SLEEP mode.

Users cannot write to these bits when the related RS-CAN-FD channel is in CH_HALT or CH_OPERATION mode.

Users should not write to these bits when the related RS-CAN-FD channel is in CH_SLEEP.

4.3.55.7 CFDTXQCC1n.TXQFIE

TXQ Full Interrupt Enable

If this bit is set, then an interrupt will be generated based on the setting of the **CFDTXQSTS1n.TXQFIF** bit.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

Users should not write to this bit when the RS-CAN-FD module is in CH_SLEEP mode.

Users should write 1 to this bit only when the Gateway mode (**CFDTXQCC1n.TXQGWE=1**).

4.3.55.8 CFDTXQCC1n.TXQOFRXIE

TXQ One Frame Reception Interrupt Enable

If this bit is set, then an interrupt will be generated based on the setting of the **CFDTXQSTS1n.TXQOFRXIF** bit.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

Users should not write to this bit when the RS-CAN-FD module is in CH_SLEEP mode.

Users should write 1 to this bit only when the Gateway mode (**CFDTXQCC1n.TXQGWE=1**).

4.3.55.9 CFDTXQCC1n.TXQOFTXIE

TXQ One Frame Transmission Interrupt Enable

If this bit is set, then an interrupt will be generated based on the setting of the **CFDTXQSTS1n.TXQOFTXIF** bit.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

Users should not write to this bit when the RS-CAN-FD module is in CH_SLEEP mode.

4.3.55.10 CFDTXQCC1n.QMEIE

TXQ Message lost Error Interrupt Enable for FFI Mode

This bit is used instead of **CFDGCTR.QMEIE** when **CFDGFFIMC.FFIEN** is 1'b1 and **CFDGVMEM.TXQMLT** is 1.

Users can only access to this bit when **CFDGFFIMC.FFIEN** is 1'b1.

Users cannot write to this bit when RS-CAN-FD module is in GL_SLEEP mode.

4.3.55.11 CFDTXQCC1n.QOWEIE

TXQ Message overwrite Error Interrupt Enable for FFI Mode

This bit is used instead of **CFDGCTR.QOWEIE** when **CFDGFFIMC.FFIEN** is 1'b1 and **CFDGVMEIS.TXQOW** is 1.

Users can only access to this bit when **CFDGFFIMC.FFIEN** is 1'b1.

Users cannot write to this bit when RS-CAN-FD module is in GL_SLEEP mode.

4.3.56 CFDTXQCC2n

TX Queue Configuration / Control Registers2 n

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24	
Value after reset	-	-	-	-	-	-	QOWEIE	QMEIE	
	b23	b22	b21	b20	b19	b18	b17	b16	
Value after reset	-	-	-	-	-	TXQOFTXIE	TXQOFRXIE	TXQFIE	
	b15	b14	b13	b12	b11	b10	b9	b8	
Value after reset	-	-	-			TXQDC[4:0]	0	0	
	b7	b6	b5	b4	b3	b2	b1	b0	
Value after reset	TXQIM	-	TXQTXIE	-	-	TXQOWE	TXQGWE	TXQE	
Bit	Symbol	Bit name	Function						R/W
b[31:26]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[25]	QOWEIE	TXQ Message overwrite Error Interrupt Enable for FFI Mode	1'b0: TXQ Message overwrite Error Interrupt Disabled 1'b1: TXQ Message overwrite Error Interrupt Enabled						R/W
b[24]	QMEIE	TXQ Message lost Error Interrupt Enable for FFI Mode	1'b0: TXQ Message Lost Error Interrupt Disabled 1'b1: TXQ Message Lost Error Interrupt Enabled						R/W
b[23:19]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[18]	TXQOFTXIE	TXQ One Frame Transmission Interrupt Enable	1'b0: One Frame TX Interrupt generation disabled 1'b1: One Frame TX Interrupt generation enabled						R/W
b[17]	TXQOFRXIE	TXQ One Frame Reception Interrupt Enable	1'b0: One Frame RX Interrupt generation disabled 1'b1: One Frame RX Interrupt generation enabled						R/W
b[16]	TXQFIE	TXQ Full interrupt Enable	1'b0: TX Queue Full Interrupt generation disabled 1'b1: TX Queue Full Interrupt generation enabled						R/W
b[15:13]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[12:8]	TXQDC	TX Queue Depth Configuration	5'b00000: 0 messages 5'b00001: Reserved 5'b00010: 3 messages 5'b00011: 4 messages : 5'b11110: 31 messages 5'b11111: 32 messages						R/W
b[7]	TXQIM	TX Queue Interrupt Mode	1'b0: when the last message is successfully transmitted 1'b1: At every successful transmission						R/W
b[6]	-	Reserved	This bit is read as 0b. The write value should be always 0b.						R/W
b[5]	TXQTXIE	TX Queue TX Interrupt Enable	1'b0: TX Queue TX Interrupt disabled 1'b1: TX Queue TX Interrupt enabled						R/W
b[4:3]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[2]	TXQOWE	TX Queue Overwrite Mode Enable	1'b0: TX Queue OW mode disabled 1'b1: TX Queue OW mode enabled						R/W
b[1]	TXQGWE	TX Queue Gateway Mode Enable	1'b0: TX Queue GW mode disabled 1'b1: TX Queue GW mode enabled						R/W
b[0]	TXQE	TX Queue Enable	1'b0: TX Queue disabled 1'b1: TX Queue enabled						R/W

The TX Queue Configuration / Control Registers are used to configure the TX Queue transmission.

TXQ2 is composed of TXMB32 to TXMB63 (at the maximum) when **TXQE** is enabled.

4.3.56.1 CFDTXQCC2n.TXQE

TX Queue Enable

Users cannot write to this bit when the related RS-CAN-FD channel is in CH_RESET mode.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

This bit cannot be set if the configured TX Queue depth is 5'h0 (**CFDTXQCC2n.TXQDC == 5'h0**).

Users cannot write to this bit when the related channel is in CH_SLEEP mode.

This bit is cleared automatically when the related RS-CAN-FD channel is in CH_RESET mode.

4.3.56.2 CFDTXQCC2n.TXQGWE

TX Queue Gateway Mode Enable

When this bit is set, the TX queue is in TX queue GW mode.

Users cannot write to the bit when the RS-CAN-FD module is in GL_SLEEP mode.

Users cannot write to the bit when the related RS-CAN-FD channel is in CH_HALT or CH_OPERATION mode.

Users should not write to the bit when the related RS-CAN-FD channel is in CH_SLEEP.

When this bit is set, CPU must not access the TX queue.

4.3.56.3 CFDTXQCC2n.TXQOWE

TX Queue Overwrite Mode Enable

When this bit is set, the TX queue is in TX queue overwrite mode.

An overwrite function is valid when the same ID as ID of the data written in from the gateway or CPU is in TX Queue, e.g. when a frame is received and is stored into the TX queue, if a message with the same ID has been stored in the TX queue, the old message will be overwritten by the new message.

Users cannot write to the bit when the RS-CAN-FD module is in GL_SLEEP mode.

Users cannot write to the bit when the related RS-CAN-FD channel is in CH_HALT or CH_OPERATION mode.

Users should not write to the bit when the related RS-CAN-FD channel is in CH_SLEEP.

When users use the function in GW mode, the depth of TXQ (**CFDTXQCC2n.TXQDC**) should be configured to the value which is the various number of ID which is used in the TX queue plus 3.

Then the function is valid for the standard ID frame and is invalid for the extended ID frame.

Users should not write change for this bit when the **CFDTXQCC2n.TXQE** bit is 1'b1.

4.3.56.4 CFDTXQCC2n.TXQTXIE

TX Queue TX Interrupt Enable

If this bit is set, then an interrupt will be generated based on the setting of the TXQIM bit.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

Users should not write to this bit when the RS-CAN-FD module is in CH_SLEEP mode.

4.3.56.5 CFDTXQCC2n.TXQIM

TX Queue Interrupt Mode

This bit selects the Interrupt generation condition for the TX Queue.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

Users should not write to this bit when the RS-CAN-FD module is in CH_SLEEP mode.

Users should not write to this bit when the RS-CAN-FD module is in CH_HALT or CH_OPERATION mode.

4.3.56.6 CFDTXQCC2n.TXQDC

TX Queue Depth Configuration

These bits select the depth of the transmission queue. The Message Buffer selection starts from MB[32] up to MB[63] depending upon the configured depth.

When users use TXQ3 and TXQ2 simultaneously, the total depth of TXQ3 and TXQ2 should be 32 or less.

Users cannot write to these bits when the RS-CAN-FD module is in GL_SLEEP mode.

Users cannot write to these bits when the related RS-CAN-FD channel is in CH_HALT or CH_OPERATION mode.

Users should not write to these bits when the related RS-CAN-FD channel is in CH_SLEEP.

4.3.56.7 CFDTXQCC2n.TXQFIE

TXQ Full Interrupt Enable

If this bit is set, then an interrupt will be generated based on the setting of the **CFDTXQSTS2n.TXQFIF** bit.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

Users should not write to this bit when the RS-CAN-FD module is in CH_SLEEP mode.

Users should write 1 to this bit only when the Gateway mode (**CFDTXQCC2n.TXQGWE=1**)

4.3.56.8 CFDTXQCC2n.TXQOFRXIE

TXQ One Frame Reception Interrupt Enable

If this bit is set, then an interrupt will be generated based on the setting of the **CFDTXQSTS2n.TXQOFRXIF** bit.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

Users should not write to this bit when the RS-CAN-FD module is in CH_SLEEP mode.

Users should write 1 to this bit only when the Gateway mode (**CFDTXQCC2n.TXQGWE=1**)

4.3.56.9 CFDTXQCC2n.TXQOFTXIE

TXQ One Frame Transmission Interrupt Enable

If this bit is set, then an interrupt will be generated based on the setting of the **CFDTXQSTS2n.TXQOFTXIF** bit.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

Users should not write to this bit when the RS-CAN-FD module is in CH_SLEEP mode.

4.3.56.10 CFDTXQCC2n.QMEIE

TXQ Message lost Error Interrupt Enable for FFI Mode

This bit is used instead of **CFDGCTR.QMEIE** when **CFDGFFIMC.FFIEN** is 1'b1 and **CFDGVMEIS.TXQMLT** is 1.

Users can only access to this bit when **CFDGFFIMC.FFIEN** is 1'b1.

Users cannot write to this bit when RS-CAN-FD module is in GL_SLEEP mode.

4.3.56.11 CFDTXQCC2n.QOWEIE

TXQ Message overwrite Error Interrupt Enable for FFI Mode

This bit is used instead of **CFDGCTR.QOWEIE** when **CFDGFFIMC.FFIEN** is 1'b1 and **CFDGVMEIS.TXQOW** is 1.

Users can only access to this bit when **CFDGFFIMC.FFIEN** is 1'b1.



Users cannot write to this bit when RS-CAN-FD module is in GL_SLEEP mode.

4.3.57 CFDTXQCC3n

TX Queue Configuration / Control Registers3 n

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
Value after reset	-	-	-	DMAE	-	-	QOWEIE	-
	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset	-	-	-	-	-	TXQOFTXIE	-	-
	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
Value after reset	-	-	-			TXQDC[4:0]		
	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset	TXQIM	-	TXQTXIE	-	-	TXQOWE	-	TXQE
	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:19]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[28]	DMAE	DMA TX Transfer Enable for TXQ for FFI Mode	1'b0: DMA Transfer Request disabled for channel n 1'b1: DMA Transfer Request enabled for channel n	R/W
b[27:26]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[25]	QOWEIE	TXQ Message overwrite Error Interrupt Enable for FFI Mode	1'b0: TXQ Message overwrite Error Interrupt Disabled 1'b1: TXQ Message overwrite Error Interrupt Enabled	R/W
b[24:19]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[18]	TXQOFTXIE	TXQ One Frame Transmission Interrupt Enable	1'b0: One Frame TX Interrupt generation disabled 1'b1: One Frame TX Interrupt generation enabled	R/W
b[17:13]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[12:8]	TXQDC	TX Queue Depth Configuration	5'b00000: 0 messages 5'b00001: Reserved 5'b00010: 3 messages 5'b00011: 4 messages : 5'b11110: 31 messages 5'b11111: 32 messages	R/W
b[7]	TXQIM	TX Queue Interrupt Mode	1'b0: when the last message is successfully transmitted 1'b1: At every successful transmission	R/W
b[6]	-	Reserved	This bit is read as 0b. The write value should be always 0b.	R/W
b[5]	TXQTXIE	TX Queue TX Interrupt Enable	1'b0: TX Queue TX Interrupt disabled 1'b1: TX Queue TX Interrupt enabled	R/W
b[4:3]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[2]	TXQOWE	TX Queue Overwrite Mode Enable	1'b0: TX Queue OW mode disabled 1'b1: TX Queue OW mode enabled	R/W
b[1]	-	Reserved	This bit is read as 0b. The write value should be always 0b.	R/W
b[0]	TXQE	TX Queue Enable	1'b0: TX Queue disabled 1'b1: TX Queue enabled	R/W

The TX Queue Configuration / Control Registers are used to configure the TX Queue transmission.

TXQ3 is composed of TXMB63 to TXMB32 (at the maximum) when TXQE is enabled.

4.3.57.1 CFDTXQCC3n.TXQE

TX Queue Enable

Users cannot write to this bit when the related RS-CAN-FD channel is in CH_RESET mode.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

This bit cannot be set if the configured TX Queue depth is 5'h0 (**CFDTXQCC3n.TXQDC == 5'h0**).

Users cannot write to this bit when the related channel is in CH_SLEEP mode.

This bit is cleared automatically when the related RS-CAN-FD channel is in CH_RESET mode.

4.3.57.2 CFDTXQCC3n.TXQOWE

TX Queue Overwrite Mode Enable

When this bit is set, the TX queue is in TX queue overwrite mode.

An overwrite function is valid when the same ID as ID of the data written in from CPU is in TX Queue, e.g. when a frame is received and is stored into the TX queue, if a message with the same ID has been stored in the TX queue, the old message will be overwritten by the new message.

Users cannot write to the bit when the RS-CAN-FD module is in GL_SLEEP mode.

Users cannot write to the bit when the related RS-CAN-FD channel is in CH_HALT or CH_OPERATION mode.

Users should not write to the bit when the related RS-CAN-FD channel is in CH_SLEEP.

When users use the function, the depth of TXQ (**CFDTXQCC3n.TXQDC**) should be configured to the value which is the various number of ID which is used in the TX queue plus 3.

Then the function is valid for the standard ID frame and is invalid for the extended ID frame.

Users should not write change for this bit when the **CFDTXQCC3n.TXQE** bit is 1'b1.

4.3.57.3 CFDTXQCC3n.TXQTXIE

TX Queue TX Interrupt Enable

If this bit is set, then an interrupt will be generated based on the setting of the TXQIM bit.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

Users should not write to this bit when the RS-CAN-FD module is in CH_SLEEP mode.

4.3.57.4 CFDTXQCC3n.TXQIM

TX Queue Interrupt Mode

This bit selects the Interrupt generation condition for the TX Queue.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

Users should not write to this bit when the RS-CAN-FD module is in CH_SLEEP mode.

Users should not write to this bit when the RS-CAN-FD module is in CH_HALT or CH_OPERATION mode.

4.3.57.5 CFDTXQCC3n.TXQDC

TX Queue Depth Configuration

These bits select the depth of the transmission queue. The Message Buffer selection starts from MB[63] down to MB[32] depending upon the configured depth.

When users use TXQ3 and TXQ2 simultaneously, the total depth of TXQ3 and TXQ2 should be 32 or less.

Users cannot write to these bits when the RS-CAN-FD module is in GL_SLEEP mode.

Users cannot write to these bits when the related RS-CAN-FD channel is in CH_HALT or CH_OPERATION mode.

Users should not write to these bits when the related RS-CAN-FD channel is in CH_SLEEP.

4.3.57.6 CFDTXQCC3n.TXQOFTXIE

TXQ One Frame Transmission Interrupt Enable

If this bit is set, then an interrupt will be generated based on the setting of the **CFDTXQSTS3n.TXQOFTXIF** bit.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

Users should not write to this bit when the RS-CAN-FD module is in CH_SLEEP mode.

4.3.57.7 CFDTXQCC3n.QOWEIE

TXQ Message overwrite Error Interrupt Enable for FFI Mode

This bit is used instead of **CFDGCTR.QOWEIE** when **CFDGFFIMC.FFIEN** is 1'b1 and **CFDGVMEMIS.TXQOW** is 1.

Users can only access to this bit when **CFDGFFIMC.FFIEN** is 1'b1.

Users cannot write to this bit when RS-CAN-FD module is in GL_SLEEP mode.

4.3.57.8 CFDTXQCC3n.DMAE

DMA TX Transfer Enable for TXQ for FFI Mode

Users can only access to this bit when **CFDGFFIMC.FFIEN** is 1'b1.

This bit will be cleared automatically when the RS-CAN-FD module enters GL_RESET.

Users cannot write to this bit when RS-CAN-FD module is in GL_SLEEP mode.

4.3.58 CFDTXQSTS0n

TX Queue Status Registers0 n

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
Value after reset	-	-	-	TXQDMASTS	-	-	-	-
	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset	-	-	-	TXQMOW	TXQMLT	TXQOFTXIF	TXQOFRXIF	TXQFIF
	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
Value after reset	-	-	-	TXQMC[5:0]				
	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset	-	-	-	-	-	TXQTXIF	TXQFLL	TXQEMP
	0	0	0	0	0	0	0	1
Bit	Symbol	Bit name	Function					R/W
b[31:29]	-	Reserved	These bits are read as 0. The write value should be always 0.					R/W
b[28]	TXQDMASTS	DMA TX Transfer Status for TXQ0	1'b0: DMA transfer stopped 1'b1: DMA transfer enable					R
b[27:21]	-	Reserved	These bits are read as 0. The write value should be always 0.					R/W
b[20]	TXQMOW	TXQ message overwrite	1'b0: No Message overwrite in TXQ 1'b1: Message overwrite in TXQ					R/W
b[19]	TXQMLT	TXQ Message Lost	1'b0: No Message Lost in TXQ 1'b1: TXQ Message Lost					R/W
b[18]	TXQOFTXIF	TXQ One Frame Transmission Interrupt Flag	If one frame transmits from TXQ, an interrupt will set.					R/W
b[17]	TXQOFRXIF	TXQ One Frame Reception Interrupt Flag	If TXQ receives one frame, an interrupt will set.					R/W
b[16]	TXQFIF	TXQ Full Interrupt Flag	If TXQ will be in full status, an interrupt will set.					R/W
b[15:14]	-	Reserved	These bits are read as 0. The write value should be always 0.					R/W
b[13:8]	TXQMC	TX Queue Message Count	No. of Messages in the TX Queue					R
b[7:3]	-	Reserved	These bits are read as 0. The write value should be always 0.					R/W
b[2]	TXQTXIF	TX Queue TX Interrupt Flag	1'b0: TX Queue interrupt condition not satisfied after Frame TX 1'b1: TX Queue interrupt condition satisfied after Frame TX					R/W
b[1]	TXQFLL	TX Queue Full	1'b0: TX Queue Not Full 1'b1: TX Queue Full					R
b[0]	TXQEMP	TX Queue Empty	1'b0: TX Queue Not Empty 1'b1: TX Queue Empty					R

The TX Queue status registers show the status of the TX Queue of corresponding CAN Channel.

4.3.58.1 CFDTXQSTS0n.TXQEMP

TX Queue Empty

This bit is set automatically when the TX Queue is disabled or no messages are stored in the TX Queue.

This bit is set automatically when the last message is transmitted from the TX Queue.

This bit is set automatically when the related RS-CAN-FD channel enters CH_RESET mode.

The bit is cleared automatically when the first message to be transmitted is stored in the TX Queue.

4.3.58.2 CFDTXQSTS0n.TXQFLL

TX Queue Full

This bit is set automatically when the number of CAN messages stored in the TX Queue matches the configured TX Queue depth.

This bit is cleared automatically when the number of CAN messages stored in the TX Queue is less than the configured TX Queue depth.

This bit is cleared automatically when the related channel enters CH_RESET mode.

4.3.58.3 CFDTXQSTS0n.TXQTXIF

TX Queue TX Interrupt Flag

This bit will not be cleared automatically if the TX Queue is disabled.

Users cannot write to this bit when the related RS-CAN-FD channel is in CH_SLEEP or CH_RESET mode.

When stopping TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue.

This bit is set automatically when the configured interrupt condition is satisfied for the TX Queue.

The bit is cleared by writing 1'b0 to it.

The bit is cleared if the related RS-CAN-FD channel enters CH_RESET mode.

Do not use bit clear instruction for clearing this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Then the other bits remain set to 1'b1.

Writing 1'b1 has no influence on the bit values.

4.3.58.4 CFDTXQSTS0n.TXQMC

TX Queue Message Count

These bits show the number of CAN messages in the TX Queue.

These bits are cleared automatically when the related channel is in CH_RESET mode.

4.3.58.5 CFDTXQSTS0n.TXQFIF

TXQ Full Interrupt Flag

This bit will not be cleared automatically if the TX Queue is disabled.

Users cannot write to this bit when the related RS-CAN-FD channel is in CH_SLEEP or CH_RESET mode.

When stopping TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue.

Only when the Gateway mode (**CFDTXQCC0n.TXQGWE=1**), this bit is set automatically when TX Queue transits to a buffer full status.

The bit is cleared by writing 1'b0 to it.

The bit is cleared if the related RS-CAN-FD channel enters CH_RESET mode.

Do not use bit clear instruction for clearing this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Then the other bits remain set to 1'b1.

Writing 1'b1 has no influence on the bit values.

4.3.58.6 CFDTXQSTS0n.TXQOFRXIF

TXQ One Frame Reception Interrupt Flag

This bit will not be cleared automatically if the TX Queue is disabled.

Users cannot write to this bit when the related RS-CAN-FD channel is in CH_SLEEP or CH_RESET mode.

When stopping TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue.

When receiving data is stored in TX queue in Gateway mode, this bit is set automatically.

The bit is cleared by writing 1'b0 to it.

The bit is cleared if the related RS-CAN-FD channel enters CH_RESET mode.

This function can use only the Gateway mode of TX queue.

Do not use bit clear instruction for clearing this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Then the other bits remain set to 1'b1.

Writing 1'b1 has no influence on the bit values.

4.3.58.7 CFDTXQSTS0n.TXQOFTXIF

TXQ One Frame Transmission Interrupt Flag

This bit will not be cleared automatically if the TX Queue is disabled.

Users cannot write to this bit when the related RS-CAN-FD channel is in CH_SLEEP or CH_RESET mode.

When stopping TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue.

When transmission is successful in TX queue, this bit is set automatically.

The bit is cleared by writing 1'b0 to it.

The bit is cleared if the related RS-CAN-FD channel enters CH_RESET mode.

Do not use bit clear instruction for clearing this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Then the other bits remain set to 1'b1.

Writing 1'b1 has no influence on the bit values.

4.3.58.8 CFDTXQSTS0n.TXQMLT

TXQ Message Lost

This bit will not be cleared automatically if the TX Queue is disabled.

Users cannot write to this bit when the related RS-CAN-FD channel is in CH_SLEEP or CH_RESET mode.

When stopping TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue.

When Message lost occurs in the Gateway mode of TX queue, this bit is set automatically.

The bit is cleared by writing 1'b0 to it.

The bit is cleared if the related RS-CAN-FD channel enters CH_RESET mode.

Do not use bit clear instruction for clearing this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Then the other bits remain set to 1'b1.

Writing 1'b1 has no influence on the bit values.

4.3.58.9 CFDTXQSTS0n.TXQMOW

TXQ message overwrite

This bit will not be cleared automatically if the TX Queue is disabled.

Users cannot write to this bit when the related RS-CAN-FD channel is in CH_SLEEP or CH_RESET mode.

When stopping TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue.

When **CFDTXQCC0n.TXQOWE=1** and Message overwrite occurs in TX queue, this bit is set automatically.

The bit is cleared by writing 1'b0 to it.

The bit is cleared if the related RS-CAN-FD channel enters CH_RESET mode.

Do not use bit clear instruction for clearing this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Then the other bits remain set to 1'b1.

Writing 1'b1 has no influence on the bit values.

4.3.58.10 CFDTXQSTS0n.TXQDMASTS

DMA TX Transfer Status for TXQ0

User can only read to this bit when **CFDGFFIMC.FFIEN** is 1'b1.

When **CFDGFFIMC.FFIEN** is 1'b0, this bit is always read as 0.

4.3.59 CFDTXQSTS1n

TX Queue Status Registers1 n

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24	
Value after reset	-	-	-	-	-	-	-	-	
Value after reset	0	0	0	0	0	0	0	0	
Value after reset	b23	b22	b21	b20	b19	b18	b17	b16	
Value after reset	-	-	-	TXQMOW	TXQMLT	TXQOFTXIF	TXQOFRXIF	TXQFIF	
Value after reset	0	0	0	0	0	0	0	0	
Value after reset	b15	b14	b13	b12	b11	b10	b9	b8	
Value after reset	-	-				TXQMC[5:0]			
Value after reset	0	0	0	0	0	0	0	0	
Bit	Symbol	Bit name	Function						R/W
b[31:21]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[20]	TXQMOW	TXQ message overwrite	1'b0: No Message overwrite in TXQ 1'b1: Message overwrite in TXQ						R/W
b[19]	TXQMLT	TXQ Message Lost	1'b0: No Message Lost in TXQ 1'b1: TXQ Message Lost						R/W
b[18]	TXQOFTXIF	TXQ One Frame Transmission Interrupt Flag	If one frame transmits from TXQ, an interrupt will set.						R/W
b[17]	TXQOFRXIF	TXQ One Frame Reception Interrupt Flag	If TXQ receives one frame, an interrupt will set.						R/W
b[16]	TXQFIF	TXQ Full Interrupt Flag	If TXQ will be in full status, an interrupt will set.						R/W
b[15:14]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[13:8]	TXQMC	TX Queue Message Count	No. of Messages in the TX Queue						R
b[7:3]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[2]	TXQTXIF	TX Queue TX Interrupt Flag	1'b0: TX Queue interrupt condition not satisfied after Frame TX 1'b1: TX Queue interrupt condition satisfied after Frame TX						R/W
b[1]	TXQFLL	TX Queue Full	1'b0: TX Queue Not Full 1'b1: TX Queue Full						R
b[0]	TXQEMP	TX Queue Empty	1'b0: TX Queue Not Empty 1'b1: TX Queue Empty						R

The TX Queue status registers show the status of the TX Queue of corresponding CAN Channel.

4.3.59.1 CFDTXQSTS1n.TXQEMP

TX Queue Empty

This bit is set automatically when the TX Queue is disabled or no messages are stored in the TX Queue.

This bit is set automatically when the last message is transmitted from the TX Queue.

This bit is set automatically when the related RS-CAN-FD channel enters CH_RESET mode.

The bit is cleared automatically when the first message to be transmitted is stored in the TX Queue.

4.3.59.2 CFDTXQSTS1n.TXQFLL

TX Queue Full

This bit is set automatically when the number of CAN messages stored in the TX Queue matches the

configured TX Queue depth.

This bit is cleared automatically when the number of CAN messages stored in the TX Queue is less than the configured TX Queue depth.

This bit is cleared automatically when the related channel enters CH_RESET mode.

4.3.59.3 CFDTXQSTS1n.TXQTXIF

TX Queue TX Interrupt Flag

This bit will not be cleared automatically if the TX Queue is disabled.

Users cannot write to this bit when the related RS-CAN-FD channel is in CH_SLEEP or CH_RESET mode.

When stopping TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue.

This bit is set automatically when the configured interrupt condition is satisfied for the TX Queue.

The bit is cleared by writing 1'b0 to it.

The bit is cleared if the related RS-CAN-FD channel enters CH_RESET mode.

Do not use bit clear instruction for clearing this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Then the other bits remain set to 1'b1.

Writing 1'b1 has no influence on the bit values.

4.3.59.4 CFDTXQSTS1n.TXQMC

TX Queue Message Count

These bits show the number of CAN messages in the TX Queue.

These bits are cleared automatically when the related channel is in CH_RESET mode.

4.3.59.5 CFDTXQSTS1n.TXQFIF

TXQ Full Interrupt Flag

This bit will not be cleared automatically if the TX Queue is disabled.

Users cannot write to this bit when the related RS-CAN-FD channel is in CH_SLEEP or CH_RESET mode.

When stopping TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue.

Only when the Gateway mode (**CFDTXQCC1n.TXQGWE=1**), this bit is set automatically when TX Queue transits to a buffer full status.

The bit is cleared by writing 1'b0 to it.

The bit is cleared if the related RS-CAN-FD channel enters CH_RESET mode.

Do not use bit clear instruction for clearing this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Then the other bits remain set to 1'b1.

Writing 1'b1 has no influence on the bit values.

4.3.59.6 CFDTXQSTS1n.TXQOFRXIF

TXQ One Frame Reception Interrupt Flag

This bit will not be cleared automatically if the TX Queue is disabled.

Users cannot write to this bit when the related RS-CAN-FD channel is in CH_SLEEP or CH_RESET mode.

When stopping TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue.

When receiving data is stored in TX queue in Gateway mode, this bit is set automatically.

The bit is cleared by writing 1'b0 to it.

The bit is cleared if the related RS-CAN-FD channel enters CH_RESET mode.

This function can use only the Gateway mode of TX queue.

Do not use bit clear instruction for clearing this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Then the other bits remain set to 1'b1.

Writing 1'b1 has no influence on the bit values.

4.3.59.7 CFDTXQSTS1n.TXQOFTXIF

TXQ One Frame Transmission Interrupt Flag

This bit will not be cleared automatically if the TX Queue is disabled.

Users cannot write to this bit when the related RS-CAN-FD channel is in CH_SLEEP or CH_RESET mode.

When stopping TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue.

When transmission is successful in TX queue, this bit is set automatically.

The bit is cleared by writing 1'b0 to it.

The bit is cleared if the related RS-CAN-FD channel enters CH_RESET mode.

Do not use bit clear instruction for clearing this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Then the other bits remain set to 1'b1.

Writing 1'b1 has no influence on the bit values.

4.3.59.8 CFDTXQSTS1n.TXQMLT

TXQ Message Lost

This bit will not be cleared automatically if the TX Queue is disabled.

Users cannot write to this bit when the related RS-CAN-FD channel is in CH_SLEEP or CH_RESET mode.

When stopping TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue.

When Message lost occurs in the Gateway mode of TX queue, this bit is set automatically.

The bit is cleared by writing 1'b0 to it.

The bit is cleared if the related RS-CAN-FD channel enters CH_RESET mode.

Do not use bit clear instruction for clearing this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Then the other bits remain set to 1'b1.

Writing 1'b1 has no influence on the bit values.

4.3.59.9 CFDTXQSTS1n.TXQMOW

TXQ message overwrite

This bit will not be cleared automatically if the TX Queue is disabled.

Users cannot write to this bit when the related RS-CAN-FD channel is in CH_SLEEP or CH_RESET mode.

When stopping TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue.

When **CFDTXQCC1n.TXQOWE=1** and Message overwrite occurs in TX queue, this bit is set automatically.

The bit is cleared by writing 1'b0 to it.

The bit is cleared if the related RS-CAN-FD channel enters CH_RESET mode.

Do not use bit clear instruction for clearing this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Then the other bits remain set to 1'b1.

Writing 1'b1 has no influence on the bit values.

4.3.60 CFDTXQSTS2n

TX Queue Status Registers2 n

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24	
Value after reset	-	-	-	-	-	-	-	-	
	0	0	0	0	0	0	0	0	
	b23	b22	b21	b20	b19	b18	b17	b16	
Value after reset	-	-	-	TXQMOW	TXQMLT	TXQOFTXIF	TXQOFRXIF	TXQFIF	
	0	0	0	0	0	0	0	0	
	b15	b14	b13	b12	b11	b10	b9	b8	
Value after reset	-	-			TXQMC[5:0]				
	0	0	0	0	0	0	0	0	
	b7	b6	b5	b4	b3	b2	b1	b0	
Value after reset	-	-	-	-	-	TXQTXIF	TXQFLL	TXQEMP	
	0	0	0	0	0	0	0	1	
Bit	Symbol	Bit name	Function						R/W
b[31:21]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[20]	TXQMOW	TXQ message overwrite	1'b0: No Message overwrite in TXQ 1'b1: Message overwrite in TXQ						R/W
b[19]	TXQMLT	TXQ Message Lost	1'b0: No Message Lost in TXQ 1'b1: TXQ Message Lost						R/W
b[18]	TXQOFTXIF	TXQ One Frame Transmission Interrupt Flag	If one frame transmits from TXQ, an interrupt will set.						R/W
b[17]	TXQOFRXIF	TXQ One Frame Reception Interrupt Flag	If TXQ receives one frame, an interrupt will set.						R/W
b[16]	TXQFIF	TXQ Full Interrupt Flag	If TXQ will be in full status, an interrupt will set.						R/W
b[15:14]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[13:8]	TXQMC	TX Queue Message Count	No. of Messages in the TX Queue						R
b[7:3]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[2]	TXQTXIF	TX Queue TX Interrupt Flag	1'b0: TX Queue interrupt condition not satisfied after Frame TX 1'b1: TX Queue interrupt condition satisfied after Frame TX						R/W
b[1]	TXQFLL	TX Queue Full	1'b0: TX Queue Not Full 1'b1: TX Queue Full						R
b[0]	TXQEMP	TX Queue Empty	1'b0: TX Queue Not Empty 1'b1: TX Queue Empty						R

The TX Queue status registers show the status of the TX Queue of corresponding CAN Channel.

4.3.60.1 CFDTXQSTS2n.TXQEMP

TX Queue Empty

This bit is set automatically when the TX Queue is disabled or no messages are stored in the TX Queue.

This bit is set automatically when the last message is transmitted from the TX Queue.

This bit is set automatically when the related RS-CAN-FD channel enters CH_RESET mode.

The bit is cleared automatically when the first message to be transmitted is stored in the TX Queue.

4.3.60.2 CFDTXQSTS2n.TXQFLL

TX Queue Full

This bit is set automatically when the number of CAN messages stored in the TX Queue matches the configured TX Queue depth.

This bit is cleared automatically when the number of CAN messages stored in the TX Queue is less than the

configured TX Queue depth.

This bit is cleared automatically when the related channel enters CH_RESET mode.

4.3.60.3 CFDTXQSTS2n.TXQTXIF

TX Queue TX Interrupt Flag

This bit will not be cleared automatically if the TX Queue is disabled.

Users cannot write to this bit when the related RS-CAN-FD channel is in CH_SLEEP or CH_RESET mode.

When stopping TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue.

This bit is set automatically when the configured interrupt condition is satisfied for the TX Queue.

The bit is cleared by writing 1'b0 to it.

The bit is cleared if the related RS-CAN-FD channel enters CH_RESET mode.

Do not use bit clear instruction for clearing this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Then the other bits remain set to 1'b1.

Writing 1'b1 has no influence on the bit values.

4.3.60.4 CFDTXQSTS2n.TXQMC

TX Queue Message Count

These bits show the number of CAN messages in the TX Queue.

These bits are cleared automatically when the related channel is in CH_RESET mode.

4.3.60.5 CFDTXQSTS2n.TXQFIF

TXQ Full Interrupt Flag

This bit will not be cleared automatically if the TX Queue is disabled.

Users cannot write to this bit when the related RS-CAN-FD channel is in CH_SLEEP or CH_RESET mode.

When stopping TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue.

Only when the Gateway mode (**CFDTXQCC2n.TXQGWE=1**), this bit is set automatically when TX Queue transits to a buffer full status.

The bit is cleared by writing 1'b0 to it.

The bit is cleared if the related RS-CAN-FD channel enters CH_RESET mode.

Do not use bit clear instruction for clearing this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Then the other bits remain set to 1'b1.

Writing 1'b1 has no influence on the bit values.

4.3.60.6 CFDTXQSTS2n.TXQOFRXIF

TXQ One Frame Reception Interrupt Flag

This bit will not be cleared automatically if the TX Queue is disabled.

Users cannot write to this bit when the related RS-CAN-FD channel is in CH_SLEEP or CH_RESET mode.

When stopping TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue.

When receiving data is stored in TX queue in Gateway mode, this bit is set automatically.

The bit is cleared by writing 1'b0 to it.

The bit is cleared if the related RS-CAN-FD channel enters CH_RESET mode.

This function can use only the Gateway mode of TX queue.

Do not use bit clear instruction for clearing this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Then the other bits remain set to 1'b1.

Writing 1'b1 has no influence on the bit values.

4.3.60.7 CFDTXQSTS2n.TXQOFTXIF

TXQ One Frame Transmission Interrupt Flag

This bit will not be cleared automatically if the TX Queue is disabled.

Users cannot write to this bit when the related RS-CAN-FD channel is in CH_SLEEP or CH_RESET mode.

When stopping TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue.

When transmission is successful in TX queue, this bit is set automatically.

The bit is cleared by writing 1'b0 to it.

The bit is cleared if the related RS-CAN-FD channel enters CH_RESET mode.

Do not use bit clear instruction for clearing this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Then the other bits remain set to 1'b1.

Writing 1'b1 has no influence on the bit values.

4.3.60.8 CFDTXQSTS2n.TXQMLT

TXQ Message Lost

This bit will not be cleared automatically if the TX Queue is disabled.

Users cannot write to this bit when the related RS-CAN-FD channel is in CH_SLEEP or CH_RESET mode.

When stopping TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue.

When Message lost occurs in the Gateway mode of TX queue, this bit is set automatically.

The bit is cleared by writing 1'b0 to it.

The bit is cleared if the related RS-CAN-FD channel enters CH_RESET mode.

Do not use bit clear instruction for clearing this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Then the other bits remain set to 1'b1.

Writing 1'b1 has no influence on the bit values.

4.3.60.9 CFDTXQSTS2n.TXQMOW

TXQ message overwrite

This bit will not be cleared automatically if the TX Queue is disabled.

Users cannot write to this bit when the related RS-CAN-FD channel is in CH_SLEEP or CH_RESET mode.

When stopping TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue.

When **CFDTXQCC2n.TXQOWE=1** and Message overwrite occurs in TX queue, this bit is set automatically.

The bit is cleared by writing 1'b0 to it.

The bit is cleared if the related RS-CAN-FD channel enters CH_RESET mode.

Do not use bit clear instruction for clearing this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Then the other bits remain set to 1'b1.

Writing 1'b1 has no influence on the bit values.

4.3.61 CFDTXQSTS3n

TX Queue Status Registers3 n

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
Value after reset	-	-	-	TXQDMASTS	-	-	-	-
	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset	-	-	-	TXQMOW	-	TXQOFTXIF	-	-
	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
Value after reset	-	-	-	-	-	TXQMC[5:0]	-	-
	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset	-	-	-	-	-	TXQTXIF	TXQFLL	TXQEMP
	0	0	0	0	0	0	0	1
Bit	Symbol	Bit name	Function					R/W
b[31:29]	-	Reserved	These bits are read as 0. The write value should be always 0.					R/W
b[28]	TXQDMASTS	DMA TX Transfer Status for TXQ3	1'b0: DMA transfer stopped 1'b1: DMA transfer enable					R
b[27:21]	-	Reserved	These bits are read as 0. The write value should be always 0.					R/W
b[20]	TXQMOW	TXQ message overwrite	1'b0: No Message overwrite in TXQ 1'b1: Message overwrite in TXQ					R/W
b[19]	-	Reserved	This bit is read as 0b. The write value should be always 0b.					R/W
b[18]	TXQOFTXIF	TXQ One Frame Transmission Interrupt Flag	If one frame transmits from TXQ, an interrupt will set.					R/W
b[17:14]	-	Reserved	These bits are read as 0. The write value should be always 0.					R/W
b[13:8]	TXQMC	TX Queue Message Count	No. of Messages in the TX Queue					R
b[7:3]	-	Reserved	These bits are read as 0. The write value should be always 0.					R/W
b[2]	TXQTXIF	TX Queue TX Interrupt Flag	1'b0: TX Queue interrupt condition not satisfied after Frame TX 1'b1: TX Queue interrupt condition satisfied after Frame TX					R/W
b[1]	TXQFLL	TX Queue Full	1'b0: TX Queue Not Full 1'b1: TX Queue Full					R
b[0]	TXQEMP	TX Queue Empty	1'b0: TX Queue Not Empty 1'b1: TX Queue Empty					R

The TX Queue status registers show the status of the TX Queue of corresponding CAN Channel.

4.3.61.1 CFDTXQSTS3n.TXQEMP

TX Queue Empty

This bit is set automatically when the TX Queue is disabled or no messages are stored in the TX Queue.

This bit is set automatically when the last message is transmitted from the TX Queue.

This bit is set automatically when the related RS-CAN-FD channel enters CH_RESET mode.

The bit is cleared automatically when the first message to be transmitted is stored in the TX Queue.

4.3.61.2 CFDTXQSTS3n.TXQFLL

TX Queue Full

This bit is set automatically when the number of CAN messages stored in the TX Queue matches the configured TX Queue depth.

This bit is cleared automatically when the number of CAN messages stored in the TX Queue is less than the configured TX Queue depth.

This bit is cleared automatically when the related channel enters CH_RESET mode.

4.3.61.3 CFDTXQSTS3n.TXQTXIF

TX Queue TX Interrupt Flag

This bit will not be cleared automatically if the TX Queue is disabled.

Users cannot write to this bit when the related RS-CAN-FD channel is in CH_SLEEP or CH_RESET mode.

When stopping TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue.

This bit is set automatically when the configured interrupt condition is satisfied for the TX Queue.

The bit is cleared by writing 1'b0 to it.

The bit is cleared if the related RS-CAN-FD channel enters CH_RESET mode.

Do not use bit clear instruction for clearing this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Then the other bits remain set to 1'b1.

Writing 1'b1 has no influence on the bit values.

4.3.61.4 CFDTXQSTS3n.TXQMC

TX Queue Message Count

These bits show the number of CAN messages in the TX Queue.

These bits are cleared automatically when the related channel is in CH_RESET mode.

4.3.61.5 CFDTXQSTS3n.TXQOFTXIF

TXQ One Frame Transmission Interrupt Flag

This bit will not be cleared automatically if the TX Queue is disabled.

Users cannot write to this bit when the related RS-CAN-FD channel is in CH_SLEEP or CH_RESET mode.

When stopping TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue.

When transmission is successful in TX queue, this bit is set automatically.

The bit is cleared by writing 1'b0 to it.

The bit is cleared if the related RS-CAN-FD channel enters CH_RESET mode.

Do not use bit clear instruction for clearing this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Then the other bits remain set to 1'b1.

Writing 1'b1 has no influence on the bit values.

4.3.61.6 CFDTXQSTS3n.TXQMOW

TXQ message overwrite

This bit will not be cleared automatically if the TX Queue is disabled.

Users cannot write to this bit when the related RS-CAN-FD channel is in CH_SLEEP or CH_RESET mode.

When stopping TX Queue, this bit should be cleared, after disabling TXQE and checking an empty state of TX Queue.

When CFDTXQCC3n.TXQOWE=1 and Message overwrite occurs in TX queue, this bit is set automatically.

The bit is cleared by writing 1'b0 to it.

The bit is cleared if the related RS-CAN-FD channel enters CH_RESET mode.

Do not use bit clear instruction for clearing this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Then the other bits remain set to 1'b1.

Writing 1'b1 has no influence on the bit values.

4.3.61.7 CFDTXQSTS3n.TXQDMASTS

DMA TX Transfer Status for TXQ3

User can only read to this bit when **CFDGFFIMC.FFIEN** is 1'b1.

When **CFDGFFIMC.FFIEN** is 1'b0, this bit is always read as 0.

4.3.62 CFDTXQPCTR0n

TX Queue Pointer Control Registers0 n

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24	
Value after reset	-	-	-	-	-	-	-	-	
	0	0	0	0	0	0	0	0	
	b23	b22	b21	b20	b19	b18	b17	b16	
Value after reset	-	-	-	-	-	-	-	-	
	0	0	0	0	0	0	0	0	
	b15	b14	b13	b12	b11	b10	b9	b8	
Value after reset	-	-	-	-	-	-	-	-	
	0	0	0	0	0	0	0	0	
	b7	b6	b5	b4	b3	b2	b1	b0	
	TXQPC[7:0]								
Value after reset	0	0	0	0	0	0	0	0	
Bit	Symbol	Bit name	Function						R/W
b[31:8]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[7:0]	TXQPC	TX Queue Pointer Control	Increments the write pointer to the TX Queue buffer in the corresponding channel						W

These registers can be used to confirm storage of a full message in the corresponding TX Queue Buffers.

4.3.62.1 CFDTXQPCTR0n.TXQPC

TX Queue Pointer Control

When the value 8'hFF is written to these bits, then the Write Pointer of the corresponding TX Queue Buffer is updated and a transmit request is initiated for this message.

Read value from these bits is always 8'h0.

Users cannot write to these bits when the related RS-CAN-FD channel is in CH_SLEEP or CH_RESET mode.

Users should only write 8'hFF to this register when the corresponding TX Queue is enabled and not full.

Users should only write 8'hFF to this register when the Common FIFO is enabled and is not configured in GW mode.

Users should not write to the FIFO control registers when DMA is enabled.

4.3.63 CFDTXQPCTR1n

TX Queue Pointer Control Registers1 n

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24	
Value after reset	-	-	-	-	-	-	-	-	
	0	0	0	0	0	0	0	0	
	b23	b22	b21	b20	b19	b18	b17	b16	
Value after reset	-	-	-	-	-	-	-	-	
	0	0	0	0	0	0	0	0	
	b15	b14	b13	b12	b11	b10	b9	b8	
Value after reset	-	-	-	-	-	-	-	-	
	0	0	0	0	0	0	0	0	
	b7	b6	b5	b4	b3	b2	b1	b0	
	TXQPC[7:0]								
Value after reset	0	0	0	0	0	0	0	0	
Bit	Symbol	Bit name	Function						R/W
b[31:8]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[7:0]	TXQPC	TX Queue Pointer Control	Increments the write pointer to the TX Queue buffer in the corresponding channel						W

These registers can be used to confirm storage of a full message in the corresponding TX Queue Buffers.

4.3.63.1 CFDTXQPCTR1n.TXQPC

TX Queue Pointer Control

When the value 8'hFF is written to these bits, then the Write Pointer of the corresponding TX Queue Buffer is updated and a transmit request is initiated for this message.

Read value from these bits is always 8'h0.

Users cannot write to these bits when the related RS-CAN-FD channel is in CH_SLEEP or CH_RESET mode.

Users should only write 8'hFF to this register when the corresponding TX Queue is enabled and not full.

Users should only write 8'hFF to this register when the Common FIFO is enabled and is not configured in GW mode.

4.3.64 CFDTXQPCTR2n

TX Queue Pointer Control Registers2 n

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24	
	-	-	-	-	-	-	-	-	
Value after reset	0	0	0	0	0	0	0	0	
	b23	b22	b21	b20	b19	b18	b17	b16	
	-	-	-	-	-	-	-	-	
Value after reset	0	0	0	0	0	0	0	0	
	b15	b14	b13	b12	b11	b10	b9	b8	
	-	-	-	-	-	-	-	-	
Value after reset	0	0	0	0	0	0	0	0	
	b7	b6	b5	b4	b3	b2	b1	b0	
	TXQPC[7:0]								
Value after reset	0	0	0	0	0	0	0	0	
Bit	Symbol	Bit name	Function						R/W
b[31:8]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[7:0]	TXQPC	TX Queue Pointer Control	Increments the write pointer to the TX Queue buffer in the corresponding channel						W

These registers can be used to confirm storage of a full message in the corresponding TX Queue Buffers.

4.3.64.1 CFDTXQPCTR2n.TXQPC

TX Queue Pointer Control

When the value 8'hFF is written to these bits, then the Write Pointer of the corresponding TX Queue Buffer is updated and a transmit request is initiated for this message.

Read value from these bits is always 8'h0.

Users cannot write to these bits when the related RS-CAN-FD channel is in CH_SLEEP or CH_RESET mode.

Users should only write 8'hFF to this register when the corresponding TX Queue is enabled and not full.

Users should only write 8'hFF to this register when the Common FIFO is enabled and is not configured in GW mode.

4.3.65 CFDTXQPCTR3n

TX Queue Pointer Control Registers3 n

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24	
Value after reset	-	-	-	-	-	-	-	-	
	0	0	0	0	0	0	0	0	
	b23	b22	b21	b20	b19	b18	b17	b16	
Value after reset	-	-	-	-	-	-	-	-	
	0	0	0	0	0	0	0	0	
	b15	b14	b13	b12	b11	b10	b9	b8	
Value after reset	-	-	-	-	-	-	-	-	
	0	0	0	0	0	0	0	0	
	b7	b6	b5	b4	b3	b2	b1	b0	
	TXQPC[7:0]								
Value after reset	0	0	0	0	0	0	0	0	
Bit	Symbol	Bit name	Function						R/W
b[31:8]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[7:0]	TXQPC	TX Queue Pointer Control	Increments the write pointer to the TX Queue buffer in the corresponding channel						W

These registers can be used to confirm storage of a full message in the corresponding TX Queue Buffers.

4.3.65.1 CFDTXQPCTR3n.TXQPC

TX Queue Pointer Control

When the value 8'hFF is written to these bits, then the Write Pointer of the corresponding TX Queue Buffer is updated and a transmit request is initiated for this message.

Read value from these bits is always 8'h0.

Users cannot write to these bits when the related RS-CAN-FD channel is in CH_SLEEP or CH_RESET mode.

Users should only write 8'hFF to this register when the corresponding TX Queue is enabled and not full.

Users should not write to the FIFO control registers when DMA is enabled.

4.3.66 CFDTXQESTS

TX Queue Empty Status Registers

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24	
	TXQxEMP[31:24]								
Value after reset	1	1	1	1	1	1	1	1	
	b23	b22	b21	b20	b19	b18	b17	b16	
Value after reset	1	1	1	1	1	1	1	1	
	b15	b14	b13	b12	b11	b10	b9	b8	
Value after reset	1	1	1	1	1	1	1	1	
	b7	b6	b5	b4	b3	b2	b1	b0	
Value after reset	1	1	1	1	1	1	1	1	
Bit	Symbol	Bit name	Function						R/W
b[31:0]	TXQxEMP	TXQ empty Status	1'b0: TXQ not empty 1'b1: TXQ empty						R

The TXQ status register bits show the status of the Empty bits of the TXQ Buffers..

4.3.66.1 CFDTXQESTS.TXQxEMP ($x = (n+1) * \text{TXQ Number} = 0 \text{ to } (((n+1)*4 - 1))$)

TXQ empty Status

Each bit is set automatically when the corresponding bit is set in the TXQ Status Registers.

Each bit is cleared automatically when the corresponding bit is cleared in the TXQ Status Registers.

This bit is set when RS-CAN-FD module enters GL_RESET mode.

Bit position	Corresponding TXQueue
0	channel 0 TX Queue0
1	channel 0 TX Queue1
2	channel 0 TX Queue2
3	channel 0 TX Queue3
4	channel 1 TX Queue0
5	channel 1 TX Queue1
:	:
26	channel 6 TX Queue2
27	channel 6 TX Queue3
28	channel 7 TX Queue0
29	channel 7 TX Queue1
30	channel 7 TX Queue2
31	channel 7 TX Queue3

4.3.67 CFDTXQFISTS

TX Queue Full Interrupt Status Registers

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
	-	TXQxFULL[30:28]			-	TXQxFULL[26:24]		
Value after reset	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset	-	TXQxFULL[22:20]			-	TXQxFULL[18:16]		
Value after reset	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
Value after reset	-	TXQxFULL[14:12]			-	TXQxFULL[10:8]		
Value after reset	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset	-	TXQxFULL[6:4]			-	TXQxFULL[2:0]		
Value after reset	0	0	0	0	0	0	0	0
Bit	Symbol	Bit name	Function					R/W
b[31]	-	Reserved	This bit is read as 0b. The write value should be always 0b.					R/W
b[30:28]	TXQxFULL[30:28]	TXQ Full Interrupt Status	1'b0: TXQ Full Interrupt is not set 1'b1: TXQ Full Interrupt is set					R
b[27]	-	Reserved	This bit is read as 0b. The write value should be always 0b.					R/W
b[26:24]	TXQxFULL[26:24]	TXQ Full Interrupt Status	1'b0: TXQ Full Interrupt is not set 1'b1: TXQ Full Interrupt is set					R
b[23]	-	Reserved	This bit is read as 0b. The write value should be always 0b.					R/W
b[22:20]	TXQxFULL[22:20]	TXQ Full Interrupt Status	1'b0: TXQ Full Interrupt is not set 1'b1: TXQ Full Interrupt is set					R
b[19]	-	Reserved	This bit is read as 0b. The write value should be always 0b.					R/W
b[18:16]	TXQxFULL[18:16]	TXQ Full Interrupt Status	1'b0: TXQ Full Interrupt is not set 1'b1: TXQ Full Interrupt is set					R
b[15]	-	Reserved	This bit is read as 0b. The write value should be always 0b.					R/W
b[14:12]	TXQxFULL[14:12]	TXQ Full Interrupt Status	1'b0: TXQ Full Interrupt is not set 1'b1: TXQ Full Interrupt is set					R
b[11]	-	Reserved	This bit is read as 0b. The write value should be always 0b.					R/W
b[10:8]	TXQxFULL[10:8]	TXQ Full Interrupt Status	1'b0: TXQ Full Interrupt is not set 1'b1: TXQ Full Interrupt is set					R
b[7]	-	Reserved	This bit is read as 0b. The write value should be always 0b.					R/W
b[6:4]	TXQxFULL[6:4]	TXQ Full Interrupt Status	1'b0: TXQ Full Interrupt is not set 1'b1: TXQ Full Interrupt is set					R
b[3]	-	Reserved	This bit is read as 0b. The write value should be always 0b.					R/W
b[2:0]	TXQxFULL[2:0]	TXQ Full Interrupt Status	1'b0: TXQ Full Interrupt is not set 1'b1: TXQ Full Interrupt is set					R

The TXQ status register bits show the status of the Full Interrupt bits of the TXQ Buffers.

4.3.67.1 CFDTXQFISTS.TXQxFULL (x = (n+1) * TXQ Number = 0 to (((n+1)*4 - 1))

TXQ Full Status

Each bit is set automatically when the corresponding bit is set in the TXQ Status Registers.

Each bit is cleared automatically when the corresponding bit is cleared in the TXQ Status Registers.

This bit is cleared when RS-CAN-FD module enters GL_RESET mode.

Bit position	Corresponding TXQueue
0	channel 0 TX Queue0
1	channel 0 TX Queue1
2	channel 0 TX Queue2
3	reserve
4	channel 1 TX Queue0
5	channel 1 TX Queue1
:	:
26	channel 6 TX Queue2
27	reserve
28	channel 7 TX Queue0
29	channel 7 TX Queue1
30	channel 7 TX Queue2
31	reserve

4.3.68 CFDTXQMSTS

TX Queue Message lost Status Registers

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
Value after reset	-		TXQxML [30:28]		-		TXQxML [26:24]	
	0	0	0	0	0	0	0	0
Value after reset	b23	b22	b21	b20	b19	b18	b17	b16
	-		TXQxML [22:20]		-		TXQxML [18:16]	
	0	0	0	0	0	0	0	0
Value after reset	b15	b14	b13	b12	b11	b10	b9	b8
	-		TXQxML [14:12]		-		TXQxML [10:8]	
	0	0	0	0	0	0	0	0
Value after reset	b7	b6	b5	b4	b3	b2	b1	b0
	-		TXQxML [6:4]		-		TXQxML [2:0]	
	0	0	0	0	0	0	0	0
Bit	Symbol	Bit name	Function					R/W
b[31]	-	Reserved	This bit is read as 0b. The write value should be always 0b.					R/W
b[30:28]	TXQxML [30:28]	TXQ message lost Status	1'b0: TXQ message lost flag is not set 1'b1: TXQ message lost flag is set					R
b[27]	-	Reserved	This bit is read as 0b. The write value should be always 0b.					R/W
b[26:24]	TXQxML [26:24]	TXQ message lost Status	1'b0: TXQ message lost flag is not set 1'b1: TXQ message lost flag is set					R
b[23]	-	Reserved	This bit is read as 0b. The write value should be always 0b.					R/W
b[22:20]	TXQxML [22:20]	TXQ message lost Status	1'b0: TXQ message lost flag is not set 1'b1: TXQ message lost flag is set					R
b[19]	-	Reserved	This bit is read as 0b. The write value should be always 0b.					R/W
b[18:16]	TXQxML [18:16]	TXQ message lost Status	1'b0: TXQ message lost flag is not set 1'b1: TXQ message lost flag is set					R
b[15]	-	Reserved	This bit is read as 0b. The write value should be always 0b.					R/W
b[14:12]	TXQxML [14:12]	TXQ message lost Status	1'b0: TXQ message lost flag is not set 1'b1: TXQ message lost flag is set					R
b[11]	-	Reserved	This bit is read as 0b. The write value should be always 0b.					R/W
b[10:8]	TXQxML [10:8]	TXQ message lost Status	1'b0: TXQ message lost flag is not set 1'b1: TXQ message lost flag is set					R
b[7]	-	Reserved	This bit is read as 0b. The write value should be always 0b.					R/W
b[6:4]	TXQxML [6:4]	TXQ message lost Status	1'b0: TXQ message lost flag is not set 1'b1: TXQ message lost flag is set					R
b[3]	-	Reserved	This bit is read as 0b. The write value should be always 0b.					R/W
b[2:0]	TXQxML [2:0]	TXQ message lost Status	1'b0: TXQ message lost flag is not set 1'b1: TXQ message lost flag is set					R

The TXQ status register bits show the status of the Message lost bits of the TXQ Buffers.

4.3.68.1 CFDTXQMSTS.TXQxML ($x = (n+1) * \text{TXQ Number} = 0 \text{ to } ((n+1)*4 - 1)$)

TXQ message lost Status

Each bit is set automatically when the corresponding bit is set in the TXQ Status Registers.

Each bit is cleared automatically when the corresponding bit is cleared in the TXQ Status Registers.
This bit is cleared when RS-CAN-FD module enters GL_RESET mode.

Bit position	Corresponding TXQueue
0	channel 0 TX Queue0
1	channel 0 TX Queue1
2	channel 0 TX Queue2
3	reserve
4	channel 1 TX Queue0
5	channel 1 TX Queue1
:	:
26	channel 6 TX Queue2
27	reserve
28	channel 7 TX Queue0
29	channel 7 TX Queue1
30	channel 7 TX Queue2
31	reserve

4.3.69 CFDTXQOWSTS

TX Queue Message Overwrite Status Registers

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24	
	TXQxOW[31:24]								
Value after reset	0	0	0	0	0	0	0	0	
	b23	b22	b21	b20	b19	b18	b17	b16	
Value after reset	0	0	0	0	0	0	0	0	
	b15	b14	b13	b12	b11	b10	b9	b8	
Value after reset	0	0	0	0	0	0	0	0	
	b7	b6	b5	b4	b3	b2	b1	b0	
Value after reset	0	0	0	0	0	0	0	0	
	Bit	Symbol	Bit name	Function					R/W
b[31:0]	TXQxOW	TXQ message overwrite Status		1'b0: TXQ message overwrite flag is not set 1'b1: TXQ message overwrite flag is set					R

The TXQ status register bits show the status of the Message overwrite bits of the TXQ Buffers.

4.3.69.1 CFDTXQOWSTS.TXQxOW (x = (n+1) * TXQ Number = 0 to (((n+1)*4 -1))

TXQ message overwrite Status

Each bit is set automatically when the corresponding bit is set in the TXQ Status Registers.

Each bit is cleared automatically when the corresponding bit is cleared in the TXQ Status Registers.
This bit is cleared when RS-CAN-FD module enters GL_RESET mode.

Bit position	Corresponding TXQueue
0	channel 0 TX Queue0
1	channel 0 TX Queue1
2	channel 0 TX Queue2
3	channel 0 TX Queue3
4	channel 1 TX Queue0
5	channel 1 TX Queue1
:	:
26	channel 6 TX Queue2
27	channel 6 TX Queue3
28	channel 7 TX Queue0
29	channel 7 TX Queue1
30	channel 7 TX Queue2
31	channel 7 TX Queue3

4.3.70 CFDTXQISTS

TX Queue Interrupt Status Registers

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24	
	TXQxISF[31:24]								
Value after reset	0	0	0	0	0	0	0	0	
	b23	b22	b21	b20	b19	b18	b17	b16	
	TXQxISF[23:16]								
Value after reset	0	0	0	0	0	0	0	0	
	b15	b14	b13	b12	b11	b10	b9	b8	
	TXQxISF[15:8]								
Value after reset	0	0	0	0	0	0	0	0	
	b7	b6	b5	b4	b3	b2	b1	b0	
	TXQxISF[7:0]								
Value after reset	0	0	0	0	0	0	0	0	
Bit	Symbol	Bit name	Function						R/W
b[31:0]	TXQxISF	TXQ Interrupt Status Flag	1'b0: TXQ Interrupt flag is not set 1'b1: TXQ Interrupt flag is set						R

The TXQ status register bits show the status of the Interrupt flag bits of the TXQ Buffers.

4.3.70.1 CFDTXQISTS.TXQxISF (x = (n+1) * TXQ Number = 0 to (((n+1)*4 -1))

TXQ Interrupt Status Flag

Each bit is set automatically when the corresponding bit is set in the TXQ Status Registers.

Each bit is cleared automatically when the corresponding bit is cleared in the TXQ Status Registers.
This bit is cleared when RS-CAN-FD module enters GL_RESET mode.

Bit position	Corresponding TXQueue
0	channel 0 TX Queue0
1	channel 0 TX Queue1
2	channel 0 TX Queue2
3	channel 0 TX Queue3
4	channel 1 TX Queue0
5	channel 1 TX Queue1
:	:
26	channel 6 TX Queue2
27	channel 6 TX Queue3
28	channel 7 TX Queue0
29	channel 7 TX Queue1
30	channel 7 TX Queue2
31	channel 7 TX Queue3

4.3.71 CFDTXQOFTISTS

TX Queue One Frame TX Interrupt Status Registers

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24	
	TXQxOFTISF[31:24]								
Value after reset	0	0	0	0	0	0	0	0	
	b23	b22	b21	b20	b19	b18	b17	b16	
	TXQxOFTISF[23:16]								
Value after reset	0	0	0	0	0	0	0	0	
	b15	b14	b13	b12	b11	b10	b9	b8	
	TXQxOFTISF[15:8]								
Value after reset	0	0	0	0	0	0	0	0	
	b7	b6	b5	b4	b3	b2	b1	b0	
	TXQxOFTISF[7:0]								
Value after reset	0	0	0	0	0	0	0	0	
Bit	Symbol	Bit name	Function						R/W
b[31:0]	TXQxOFTISF	TXQ One Frame TX Interrupt Status Flag	1'b0: TXQ One Frame TX Interrupt flag is not set 1'b1: TXQ One Frame TX Interrupt flag is set						R

The TXQ status register bits show the status of the One Frame TX Interrupt flag bits of the TXQ Buffers.

4.3.71.1 CFDTXQOFTISTS.TXQxOFTISF ($x = (n+1) * \text{TXQ Number} = 0 \text{ to } (((n+1)*4 - 1))$)

TXQ One Frame TX Interrupt Status Flag

Each bit is set automatically when the corresponding bit is set in the TXQ Status Registers.

Each bit is cleared automatically when the corresponding bit is cleared in the TXQ Status Registers.
This bit is cleared when RS-CAN-FD module enters GL_RESET mode.

Bit position	Corresponding TXQueue
0	channel 0 TX Queue0
1	channel 0 TX Queue1
2	channel 0 TX Queue2
3	channel 0 TX Queue3
4	channel 1 TX Queue0
5	channel 1 TX Queue1
:	:
26	channel 6 TX Queue2
27	channel 6 TX Queue3
28	channel 7 TX Queue0
29	channel 7 TX Queue1
30	channel 7 TX Queue2
31	channel 7 TX Queue3

4.3.72 CFDTXQOFRISTS

TX Queue One Frame RX Interrupt Status Registers

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
	-	TXQxOFRISF[30:28]			-	TXQxOFRISF[26:24]		
Value after reset	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
	-	TXQxOFRISF[22:20]			-	TXQxOFRISF[18:16]		
Value after reset	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
	-	TXQxOFRISF[14:12]			-	TXQxOFRISF[10:8]		
Value after reset	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
	-	TXQxOFRISF[6:4]			-	TXQxOFRISF[2:0]		
Value after reset	0	0	0	0	0	0	0	0
Bit	Symbol	Bit name	Function					R/W
b[31]	-	Reserved	This bit is read as 0b. The write value should be always 0b					R/W
b[30:28]	TXQxOFRISF [30:28]	TXQ One Frame RX Interrupt Status Flag	1'b0: TXQ One Frame RX Interrupt flag is not set 1'b1: TXQ One Frame RX Interrupt flag is set					R
b[27]	-	Reserved	This bit is read as 0b. The write value should be always 0b.					R/W
b[26:24]	TXQxOFRISF [26:24]	TXQ One Frame RX Interrupt Status Flag	1'b0: TXQ One Frame RX Interrupt flag is not set 1'b1: TXQ One Frame RX Interrupt flag is set					R
b[23]	-	Reserved	This bit is read as 0b. The write value should be always 0b					R/W
b[22:20]	TXQxOFRISF [22:20]	TXQ One Frame RX Interrupt Status Flag	1'b0: TXQ One Frame RX Interrupt flag is not set 1'b1: TXQ One Frame RX Interrupt flag is set					R
b[19]	-	Reserved	This bit is read as 0b. The write value should be always 0b					R/W
b[18:16]	TXQxOFRISF [18:16]	TXQ One Frame RX Interrupt Status Flag	1'b0: TXQ One Frame RX Interrupt flag is not set 1'b1: TXQ One Frame RX Interrupt flag is set					R
b[15]	-	Reserved	This bit is read as 0b. The write value should be always 0b					R/W
b[14:12]	TXQxOFRISF [14:12]	TXQ One Frame RX Interrupt Status Flag	1'b0: TXQ One Frame RX Interrupt flag is not set 1'b1: TXQ One Frame RX Interrupt flag is set					R
b[11]	-	Reserved	This bit is read as 0b. The write value should be always 0b					R/W
b[10:8]	TXQxOFRISF [10:8]	TXQ One Frame RX Interrupt Status Flag	1'b0: TXQ One Frame RX Interrupt flag is not set 1'b1: TXQ One Frame RX Interrupt flag is set					R
b[7]	-	Reserved	This bit is read as 0b. The write value should be always 0b					R/W
b[6:4]	TXQxOFRISF [6:4]	TXQ One Frame RX Interrupt Status Flag	1'b0: TXQ One Frame RX Interrupt flag is not set 1'b1: TXQ One Frame RX Interrupt flag is set					R
b[3]	-	Reserved	This bit is read as 0b. The write value should be always 0b					R/W
b[2:0]	TXQxOFRISF [2:0]	TXQ One Frame RX Interrupt Status Flag	1'b0: TXQ One Frame RX Interrupt flag is not set 1'b1: TXQ One Frame RX Interrupt flag is set					R

The TXQ status register bits show the status of the One Frame RX Interrupt flag bits of the TXQ Buffers.

4.3.72.1 CFDTXQOFRISTS.TXQxOFRISF ($x = (n+1) * \text{TXQ Number} = 0 \text{ to } ((n+1)*4 - 1)$)

TXQ One Frame RX Interrupt Status Flag

Each bit is set automatically when the corresponding bit is set in the TXQ Status Registers.

Each bit is cleared automatically when the corresponding bit is cleared in the TXQ Status Registers.

This bit is cleared when RS-CAN-FD module enters GL_RESET mode.

Bit position	Corresponding TXQueue
0	channel 0 TX Queue0
1	channel 0 TX Queue1
2	channel 0 TX Queue2
3	reserve
4	channel 1 TX Queue0
5	channel 1 TX Queue1
:	:
26	channel 6 TX Queue2
27	reserve
28	channel 7 TX Queue0
29	channel 7 TX Queue1
30	channel 7 TX Queue2
31	reserve

4.3.73 CFDTXQFSTS

TX Queue Full Status Registers

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24	
	TXQxFSF[31:24]								
Value after reset	0	0	0	0	0	0	0	0	
	b23	b22	b21	b20	b19	b18	b17	b16	
Value after reset	0	0	0	0	0	0	0	0	
	b15	b14	b13	b12	b11	b10	b9	b8	
Value after reset	0	0	0	0	0	0	0	0	
	b7	b6	b5	b4	b3	b2	b1	b0	
Value after reset	0	0	0	0	0	0	0	0	
Bit	Symbol	Bit name	Function						R/W
b[31:0]	TXQxFSF	TXQ Full Status Flag	1'b0: TXQ Full flag is not set 1'b1: TXQ Full flag is set						R

The TXQ status register bits show the status of the Full Status flag bits of the TXQ Buffers.

4.3.73.1 CFDTXQFSTS.TXQxFSF (x = (n+1) * TXQ Number = 0 to (((n+1)*4 -1))

TXQ Full Status Flag

Each bit is set automatically when the corresponding bit is set in the TXQ Status Registers.

Each bit is cleared automatically when the corresponding bit is cleared in the TXQ Status Registers.
This bit is cleared when RS-CAN-FD module enters GL_RESET mode.

Bit position	Corresponding TXQueue
0	channel 0 TX Queue0
1	channel 0 TX Queue1
2	channel 0 TX Queue2
3	channel 0 TX Queue3
4	channel 1 TX Queue0
5	channel 1 TX Queue1
:	:
26	channel 6 TX Queue2
27	channel 6 TX Queue3
28	channel 7 TX Queue0
29	channel 7 TX Queue1
30	channel 7 TX Queue2
31	channel 7 TX Queue3

4.3.74 CFDTHLCCn

TX History List Configuration / Control Register n

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
	-	-	-	-	-	-	-	THLEIE
Value after reset	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
	-	-	-	-	-	-	-	-
Value after reset	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
	-	-	-	-	THLDGE	THLDTE	THLIM	THLIE
Value after reset	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
	-	-	-	-	-	-	-	THLE
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:25]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[24]	THLEIE	TX History List Entry Lost Interrupt Enable	1'b0: TX History List Entry Lost Interrupt Disabled 1'b1: TX History List Entry Lost Interrupt Enabled	R/W
b[23:12]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[11]	THLDGE	TX History List Dedicated GW Enable	1'b0: Not dedicate Gateway FIFO + Gateway TX Queue 1'b1: Dedicate Gateway FIFO + Gateway TX Queue	R/W
b[10]	THLDTE	TX History List Dedicated TX Enable	1'b0: TX FIFO + TX Queue 1'b1: Flat TX MB + TX FIFO + TX Queue	R/W
b[9]	THLIM	TX History List Interrupt Mode	1'b0: Interrupt generated if TX History List level reaches ¾ of the TX History List depth. 1'b1: Interrupt generated for every successfully stored entry	R/W
b[8]	THLIE	TX History List Interrupt Enable	1'b0: TX History List Interrupt disabled 1'b1: TX History List Interrupt enabled	R/W
b[7:1]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[0]	THLE	TX History List Enable	1'b0: TX History List disabled 1'b1: TX History List enabled	R/W

The TX History List Configuration / Control register configures the TX History List functions.

4.3.74.1 CFDTHLCCn.THLE

TX History List Enable

This bit enables the TX History List Buffer when it is set.

Users cannot write to this bit when the related RS-CAN-FD channel is in CH_RESET or CH_SLEEP mode.

This bit is cleared automatically when the related RS-CAN-FD channel is in CH_RESET mode and **CFDGFFIMC.FFIEN** is 1'b0.

This bit is cleared automatically when the RS-CAN-FD is in GL_RESET mode and **CFDGFFIMC.FFIEN** is 1'b1.

4.3.74.2 CFDTHLCCn.THLIE

TX History List Interrupt Enable

This bit enables the generation of the TX History List Interrupt when it is set.

Users cannot write to this bit when RS-CAN-FD module is in GL_SLEEP mode.

4.3.74.3 CFDTHLCCn.THLIM

TX History List Interrupt Mode

This bit selects the Interrupt generation condition for the FIFO.

Users cannot write to this bit when RS-CAN-FD module is in GL_SLEEP mode.

Users should not write to this bit when RS-CAN-FD module is in GL_HALT or GL_OPERATION mode.

4.3.74.4 CFDTHLCCn.THLDTE

TX History List Dedicated TX Enable

This bit selects the conditions for storing an entry in the TX History list after successful transmission.

Users cannot write to this bit when RS-CAN-FD module is in GL_SLEEP mode.

Users should not write to this bit when RS-CAN-FD module is in GL_HALT or GL_OPERATION mode.

4.3.74.5 CFDTHLCCn.THLDGE

TX History List Dedicated GW Enable

This bit selects the conditions for storing an entry in the TX History list after successful transmission.

Users cannot write to this bit when RS-CAN-FD module is in GL_SLEEP mode.

Users should not write to this bit when RS-CAN-FD module is in GL_HALT or GL_OPERATION mode.

4.3.74.6 CFDTHLCCn.THLEIE

TX History List Entry Lost Interrupt Enable

This bit is used instead of **CFDGCTR.THLEIE** when **CFDGFFIMC.FFIEN** is 1'b1 and **CFDGVMEIS.THLELT** is 1.

User can only access to this bit when **CFDGFFIMC.FFIEN** is 1'b1.

Users cannot write to this bit when RS-CAN-FD module is in GL_SLEEP mode.

4.3.75 CFDTHLSTS_n

TX History List Status Register n

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24	
Value after reset	-	-	-	-	-	-	-	-	
	0	0	0	0	0	0	0	0	
	b23	b22	b21	b20	b19	b18	b17	b16	
Value after reset	-	-	-	-	-	-	-	-	
	0	0	0	0	0	0	0	0	
	b15	b14	b13	b12	b11	b10	b9	b8	
Value after reset	-	-	-	-	THLMC[5:0]				
	0	0	0	0	0	0	0	0	
	b7	b6	b5	b4	b3	b2	b1	b0	
Value after reset	-	-	-	-	THLIF	THLELT	THLFLL	THLEMP	
	0	0	0	0	0	0	0	1	
Bit	Symbol	Bit name	Function						R/W
b[31:14]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[13:8]	THLMC	TX History List Message Count	No. of Messages stored in TX History List						R
b[7:4]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[3]	THLIF	TX History List Interrupt Flag	1'b0: TX History List Interrupt condition not satisfied 1'b1: TX History List Interrupt condition satisfied						R/W
b[2]	THLELT	TX History List Entry Lost	1'b0: No Entry Lost in TX History List 1'b1: TX History List Entry Lost						R/W
b[1]	THLFLL	TX History List Full	1'b0: TX History List Not Full 1'b1: TX History List Full						R
b[0]	THLEMP	TX History List Empty	1'b0: TX History List Not Empty 1'b1: TX History List Empty						R

The TX History List Status register shows the status of the data stored in the TX History List Buffer.

4.3.75.1 CFDTHLSTS_n.THLEMP

TX History List Empty

This bit is set automatically when CPU has read all the entries from the TX History List Buffer.

This bit is set automatically when TX History List is disabled.

This bit is set automatically when the corresponding CAN channel enters CH_RESET mode and **CFDGFFIMC.FFIEN** is 1'b0.

This bit is cleared automatically when the first entry is stored to the TX History List.

This bit is set automatically when the RS-CAN-FD is in GL_RESET mode and **CFDGFFIMC.FFIEN** is 1'b1.

4.3.75.2 CFDTHLSTS_n.THLFLL

TX History List Full

Each TX History List can store up to 32 entries (each channel has a dedicated TX History List).

This bit is set automatically when the number of entries in the TX History List Buffer matches the TX History List depth.

This bit is cleared automatically when the number of entries in the TX History List Buffer is less than the TX History List depth.

This bit is cleared automatically when TX History List is disabled.

This bit is cleared automatically when the corresponding CAN channel enters CH_RESET mode and **CFDGFFIMC.FFIEN** is 1'b0.

This bit is cleared automatically when the RS-CAN-FD is in GL_RESET mode and **CFDGFFIMC.FFIEN** is 1'b1.

4.3.75.3 CFDTHLSTS_n.THLELT

TX History List Entry Lost

Users can write to this bit only when RS-CAN-FD module is in CH_HALT or CH_OPERATION modes. Writing 1'b1 has no influence on the bit values.

Do not use bit clear instruction for clearing this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Then the other bits remain set to 1'b1.

This bit is set when a new Entry cannot be stored as the related TX History List Buffer is already full.

This bit is cleared by writing 1'b0 to it.

This bit is automatically cleared in CH_RESET mode and **CFDGFFIMC.FFIEN** is 1'b0.

This bit is cleared automatically when the RS-CAN-FD is in GL_RESET mode and **CFDGFFIMC.FFIEN** is 1'b1.

4.3.75.4 CFDTHLSTS_n.THLIF

TX History List Interrupt Flag

Users can write to this bit only when RS-CAN-FD module is in CH_HALT or CH_OPERATION modes. Writing 1'b1 has no influence on the bit values.

Do not use bit clear instruction for clearing this bit. Use the MOV instruction to ensure that only the specified bit is cleared. Then the other bits remain set to 1'b1.

This bit is set when the configured Interrupt condition is satisfied.

The bit is cleared by writing 1'b0 to it.

This bit is automatically cleared in CH_RESET mode and **CFDGFFIMC.FFIEN** is 1'b0.

This bit is cleared automatically when the RS-CAN-FD is in GL_RESET mode and **CFDGFFIMC.FFIEN** is 1'b1.

4.3.75.5 CFDTHLSTS_n.THLMC

TX History List Message Count

These bits show the number of transmitted messages stored in the TX History List.

These bits are cleared automatically when the related channel is in CH_RESET mode **and** **CFDGFFIMC.FFIEN** is 1'b0.

This bit is cleared automatically when the RS-CAN-FD is in GL_RESET mode and **CFDGFFIMC.FFIEN** is 1'b1.

4.3.76 CFDTHLACC0n

Channel n TX History List Access Registers0

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24	
TMTS[15:8]									
Value after reset	0	0	0	0	0	0	0	0	
TMTS[7:0]									
Value after reset	0	0	0	0	0	0	0	0	
TGW CH[2:0] - - BN[6:5]									
Value after reset	0	0	0	0	0	0	0	0	
b7 b6 b5 b4 b3 b2 b1 b0									
Value after reset	0	0	0	0	0	0	0	0	
Bit	Symbol	Bit name	Function						R/W
b[31:16]	TMTS	Transmit Timestamp	Transmit Timestamp value for SW drivers						R
b[15]	TGW	Transmit Gateway Buffer indication	1'b0: not transmission from Gateway 1'b1: transmission from Gateway						R
b[14:12]	CH	Channel No.	Number of the Channel for FFI Mode						R
b[11:10]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[9:3]	BN	Buffer No.	Number of the Message Buffer						R
b[2:0]	BT	Buffer Type	3'b001: Flat TX Message Buffer 3'b010: TX FIFO MB No and GW FIFO MB No. 3'b100: TX Queue MB No.						R

The TX History List Access register provides access to the entry in the TX History List based on the Read Timestamp value.

4.3.76.1 CFDTHLACC0n.BT

Buffer Type

This bit indicates if the data has been stored following a transmission from a FIFO, a TX Queue or a TX MB.

4.3.76.2 CFDTHLACC0n.BN

Buffer No.

These bits show the Message Buffer from which transmission was successfully completed. If a message from a Common FIFO is transmitted, then these bits show the Message Buffer that is linked to the Common FIFO for transmission.

4.3.76.3 CFDTHLACC0n.CH

Channel No.

This bit indicates the transmission Channel Number when **CFDGFFIMC.FFIEN** is 1'b1.

When **CFDGFFIMC.FFIEN** is 1'b0, the read value of this bit is 0.

4.3.76.4 CFDTHLACC0n.TGW

Transmit Gateway Buffer indication

This bit is automatically set to 1, when transmission is completed in GW mode.

4.3.76.5 CFDTHLACC0n.TMTS

Transmit Timestamp

These bits indicate the Timestamp for use by the SW drivers.

In the case of can_race_ts_en=1, the information as which **CFDTHLACC0n.TMTS** is inputted from can_race_ts is reflected.

b31	b30	b29	b28	b27	b26	b25	b24
0		CH[2:0]			TSIDCNT[8:5]		
b23	b22	b21	b20	b19	b18	b17	b16
		TSIDCNT[4:0]			0	0	0

4.3.77 CFDTHLACC1n

Channel n TX History List Access Registers1

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24	
Value after reset	-	-	-	-	-	-	-	-	
	0	0	0	0	0	0	0	0	
	b23	b22	b21	b20	b19	b18	b17	b16	
Value after reset	-	-	-	-	-	-	-	TIFL[1:0]	
	0	0	0	0	0	0	0	0	
	b15	b14	b13	b12	b11	b10	b9	b8	
Value after reset	-	-	-	-	TID[15:8]				
	0	0	0	0	0	0	0	0	
	b7	b6	b5	b4	b3	b2	b1	b0	
Value after reset	-	-	-	-	TID[7:0]				
	0	0	0	0	0	0	0	0	
Bit	Symbol	Bit name	Function						R/W
b[31:18]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[17:16]	TIFL	Transmit Information Label	MB information label or TX FIFO information label or AFL information label is stored for SW drivers						R
b[15:0]	TID	Transmit ID	MB reference ID or TX FIFO references ID or AFL pointer field is stored for SW drivers.						R

The TX History List Access register1 provides access to the entry in the TX History List based on the Read Pointer value.

4.3.77.1 CFDTHLACC1n.TID

Transmit ID

These bits indicate the MB reference ID (**CFDTMFDCTR.TMPTR**) or the TX FIFO reference ID (**CFDCFFDCSTS.CFPTR**) for use by the SW drivers.

In a case of transmission in GW mode, these bits indicate the AFL pointer field (**CFDGAFLP0r.GAFLPTR**) instead of the MB reference ID (**CFDTMFDCTR.TMPTR**).

4.3.77.2 CFDTHLACC1n.TIFL

Transmit Information Label

These bits indicate the MB information label (**CFDTMFDCTR.TMIFL**) or the TX FIFO information label (**CFDCFFDCSTS.CFIFL**) for use by the SW drivers.

In a case of transmission in GW mode, these bits indicate the AFL pointer field (**CFDGAFLMr.GAFLIFL1**, **CFDGAFLP0r.GAFLIFL0**) instead of the MB information label (**CFDTMFDCTR.TMIFL**).

4.3.78 CFDTHLPCTRn

TX History List Pointer Control Registers

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24	
Value after reset	-	-	-	-	-	-	-	-	
	0	0	0	0	0	0	0	0	
	b23	b22	b21	b20	b19	b18	b17	b16	
Value after reset	-	-	-	-	-	-	-	-	
	0	0	0	0	0	0	0	0	
	b15	b14	b13	b12	b11	b10	b9	b8	
Value after reset	-	-	-	-	-	-	-	-	
	0	0	0	0	0	0	0	0	
	b7	b6	b5	b4	b3	b2	b1	b0	
	THLPC[7:0]								
Value after reset	0	0	0	0	0	0	0	0	
Bit	Symbol	Bit name	Function						R/W
b[31:8]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[7:0]	THLPC	TX History List Pointer Control	Increments the write pointer to the TX History List in the corresponding channel						W

This register can be used to increment the Read Pointer of the TX History List.

4.3.78.1 CFDTHLPCTRn.THLPC

TX History List Pointer Control

When 8'hFF is written to these bits, the Read Pointer of the TX History List is moved to the next TX History List entry address.

Read value from these bits is always 8'h0.

Users can write to this bit only when RS-CAN-FD module is in CH_HALT or CH_OPERATION modes.

Users should only write 8'hFF to these registers when the corresponding TX History List is enabled and is not empty.

4.3.79 CFDGRSTC

Global Reset Control Register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
Value after reset	-	-	-	-	-	-	-	-
	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset	-	-	-	-	-	-	-	-
	b15	b14	b13	b12	b11	b10	b9	b8
Value after reset	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset	-	-	-	-	-	-	-	SRST

Bit	Symbol	Bit name	Function	R/W
b[31:16]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[15:8]	KEY	Key code	These bits control the right or wrong of rewriting of a SRST bit.	W
b[7:1]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[0]	SRST	SW reset	1'b0: normal state 1'b1: SW reset state	R/W

4.3.79.1 CFDGRSTC.SRST

SW Reset

When this bit is set, RSCAN-FD module will be in the same state as hardware reset.

When reset of IP is required, users should write 1 to this bit. Then user should write 0 to this bit.

When this bit is cleared, a RS-CAN-FD module is in GL_SLEEP mode.

When this bit is cleared, the RAM initialization sequence does not operate. The configuration of RAM is performed by SoftWare.

RAM is not initialized when software reset is performed during the initialization of RAM.

Software needs to perform the initialization of RAM.

4.3.79.2 CFDGRSTC.KEY

KEY code

When C4h is written in these bits, the write of a **CFDGRSTC.SRST** bit becomes available.

Read value from these bits is always 8'h0.

Users should write a **CFDGRSTC.SRST** bit and the **CFDGRSTC.KEY** bit simultaneously.

4.3.80 CFDGFCMC

Global Flexible CAN mode Configuration Register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
	-	-	-	-	-	-	-	-
Value after reset	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
	-	-	-	-	-	-	-	-
Value after reset	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
	-	-	-	-	-	-	-	-
Value after reset	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
	-	-	-	-	FLXC3	FLXC2	FLXC1	FLXC0
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:4]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[3]	FLXC3	Flexible CAN mode between Channel 6 and Channel 7	1'b0: Normal mode 1'b1: Flexible CAN mode	R/W
b[2]	FLXC2	Flexible CAN mode between Channel 4 and Channel 5	1'b0: Normal mode 1'b1: Flexible CAN mode	R/W
b[1]	FLXC1	Flexible CAN mode between Channel 2 and Channel 3	1'b0: Normal mode 1'b1: Flexible CAN mode	R/W
b[0]	FLXC0	Flexible CAN mode between Channel 0 and Channel 1	1'b0: Normal mode 1'b1: Flexible CAN mode	R/W

Flexible transmission buffer assignment configured in **CFDGFTBAC** register and Flexible CAN mode configured in **CFDGFCMC** register should not be used simultaneously.

4.3.80.1 CFDGFCMC.FLXC0

Flexible CAN mode between Channel 0 and Channel 1

When this bit is set, Channel 0 and Channel 1 of a RS-CAN-FD module are Flexible CAN mode. Channel 1 uses TX/RX terminal of Channel 0. The TX/RX terminal of Channel 1 can not use.

Users can only write to this bit when RS-CAN-FD module is in GL_RESET mode.

4.3.80.2 CFDGFCMC.FLXC1

Flexible CAN mode between Channel 2 and Channel 3

When this bit is set, Channel 2 and Channel 3 of a RS-CAN-FD module are Flexible CAN mode. Channel 3 uses TX/RX terminal of Channel 2. The TX/RX terminal of Channel 3 can not use.

Users can only write to this bit when RS-CAN-FD module is in GL_RESET mode.

4.3.80.3 CFDGFCMC.FLXC2

Flexible CAN mode between Channel 4 and Channel 5

When this bit is set, Channel 4 and Channel 5 of a RS-CAN-FD module are Flexible CAN mode.

Channel 5 uses TX/RX terminal of Channel 4. The TX/RX terminal of Channel 5 can not use.

Users can only write to this bit when RS-CAN-FD module is in GL_RESET mode.

4.3.80.4 CFDGFCMC.FLXC3

Flexible CAN mode between Channel 6 and Channel 7

When this bit is set, Channel 6 and Channel 7 of a RS-CAN-FD module are Flexible CAN mode.
Channel 7 uses TX/RX terminal of Channel 6. The TX/RX terminal of Channel 7 can not use.

Users can only write to this bit when RS-CAN-FD module is in GL_RESET mode.

4.3.81 CFDGBISC

Global Bus Interface Select Configuration Register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
Value after reset	-	-	-	-	-	-	-	-
	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset	-	-	-	-	-	-	-	-
	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
	KEY[7:0]							
Value after reset	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
	-	-	-	-	-	-	-	IFSW
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:16]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[15:8]	KEY	Key code	These bits control the right or wrong of rewriting of IFSW bit.	W
b[7:1]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[0]	IFSW	Select BUS Interface APB or R-ACE	1'b0: select APB Interface 1'b1: select R-ACE Interface	R/W

4.3.81.1 CFDGBISC.IFSW

Select BUS Interface APB or R-ACE

When this bit is clear, the bus interface of a RS-CAN-FD module is APB.

When this bit is set, the bus interface of a RS-CAN-FD module is R-ACE.

Users can only write to this bit when RS-CAN-FD module is in GL_RESET mode.

This bit is not reset by SW reset (**CFDFRSTC.SRST** = 1b).

Only the CPU reset is resetting this bit.

Note: When this bit is read, the bit shows the combined status of the mode select input signals *rs_canfd_bus_if_sel* and *rs_canfd_bus_if_en* and set value of this bit. See Table 3.9 for the details.

4.3.81.2 CFDGBISC.KEY

KEY code

When C4h is written in these bits, the write of **CFDGBISC.IFSW** bit becomes available.

Read value from these bits is always 8'h0.

Users should write a **CFDGBISC.IFSW** bit and the **CFDGBISC.KEY** bit simultaneously.

4.3.82 CFDGFTBAC

Global Flexible transmission buffer assignment Configuration Register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
	-	-	-	-		FLXMB3[3:0]		
Value after reset	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
	-	-	-	-		FLXMB2[3:0]		
Value after reset	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
	-	-	-	-		FLXMB1[3:0]		
Value after reset	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
	-	-	-	-		FLXMB0[3:0]		
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:28]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[27:24]	FLXMB3	Flexible transmission buffer assignment between Channel 6 and Channel 7	By setting these bits the even channel can use the configured number of TX mailboxes of the odd channel. 4'b0000: 0 4'b0001: 4 4'b0010: 8 4'b0011: 12 4'b0100: 16 4'b0101: 20 4'b0110: 24 4'b0111: 28 4'b1000: 32	R/W
b[23:20]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[19:16]	FLXMB2	Flexible transmission buffer assignment between Channel 4 and Channel 5	By setting these bits the even channel can use the configured number of TX mailboxes of the odd channel. 4'b0000: 0 4'b0001: 4 4'b0010: 8 4'b0011: 12 4'b0100: 16 4'b0101: 20 4'b0110: 24 4'b0111: 28 4'b1000: 32	R/W
b[15:12]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[11:8]	FLXMB1	Flexible transmission buffer assignment between Channel 2 and Channel 3	By setting these bits the even channel can use the configured number of TX mailboxes of the odd channel. 4'b0000: 0 4'b0001: 4 4'b0010: 8 4'b0011: 12 4'b0100: 16 4'b0101: 20 4'b0110: 24 4'b0111: 28 4'b1000: 32	R/W
b[7:4]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[3:0]	FLXMB0	Flexible transmission buffer assignment	By setting these bits the even channel can use the configured number of TX mailboxes of the odd channel 4'b0000: 0	R/W

between Channel 0 and Channel 1	4'b0001: 4 4'b0010: 8 4'b0011: 12 4'b0100: 16 4'b0101: 20 4'b0110: 24 4'b0111: 28 4'b1000: 32
------------------------------------	--------------------------------------------------------------------------------------------------------------------

4.3.82.1 CFDGFTBAC.FLXMBa (a=0 to (n+1)/2-1) (n=1, 3, 5, 7)

Flexible transmission buffer assignment between Channel n-1 and Channel n

Channel n-1 can use the number TXMB of Channel n from 0 to 32 by the configuration of these bits.

Users can only write to this bit when RS-CAN-FD module is in GL_RESET mode.

Flexible transmission buffer assignment configured in **CFDGFTBAC** register and Flexible CAN mode configured in **CFDGFCMC** register should not be used simultaneously.

4.3.83 CFDGTSTCFG

Global Test Configuration Register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24	
Value after reset	PNFS	-	-	-	-	-	RTMPS[9:8]		
	0	0	0	0	0	0	0	0	
	b23	b22	b21	b20	b19	b18	b17	b16	
Value after reset	RTMPS[7:0]								
	0	0	0	0	0	0	0	0	
	b15	b14	b13	b12	b11	b10	b9	b8	
Value after reset	-	-	-	-	-	-	-	-	
	0	0	0	0	0	0	0	0	
	b7	b6	b5	b4	b3	b2	b1	b0	
Value after reset	CnICBCE[7:0]								
	0	0	0	0	0	0	0	0	
Bit	Symbol	Bit name	Function						R/W
b[31]	PNFS	Pretended Network Filter List RAM Select	1'b0: AFL RAM, MRAM area is selected 1'b1: Pretended Network Filter RAM area is selected						R/W
b[30:26]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[25:16]	RTMPS	RAM Test Mode Page Select	Select a RAM Test Mode page						R/W
b[15:8]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[7:0]	CnICBCE	Channel n Internal CAN Bus Communication Test Mode Enable	1'b0: Channel n internal CAN bus communication disabled 1'b1: Channel n internal CAN bus communication enabled						R/W

The Global Test Configuration Register is used to configure the CAN channels joining the internal CAN Bus Communication Test Mode and the RAM Test Mode Page.

4.3.83.1 CFDGTSTCFG.CnICBCE

Channel n Internal CAN Bus Communication Test Mode Enable

If this bit is set and RS-CAN-FD module is configured in Internal CAN Bus Communication Test Mode, then CAN channel n will participate to the Internal CAN Bus Communication Test Mode operation.

Users cannot write to this bit when the RS-CAN-FD module is in GL_RESET or GL_SLEEP mode.

Users should write to this bit only when RS-CAN-FD module is in GL_HALT mode.

These bits are cleared automatically when the RS-CAN-FD module is in GL_RESET mode.

4.3.83.2 CFDGTSTCFG.RTMPS

RAM Test Mode Page Select

These bits select the RAM Page mode for CPU read / write access when RS-CAN-FD module is configured in RAM Test mode.

Refer to Section 12.2.1 for RAM Test Mode Specification.

Users cannot write to these bits when the RS-CAN-FD module is in GL_RESET or GL_SLEEP mode.

Users should only enter values from 0 to 95 (10'h5F) for the AFL RAM and 96 to 868 (10'h364) for the MB RAM.

The setting range of these bits depends on the combination of parameters. For details, refer to 5.8 software constraint.

Users should write to this bit only when RS-CAN-FD module is in GL_HALT mode.

These bits are cleared automatically when the related RS-CAN-FD channel is in GL_RESET mode.

4.3.83.3 CFDGTSTCFG.PNFS

Pretended Network Filter List RAM Select

This bit chooses either AFLRAM+MRAM or PFLRAM (for Pretended Network filter) in RAM Test Mode

Users cannot write to this bit when the RS-CAN-FD module is in GL_RESET or GL_SLEEP mode.

Users should write to this bit only when RS-CAN-FD module is in GL_HALT mode.

This bit is cleared automatically when the RS-CAN-FD module is in GL_RESET mode.

4.3.84 CFDGTSTCTR

Global Test Control Register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24	
Value after reset	-	-	-	-	-	-	-	-	
	0	0	0	0	0	0	0	0	
	b23	b22	b21	b20	b19	b18	b17	b16	
Value after reset	-	-	-	-	-	-	-	-	
	0	0	0	0	0	0	0	0	
	b15	b14	b13	b12	b11	b10	b9	b8	
Value after reset	-	-	-	-	-	-	-	-	
	0	0	0	0	0	0	0	0	
	b7	b6	b5	b4	b3	b2	b1	b0	
Value after reset	-	-	-	-	-	RTME	-	ICBCTME	
	0	0	0	0	0	0	0	0	
Bit	Symbol	Bit name	Function						R/W
b[31:3]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[2]	RTME	RAM Test Mode Enable	1'b0: RAM Test Mode disabled 1'b1: RAM Test Mode enabled						R/W
b[1]	-	Reserved	This bit is read as 0b. The write value should be always 0b.						R/W
b[0]	ICBCTME	Internal CAN Bus Communication Test Mode Enable	1'b0: Internal CAN Bus communication test mode disabled 1'b1: Internal CAN Bus communication test mode enabled						R/W

The Global Test Control register is used to control the global test modes of the RS-CAN-FD module.

4.3.84.1 CFDGTSTCTR.ICBCTME

Internal CAN Bus Communication Test Mode Enable

If this bit is set, internal CAN Bus Communication is enabled for the CAN channels that are configured for Internal CAN Bus Communication participation. Refer to Section 12 for specification of Internal CAN Bus Communication Test Mode.

Users can set this bit only when RS-CAN-FD module is in GL_HALT mode.

Users should clear this bit when RS-CAN-FD module is in GL_HALT mode.

This bit is cleared automatically when RS-CAN-FD module enters GL_RESET mode.

4.3.84.2 CFDGTSTCTR.RTME

RAM Test Mode Enable

If this bit is set, RS-CAN-FD module is configured in RAM Test Mode. Refer to Section 12.2.1 for RAM Test Mode Specification.

Users can set this bit only when RS-CAN-FD module is in GL_HALT mode.

Users should clear this bit when RS-CAN-FD module is in GL_HALT mode.

This bit is cleared automatically when RS-CAN-FD module enters GL_RESET mode.

4.3.85 CFDGFDCFG

Global FD configuration register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
Value after reset	-	-	-	-	-	-	-	-
	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset	-	-	-	-	-	-	-	-
	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
Value after reset	-	-	-	-	-	-	TSCCFG	
	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset	-	-	-	-	-	-	-	RPED
	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:10]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[9:8]	TSCCFG	Timestamp capture configuration	2'b00: Timestamp capture at the sample point of SOF (start of frame) 2'b01: Timestamp capture at frame valid indication 2'b10: Timestamp capture at the sample point of RES bit 2'b11: reserved	R/W
b[7:1]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[0]	RPED	RES bit Protocol exception disable	1'b0: Protocol exception event detection enabled 1'b1: Protocol exception event detection disabled	R/W

4.3.85.1 CFDGFDCFG.RPED

RES bit Protocol exception state disable

This bit configures the protocol exception event handling according to ISO 11898-1. When this bit is enabled then the protocol exception event detection is disabled, and the protocol controller will transmit an error frame when the protocol exception event is detected (RES –bit is sampled recessive).

Users can set this bit only when RS-CAN-FD module is in GL_RESET mode.

4.3.85.2 CFDGFDCFG.TSCCFG

Timestamp capture configuration

These bits configure the different capture points of the timestamp; for transmission and reception. When **CFDGFDCFG.TSCCFG[1:0] = 2'b10** then the timestamp capture is done for CAN-FD frames at RES bit and for Classical frames at the start of frame.

Users can set these bits only when RS-CAN-FD module is in GL_RESET mode.

Users should set 2'b01 to these bits when can_race_ts_en = 1

4.3.86 CFDGCRCCFG

Global CRC configuration register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
Value after reset	-	-	-	-	-	-	-	-
	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset	-	-	-	-	-	-	-	-
	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
Value after reset	-	-	-	-	-	-	-	-
	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset	-	-	-	-	-	-	-	NIE
	0	0	0	0	0	0	0	0*

Bit	Symbol	Bit name	Function	R/W
b[31:1]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[0]	NIE	Non ISO enable	1'b0: CAN FD frame format according to ISO 11898-1 1'b1: CAN FD frame format according to Bosch CAN FD Specification V1.0	R/W

4.3.86.1 CFDGCRCCFG.NIE

Non ISO enable

This bit enables the non ISO conform CAN-FD frame format (Bosch CAN FD Specification V1.0 format). If this bit is disabled then the CAN-FD frame format according to ISO 11898-1 (2015) is used.

Note the both frame formats are incompatible to each other.

Users can set this bit only when RS-CAN-FD module is in GL_RESET mode.

**If this register is read, the result of having taken an external terminal (rs_canfd_non_iso_sel, rs_canfd_non_iso_en) and conditions will be read to register Output.*

Conditions become the information of table3.8.

4.3.87 CFDGLOCKK

Global Lock Key Register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
Value after reset	-	-	-	-	-	-	-	-
	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset	-	-	-	-	-	-	-	-
	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
Value after reset	-	-	-	-	-	-	-	-
	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset	-	-	-	-	-	-	-	-
	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:16]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[15:0]	LOCK	Lock Key	Key bits for unlocking the protection of test modes	W

The Global Lock Key register is a write only register that is used to unlock the protection for special test bits.

Refer to Section 12.2 for Lock Key specification.

4.3.87.1 CFDGLOCKK.LOCK

Lock Key

The Unlock Key sequence must be written in these bits to configure RS-CAN-FD module in FIFO OTB disable and RAM Test.

Read value from these bits is always 16'h0.

Users can not write to these bits when RS-CAN-FD module is in GL_SLEEP or GL_RESET mode.

Users should not write to these bits when RS-CAN-FD module is in GL_OPERATION mode.

4.3.88 CFDGLOTB

Global OTB FIFO Configuration / Status Register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
	-	-	-	-	-	-	-	-
Value after reset	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
	-	-	-	-	-	-	-	-
Value after reset	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
	<i>OTBMC[4:0]</i>					<i>OTBMLT</i>	<i>OTBFLL</i>	<i>OTBEMP</i>
Value after reset	0	0	0	0	0	0	0	1
	b7	b6	b5	b4	b3	b2	b1	b0
	-	-	-	-	-	-	-	<i>OTBFE</i>
Value after reset	0	0	0	0	0	0	0	1

Bit	Symbol	Bit name	Function	R/W
b[31:16]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[15:11]	OTBMC	OTB FIFO Message Count	Number of Messages stored in FIFO	R
b[10]	OTBMLT	OTB FIFO Message Lost	1'b0: No Message Lost in FIFO 1'b1: FIFO Message Lost	R/W
b[9]	OTBFLL	OTB FIFO Full	1'b0: FIFO Not Full 1'b1: FIFO Full	R
b[8]	OTBEMP	OTB FIFO Empty	1'b0: FIFO Not Empty 1'b1: FIFO Empty	R
b[7:1]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[0]	OTBFE	OTB FIFO Enable	1'b0: FIFO disabled 1'b1: FIFO enabled	R/W

4.3.88.1 CFDGLOTB.OTBFE

OTB FIFO Enable

This bit enables the OTB FIFO when it is set. If this bit is cleared, then the OTB FIFO pointers will be cleared and the OTB FIFO will become empty.

Users can clear this bit only when RS-CAN-FD module is in GL_HALT mode and the unlock key has been written.

Users can set this bit when RS-CAN-FD module is in GL_HALT and GL_OPERATION mode.

To clear the bit a special OTB FIFO disable Test Mode key sequence is necessary. Refer to Section 12.2.3 for OTB Test Mode Specification.

This bit will be set automatically in GL_RESET mode.

4.3.88.2 CFDGLOTB.OTBEMP

OTB FIFO Empty

This bit is set automatically when all the messages have been read out from the OTB FIFO.

This bit is set automatically when OTB FIFO is disabled by setting the OTBFE bit to 1'b0.

This bit is set automatically when RS-CAN-FD module enters GL_RESET mode.

This bit is cleared automatically when the first message is stored in the OTB FIFO Buffer.

4.3.88.3 CFDGLOTB.OTBFLL

OTB FIFO Full

This bit is set automatically when the number of Routing messages stored in the FIFO matches the max. FIFO depth. The max. FIFO depth is 2 messages per Channel.

This bit is cleared automatically when number of Routing messages stored in the FIFO is less than the max. FIFO depth.

This bit is cleared automatically when OTB FIFO is disabled by setting the OTBFE bit to 1'b0.

This bit is cleared automatically when RS-CAN-FD module enters GL_RESET mode.

4.3.88.4 CFDGLOTB.OTBMLT

OTB FIFO Message Lost

Users can write to this bit only when RS-CAN-FD module in GL_HALT or GL_OPERATION mode.

This bit is set automatically when Message is lost due to attempted storage of a new message when FIFO is already full.

The bit is cleared by writing 1'b0 to it.

The bit is cleared if RS-CAN-FD module enters GL_RESET mode.

4.3.88.5 CFDGLOTB.OTBMC

OTB FIFO Message Count

These bits indicate the number of CAN messages stored in the OTB FIFO that need to be routed.

These bits are cleared automatically when RS-CAN-FD module enters GL_RESET mode.

4.3.89 CFDRPGACCK

RAM Test Page Access Registers

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
RD TA[31:24]								
Value after reset	0	0	0	0	0	0	0	0
RD TA[23:16]								
Value after reset	0	0	0	0	0	0	0	0
RD TA[15:8]								
Value after reset	0	0	0	0	0	0	0	0
RD TA[7:0]								
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:0]	RD TA	RAM Data Test Access	RAM Data Bytes	R/W

(Average no. of entries for 1 channel = no_of_entries = 64)

(k = [0...no_of_entries-1])

4.3.89.1 CFDRPGACCK.RDTA

RAM Data Test Access

Data can be read from or written into these register bits when RS-CAN-FD module is configured in RAM Test Mode.

Users can only write to this bit when RS-CAN-FD module is in GL_HALT mode, and RAM test mode is enabled.

SW Data should be read / written in the RAM Test Page Access registers during RAM Test Mode.

4.3.90 CFDCnBLCT

BUS Load counter control Register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
Value after reset	-	-	-	-	-	-	-	-
	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset	-	-	-	-	-	-	-	-
	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
Value after reset	-	-	-	-	-	-	-	BLCLD
	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset	-	-	-	-	-	-	-	BLCE
	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:9]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[8]	BLCLD	BUS Load counter load	When CFDCnBLCT.BLCLD is set, it is reset after a BUS Load counter value is loaded	W
b[7:1]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[0]	BLCE	BUS Load counter Enable	1'b0: BUS Load counter disable 1'b1: BUS Load counter enable	R/W

4.3.90.1 CFDCnBLCT.BLCE

BUS Load counter enable

The enabling signal of a BUS Load counter.

When this bit is 0, BUS Load counter stops. But counter is not clear.

The bit is cleared when the related channel enters CH_RESET mode.

This bit cannot be written in CH_SLEEP mode.

Write in this bit, after setting up a **CFDCnNCFG** register.

4.3.90.2 CFDCnBLCT.BLCLD

BUS Load counter load

When **CFDCnBLCT.BLCLD** is set, the BUS Load counter value is loaded to **CFDCnBLSTS.BLC** and then the BUS Load counter is reset.

Read value is always 0.

4.3.91 CFDCnBLSTS

BUS Load counter Register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
BLC [31:24]								
Value after reset	0	0	0	0	0	0	0	0
BLC [23:16]								
Value after reset	0	0	0	0	0	0	0	0
BLC [15:8]								
Value after reset	0	0	0	0	0	0	0	0
BLC [7:3]								
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:3]	BLC	BUS Load counter Status	BUS Load counter value	R
b[2:0]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W

4.3.91.1 CFDCnBLSTS.BLC

BUS Load counter status

The bus load counter increases by clkc period while CAN bus is in an idle state.

When **CFDCnBLCT.BLCLD** bit is set, the BUS Load counter value is loaded to **BLC** bit and then the BUS Load counter is reset.

Bit2 to Bit0 is fixed 3'b0.

These bits are set only by RS-CAN-FD module.

Writing any value has no influence on these bits.

4.3.92 CFDGFFIMC

Global Freedom from Interference Mode Configuration Register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
Value after reset	-	-	-	-	-	-	-	-
	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset	-	-	-	-	-	-	-	-
	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
Value after reset	-	-	-	-	-	-	-	-
	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset	-	-	-	-	-	-	-	FFIEN
	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:16]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[15:8]	KEY	Key code	These bits control the right or wrong of rewriting of FFIEN bit.	W
b[7:1]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[0]	FFIEN	FFI Mode Enable	1'b0: FFI Mode disabled 1'b1: FFI Mode enabled	R/W

4.3.92.1 CFDGFFIMC.FFIEN

FFI Mode Enable

The FFI Mode is enabled if **CFDGFFIMC.FFIEN** is 1'b1.

Users can only write to this bit when RS-CAN-FD module is in GL_RESET mode.

4.3.92.2 CFDGFFIMC.KEY

Key code

When C4h is written in these bits, the write of **CFDGFFIMC.FFIEN** bit becomes available.

Read value from these bits are always 8'h0.

Users should write a **CFDGFFIMC.FFIEN** bit and the **CFDGFFIMC.KEY** bit simultaneously.

4.3.93 CFDGVMEIS

Global Virtual Machine Error Interrupt Select Register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
Value after reset	-	-	-	-	-	-	-	-
	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset	-	-	-	-	-	-	-	-
	b15	b14	b13	b12	b11	b10	b9	b8
Value after reset	-	-	-	-	-	-	-	-
	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset	-	-	TXQOW	CFMOW	-	THLELT	TXQMLT	FMLT
	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:6]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[5]	TXQOW	Selection of the output destination of TXQ message overwrite interrupt	1'b0: Select Global error interrupt 1'b1: Select VM error interrupt	R/W
b[4]	CFMOW	Selection of the output destination of COMFIFO message overwrite interrupt	1'b0: Select Global error interrupt 1'b1: Select VM error interrupt	R/W
b[3]	-	Reserved	This bit is read as 0b. The write value should be always 0b.	R/W
b[2]	THLELT	Selection of the output destination of THL entry lost interrupt	1'b0: Select Global error interrupt 1'b1: Select VM error interrupt	R/W
b[1]	TXQMLT	Selection of the output destination of TXQ message lost interrupt	1'b0: Select Global error interrupt 1'b1: Select VM error interrupt	R/W
b[0]	FMLT	Selection of the output destination of FIFO message lost interrupt	1'b0: Select Global error interrupt 1'b1: Select VM error interrupt	R/W

4.3.93.1 CFDGVMEIS.FMLT

Selection of the output destination of FIFO message lost interrupt

When this bit is clear, the FIFO message lost interrupt is output to Global error interrupt.

When this bit is set, the FIFO message lost interrupt is output to VM error interrupt.

When **CFDGFFIMC.FFIEN** is 1'b0, the read value of this bit is 0.

Users can only write to this bit when RS-CAN-FD module is in GL_RESET mode and **CFDGFFIMC.FFIEN** is 1'b1.

4.3.93.2 CFDGVMEIS.TXQMLT

Selection of the output destination of TXQ message lost interrupt

When this bit is clear, the TXQ message lost interrupt is output to Global error interrupt.

When this bit is set, the TXQ message lost interrupt is output to VM error interrupt.

When **CFDGFFIMC.FFIEN** is 1'b0, the read value of this bit is 0.

Users can only write to this bit when RS-CAN-FD module is in GL_RESET mode and **CFDGFFIMC.FFIEN** is 1'b1.

4.3.93.3 CFDGVMEIS.THLELT

Selection of the output destination of THL entry lost interrupt

When this bit is clear, the THL entry lost interrupt is output to Global error interrupt.

When this bit is set, the THL entry lost interrupt is output to VM error interrupt.

When **CFDGFFIMC.FFIEN** is 1'b0, the read value of this bit is 0.

Users can only write to this bit when RS-CAN-FD module is in GL_RESET mode and **CFDGFFIMC.FFIEN** is 1'b1.

4.3.93.4 CFDGVMEIS.CFMOW

Selection of the output destination of COMFIFO message overwrite interrupt

When this bit is clear, the COMFIFO message overwrite interrupt is output to Global error interrupt.

When this bit is set, the COMFIFO message overwrite interrupt is output to VM error interrupt.

When **CFDGFFIMC.FFIEN** is 1'b0, the read value of this bit is 0.

Users can only write to this bit when RS-CAN-FD module is in GL_RESET mode and **CFDGFFIMC.FFIEN** is 1'b1.

4.3.93.5 CFDGVMEIS.TXQOW

Selection of the output destination of TXQ message overwrite interrupt

When this bit is clear, the TXQ message overwrite interrupt is output to Global error interrupt.

When this bit is set, the TXQ message overwrite interrupt is output to VM error interrupt.

When **CFDGFFIMC.FFIEN** is 1'b0, the read value of this bit is 0.

Users can only write to this bit when RS-CAN-FD module is in GL_RESET mode and **CFDGFFIMC.FFIEN** is 1'b1.

4.3.94 CFDVMCFGn

Global Virtual Machine Common FIFO TXQ configuration Register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
	-	-	-	-		CF2VMN[3:0]		
Value after reset	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
		CF1VMN[3:0]			CF0VMN[3:0]			
Value after reset	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
		TXQ3VMN[3:0]			TXQ2VMN[3:0]			
Value after reset	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
		TXQ1VMN[3:0]			TXQ0VMN[3:0]			
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:28]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[27:24]	CF2VMN	VM number for COMFIFO2	Virtual machine which uses COMFIFO2 is specified.	R/W
b[23:20]	CF1VMN	VM number for COMFIFO1	Virtual machine which uses COMFIFO1 is specified.	R/W
b[19:16]	CF0VMN	VM number for COMFIFO0	Virtual machine which uses COMFIFO0 is specified.	R/W
b[15:12]	TXQ3VMN	VM number for TXQ3	Virtual machine which uses TXQ3 is specified.	R/W
b[11:8]	TXQ2VMN	VM number for TXQ2	Virtual machine which uses TXQ2 is specified.	R/W
b[7:4]	TXQ1VMN	VM number for TXQ1	Virtual machine which uses TXQ1 is specified.	R/W
b[3:0]	TXQ0VMN	VM number for TXQ0	Virtual machine which uses TXQ0 is specified.	R/W

Users can not write to this register when rs_vmnum_fix of an external terminal is 1'b1.

When rs_vmnum_fix of an external terminal is 1'b1, the read value of this register is fixed value shown below.

Register name	Read value (When rs_vmnum_fix is 1'b1 and CFDGFFIMC.FFIEN is 1'b1)	Read value (When CFDGFFIMC.FFIEN is 1'b0)
CFDVMCFG0	32'h0888_8888	32'h0000_0000
CFDVMCFG1	32'h0999_9999	32'h0000_0000
CFDVMCFG2	32'h0AAA_AAAA	32'h0000_0000
CFDVMCFG3	32'h0BBB_BBBB	32'h0000_0000
CFDVMCFG4	32'h0CCC_CCCC	32'h0000_0000
CFDVMCFG5	32'h0DDD_DDDD	32'h0000_0000
CFDVMCFG6	32'h0EEE_EEEE	32'h0000_0000
CFDVMCFG7	32'h0FFF_FFFF	32'h0000_0000

Assignment of a setting value and virtual machine is shown below.

bit value	VM assignment
4'b0xxx	invalid
4'b1000	VM0
4'b1001	VM1
4'b1010	VM2
4'b1011	VM3
4'b1100	VM4
4'b1101	VM5
4'b1110	VM6
4'b1111	VM7

4.3.94.1 CFDVMCFGn.TXQ0VMN

VM number for TXQ0

This bit specifies the virtual machine number which uses TXQ0.

When **CFDGFFIMC.FFIEN** is 1'b0, the read value of this bit is 0.

Users can only write to this bit when RS-CAN-FD module is in GL_RESET mode and **CFDGFFIMC.FFIEN** is 1'b1.

4.3.94.2 CFDVMCFGn.TXQ1VMN

VM number for TXQ1

This bit specifies the virtual machine number which uses TXQ1.

When **CFDGFFIMC.FFIEN** is 1'b0, the read value of this bit is 0.

Users can only write to this bit when RS-CAN-FD module is in GL_RESET mode and **CFDGFFIMC.FFIEN** is 1'b1.

4.3.94.3 CFDVMCFGn.TXQ2VMN

VM number for TXQ2

This bit specifies the virtual machine number which uses TXQ2.

When **CFDGFFIMC.FFIEN** is 1'b0, the read value of this bit is 0.

Users can only write to this bit when RS-CAN-FD module is in GL_RESET mode and **CFDGFFIMC.FFIEN** is 1'b1.

4.3.94.4 CFDVMCFGn.TXQ3VMN

VM number for TXQ3

This bit specifies the virtual machine number which uses TXQ3.

When **CFDGFFIMC.FFIEN** is 1'b0, the read value of this bit is 0.

Users can only write to this bit when RS-CAN-FD module is in GL_RESET mode and **CFDGFFIMC.FFIEN** is 1'b1.

The register controlled by setup of **CFDVMCFGn.TXQ0VMN**, **CFDVMCFGn.TXQ1VMN**, **CFDVMCFGn.TXQ2VMN**, **CFDVMCFGn.TXQ3VMN** becomes the following.

CFDVMCFGn.TXQ0VMN	CFDVMCFGn.TXQ1VMN	CFDVMCFGn.TXQ2VMN	CFDVMCFGn.TXQ3VMN
CFDTXQCC0n	CFDTXQCC1n	CFDTXQCC2n	CFDTXQCC3n
CFDTXQSTS0n	CFDTXQSTS1n	CFDTXQSTS2n	CFDTXQSTS3n
CFDTXQPCTR0n	CFDTXQPCTR1n	CFDTXQPCTR2n	CFDTXQPCTR3n
CFDTMIDn*64	CFDTMIDn*64+31	CFDTMIDn*64+32	CFDTMIDn*64+63
CFDTMPTR n*64	CFDTMPTR n*64+31	CFDTMPTR n*64+32	CFDTMPTR n*64+63
CFDTMFCTRn*64	CFDTMFCTRn*64+31	CFDTMFCTRn*64+32	CFDTMFCTRn*64+63
CFDTMDF00n*64	CFDTMDF00n*64+31	CFDTMDF00n*64+32	CFDTMDF00n*64+63
CFDTMDF01n*64	CFDTMDF01n*64+31	CFDTMDF01n*64+32	CFDTMDF01n*64+63
CFDTMDF02n*64	CFDTMDF02n*64+31	CFDTMDF02n*64+32	CFDTMDF02n*64+63
CFDTMDF03n*64	CFDTMDF03n*64+31	CFDTMDF03n*64+32	CFDTMDF03n*64+63
CFDTMDF04n*64	CFDTMDF04n*64+31	CFDTMDF04n*64+32	CFDTMDF04n*64+63
CFDTMDF05n*64	CFDTMDF05n*64+31	CFDTMDF05n*64+32	CFDTMDF05n*64+63
CFDTMDF06n*64	CFDTMDF06n*64+31	CFDTMDF06n*64+32	CFDTMDF06n*64+63
CFDTMDF07n*64	CFDTMDF07n*64+31	CFDTMDF07n*64+32	CFDTMDF07n*64+63
CFDTMDF08n*64	CFDTMDF08n*64+31	CFDTMDF08n*64+32	CFDTMDF08n*64+63
CFDTMDF09n*64	CFDTMDF09n*64+31	CFDTMDF09n*64+32	CFDTMDF09n*64+63
CFDTMDF10n*64	CFDTMDF10n*64+31	CFDTMDF10n*64+32	CFDTMDF10n*64+63
CFDTMDF11n*64	CFDTMDF11n*64+31	CFDTMDF11n*64+32	CFDTMDF11n*64+63
CFDTMDF12n*64	CFDTMDF12n*64+31	CFDTMDF12n*64+32	CFDTMDF12n*64+63
CFDTMDF13n*64	CFDTMDF13n*64+31	CFDTMDF13n*64+32	CFDTMDF13n*64+63
CFDTMDF14n*64	CFDTMDF14n*64+31	CFDTMDF14n*64+32	CFDTMDF14n*64+63
CFDTMDF15n*64	CFDTMDF15n*64+31	CFDTMDF15n*64+32	CFDTMDF15n*64+63

4.3.94.5 CFDVMCFGn.CF0VMN

VM number for COMFIFO0

This bit specifies the virtual machine number which uses COMFIFO0.

When **CFDGFFIMC.FFIEN** is 1'b0, the read value of this bit is 0.

Users can only write to this bit when RS-CAN-FD module is in GL_RESET mode and **CFDGFFIMC.FFIEN** is 1'b1.

4.3.94.6 CFDVMCFGn.CF1VMN

VM number for COMFIFO1

This bit specifies the virtual machine number which uses COMFIFO1.

When **CFDGFFIMC.FFIEN** is 1'b0, the read value of this bit is 0.

Users can only write to this bit when RS-CAN-FD module is in GL_RESET mode and **CFDGFFIMC.FFIEN** is 1'b1.

4.3.94.7 CFDVMCFGn.CF2VMN

VM number for COMFIFO2

This bit specifies the virtual machine number which uses COMFIFO2.

When **CFDGFFIMC.FFIEN** is 1'b0, the read value of this bit is 0.

Users can only write to this bit when RS-CAN-FD module is in GL_RESET mode and **CFDGFFIMC.FFIEN** is 1'b1.

The register controlled by setup of ***CFDVMCFGn.CF0VMN , CFDVMCFGn.CF1VMN , CFDVMCFGn.CF2VMN*** becomes the following.

<i>CFDVMCFGn.CF0VMN</i>	<i>CFDVMCFGn.CF1VMN</i>	<i>CFDVMCFGn.CF2VMN</i>
<i>CFDCFCCn*3</i>	<i>CFDCFCCn*3+1</i>	<i>CFDCFCCn*3+2</i>
<i>CFDCFCCEn*3</i>	<i>CFDCFCCEn*3+1</i>	<i>CFDCFCCEn*3+2</i>
<i>CFDCFSTS_n*3</i>	<i>CFDCFSTS_n*3+1</i>	<i>CFDCFSTS_n*3+2</i>
<i>CFDCFPCTR_n*3</i>	<i>CFDCFPCTR_n*3+1</i>	<i>CFDCFPCTR_n*3+2</i>
<i>CFDCFID_n*3</i>	<i>CFDCFID_n*3+1</i>	<i>CFDCFID_n*3+2</i>
<i>CFDCFPTR_n*3</i>	<i>CFDCFPTR_n*3+1</i>	<i>CFDCFPTR_n*3+2</i>
<i>CFDCFFDCSTS_n*3</i>	<i>CFDCFFDCSTS_n*3+1</i>	<i>CFDCFFDCSTS_n*3+2</i>
<i>CFDCFDF00n*3</i>	<i>CFDCFDF00n*3+1</i>	<i>CFDCFDF00n*3+2</i>
<i>CFDCFDF01n*3</i>	<i>CFDCFDF01n*3+1</i>	<i>CFDCFDF01n*3+2</i>
<i>CFDCFDF02n*3</i>	<i>CFDCFDF02n*3+1</i>	<i>CFDCFDF02n*3+2</i>
<i>CFDCFDF03n*3</i>	<i>CFDCFDF03n*3+1</i>	<i>CFDCFDF03n*3+2</i>
<i>CFDCFDF04n*3</i>	<i>CFDCFDF04n*3+1</i>	<i>CFDCFDF04n*3+2</i>
<i>CFDCFDF05n*3</i>	<i>CFDCFDF05n*3+1</i>	<i>CFDCFDF05n*3+2</i>
<i>CFDCFDF06n*3</i>	<i>CFDCFDF06n*3+1</i>	<i>CFDCFDF06n*3+2</i>
<i>CFDCFDF07n*3</i>	<i>CFDCFDF07n*3+1</i>	<i>CFDCFDF07n*3+2</i>
<i>CFDCFDF08n*3</i>	<i>CFDCFDF08n*3+1</i>	<i>CFDCFDF08n*3+2</i>
<i>CFDCFDF09n*3</i>	<i>CFDCFDF09n*3+1</i>	<i>CFDCFDF09n*3+2</i>
<i>CFDCFDF10n*3</i>	<i>CFDCFDF10n*3+1</i>	<i>CFDCFDF10n*3+2</i>
<i>CFDCFDF11n*3</i>	<i>CFDCFDF11n*3+1</i>	<i>CFDCFDF11n*3+2</i>
<i>CFDCFDF12n*3</i>	<i>CFDCFDF12n*3+1</i>	<i>CFDCFDF12n*3+2</i>
<i>CFDCFDF13n*3</i>	<i>CFDCFDF13n*3+1</i>	<i>CFDCFDF13n*3+2</i>
<i>CFDCFDF14n*3</i>	<i>CFDCFDF14n*3+1</i>	<i>CFDCFDF14n*3+2</i>
<i>CFDCFDF15n*3</i>	<i>CFDCFDF15n*3+1</i>	<i>CFDCFDF15n*3+2</i>

4.3.95 CFDVMRFCFG

Global Virtual Machine RX FIFO configuration Register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
			VMN7[3:0]			VMN6[3:0]		
Value after reset	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
		VMN5[3:0]			VMN4[3:0]			
Value after reset	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
		VMN3[3:0]			VMN2[3:0]			
Value after reset	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
		VMN1[3:0]			VMN0[3:0]			
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:28]	VMN7	VM number for RXFIFO7	Virtual machine which uses RXFIFO7 is specified.	R/W
b[27:24]	VMN6	VM number for RXFIFO6	Virtual machine which uses RXFIFO6 is specified.	R/W
b[23:20]	VMN5	VM number for RXFIFO5	Virtual machine which uses RXFIFO5 is specified.	R/W
b[19:16]	VMN4	VM number for RXFIFO4	Virtual machine which uses RXFIFO4 is specified.	R/W
b[15:12]	VMN3	VM number for RXFIFO3	Virtual machine which uses RXFIFO3 is specified.	R/W
b[11:8]	VMN2	VM number for RXFIFO2	Virtual machine which uses RXFIFO2 is specified.	R/W
b[7:4]	VMN1	VM number for RXFIFO1	Virtual machine which uses RXFIFO1 is specified.	R/W
b[3:0]	VMN0	VM number for RXFIFO0	Virtual machine which uses RXFIFO0 is specified.	R/W

Assignment of a setting value and virtual machine is shown below.

bit value	VM assignment
4'b0xxx	invalid
4'b1000	VM0
4'b1001	VM1
4'b1010	VM2
4'b1011	VM3
4'b1100	VM4
4'b1101	VM5
4'b1110	VM6
4'b1111	VM7

4.3.95.1 CFDVMRFCFG.VMNx (x = FIFO index = [0...a-1])(a = No. of RX FIFOs = 8)

VM number for RXFIFO

This bit specifies the virtual machine number which uses RXFIFO.

When **CFDGFFIMC.FFIEN** is 1'b0, the read value of this bit is 0.

Users can only write to this bit when RS-CAN-FD module is in GL_RESET mode and **CFDGFFIMC.FFIEN** is 1'b1.

The register controlled by setup of CFDVMRFCFG.VMNx becomes the following.

CFDRFCCx
CFDRFSTSx
CFDRFPCTR_x
CFDRFID_x
CFDRFPTRx
CFDRFFDSTSx
CFDRFDF00x
CFDRFDF01x
CFDRFDF02x
CFDRFDF03x
CFDRFDF04x
CFDRFDF05x
CFDRFDF06x
CFDRFDF07x
CFDRFDF08x
CFDRFDF09x
CFDRFDF10x
CFDRFDF11x
CFDRFDF12x
CFDRFDF13x
CFDRFDF14x
CFDRFDF15x

4.3.96 CFDVMISTSn

Virtual Machine Interrupt Status Register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
Value after reset	-	-	-	-	-	-	-	-
	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset	-	-	THLELT	TXQMOW	TXQMLT	CFMOW	CFMLT	RFMLT
	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
Value after reset	-	TXQFIF	TXQOFRXIF	CFFIF	CFOFRXINT	CFRXINT	RFFIF	RFIF
	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset	-	-	-	THLIF	TXQOFTXIF	TXQTXIF	CFOFTXINT	CFTXINT
	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:22]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[21]	THLELT	TX History List Entry Lost	1'b0: Corresponding TX History List Entry Lost Error not detected 1'b1: Corresponding TX History List Entry Lost Error detected	R
b[20]	TXQMOW	TXQ Message Overwrite	1'b0: Corresponding TXQ Message Overwrite Error not detected 1'b1: Corresponding TXQ Message Overwrite Error detected	R
b[19]	TXQMLT	TXQ Message Lost	1'b0: Corresponding TXQ Message Lost Error not detected 1'b1: Corresponding TXQ Message Lost Error detected	R
b[18]	CFMOW	COMFIFO Message Overwrite	1'b0: Corresponding Common FIFO Message Overwrite Error not detected 1'b1: Corresponding Common FIFO Message Overwrite Error detected	R
b[17]	CFMLT	COMFIFO Message Lost	1'b0: Corresponding Common FIFO Message Lost Error not detected 1'b1: Corresponding Common FIFO Message Lost Error detected	R
b[16]	RFMLT	RXFIFO Message Lost	1'b0: Corresponding RXFIFO Message Lost Error not detected 1'b1: Corresponding RXFIFO Message Lost Error detected	R
b[15]	-	Reserved	This bit is read as 0b. The write value should be always 0b.	R/W
b[14]	TXQFIF	TXQ Full Interrupt Flag	1'b0: Corresponding TXQ Full interrupt flag is not set 1'b1: Corresponding TXQ Full interrupt flag is set	R
b[13]	TXQOFRXIF	TXQ One Frame RX Interrupt Flag	1'b0: Corresponding TXQ One Frame RX interrupt flag is not set 1'b1: Corresponding TXQ One Frame RX interrupt flag is set	R
b[12]	CFFIF	COMFIFO Full Interrupt Flag	1'b0: Corresponding Common FIFO Full interrupt flag is not set 1'b1: Corresponding Common FIFO Full interrupt flag is set	R
b[11]	CFOFRXINT	COMFIFO One Frame RX Interrupt Flag	1'b0: Corresponding Common FIFO One Frame RX interrupt flag is not set 1'b1: Corresponding Common FIFO One Frame RX interrupt flag is set	R
b[10]	CFRXINT	COMFIFO RX Interrupt Flag	1'b0: Corresponding Common FIFO RX interrupt flag is not set 1'b1: Corresponding Common FIFO RX interrupt flag is set	R
b[9]	RFFIF	RXFIFO Full Interrupt Flag	1'b0: Corresponding RXFIFO Full interrupt flag is not set 1'b1: Corresponding RXFIFO Full interrupt flag is set	R
b[8]	RFIF	RXFIFO Interrupt Flag	1'b0: Corresponding RXFIFO interrupt flag is not set 1'b1: Corresponding RXFIFO interrupt flag is set	R
b[7:5]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[4]	THLIF	TX History List entry Interrupt Flag	1'b0: Corresponding THL Entry interrupt flag is not set 1'b1: Corresponding THL Entry interrupt flag is set	R
b[3]	TXQOFTXIF	TXQ One Frame TX Interrupt Flag	1'b0: Corresponding TXQ One Frame TX interrupt flag is not set 1'b1: Corresponding TXQ One Frame TX interrupt flag is set	R
b[2]	TXQTXIF	TXQ TX Interrupt Flag	1'b0: Corresponding TXQ TX interrupt flag is not set 1'b1: Corresponding TXQ TX interrupt flag is set	R

b[1]	CFOFTXINT	COMFIFO One Frame TX Interrupt Flag	1'b0: Corresponding Common FIFO One Frame TX interrupt flag is not set 1'b1: Corresponding Common FIFO One Frame TX interrupt flag is set	R
b[0]	CFTXINT	COMFIFO TX Interrupt Flag	1'b0: Corresponding Common FIFO TX interrupt flag is not set 1'b1: Corresponding Common FIFO TX interrupt flag is set	R

When **CFDGFFIMC.FFIEN** is 1'b1, this register is valid.

4.3.96.1 CFDVMISTSn.CFTXINT (n = Virtual Machine index = [0...n-1])(n = No. of Channels)

COMFIFO TX Interrupt Flag

This bit is set automatically when the related Common TX/GW FIFO Interrupt Flag is set when the Interrupt is enabled.

This bit is cleared automatically when related Common TX/GW FIFO Interrupt Flag is cleared or the Interrupt enable is disabled.

This bit will be cleared automatically in GL_RESET mode or related channel enter CH_RESET mode if the FIFO is configured in TX or GW mode.

When **CFDGFFIMC.FFIEN** is 1'b0, the read value of this bit is 0.

4.3.96.2 CFDVMISTSn.CFOFTXINT (n = Virtual Machine index = [0...n-1])(n = No. of Channels)

COMFIFO One Frame TX Interrupt Flag

This bit is set automatically when the related Common FIFO One Frame Transmission Interrupt Flag is set when the Interrupt is enabled.

This bit is cleared automatically when related Common FIFO One Frame Transmission Interrupt Flag is cleared or the Interrupt enable is disabled.

This bit will be cleared automatically in GL_RESET mode or related channel enter CH_RESET mode if the FIFO is configured in TX or GW mode.

When **CFDGFFIMC.FFIEN** is 1'b0, the read value of this bit is 0.

4.3.96.3 CFDVMISTSn.TXQTXIF (n = Virtual Machine index = [0...n-1])(n = No. of Channels)

TXQ TX Interrupt Flag Channel

This bit is set automatically when the related TX Queue Interrupt flag is set when the Interrupt is enabled.

This bit is cleared automatically when related TX Queue Interrupt flag is cleared or the Interrupt enable is disabled.

The bit is cleared if the related RS-CAN-FD channel enters CH_RESET mode.

When **CFDGFFIMC.FFIEN** is 1'b0, the read value of this bit is 0.

4.3.96.4 CFDVMISTSn.TXQOFTXIF (n = Virtual Machine index = [0...n-1])(n = No. of Channels)

TXQ One Frame TX Interrupt Flag

This bit is set automatically when the related TX Queue One Frame TX Interrupt flag is set when the Interrupt is enabled.

This bit is cleared automatically when related TX Queue One Frame TX Interrupt flag is cleared or the Interrupt enable is disabled.

The bit is cleared if the related RS-CAN-FD channel enters CH_RESET mode.

When **CFDGFFIMC.FFIEN** is 1'b0, the read value of this bit is 0.

4.3.96.5 CFDVMISTSn.THLIF (n = Virtual Machine index = [0...n-1])(n = No. of Channels)

TX History List entry Interrupt Flag

This bit is set automatically when the related TX History List Interrupt Flag is set when the Interrupt is enabled.

This bit is cleared automatically when related TX History List Interrupt Flag is cleared or the Interrupt enable is disabled.

This bit will be cleared automatically in GL_RESET mode.

When **CFDGFFIMC.FFIEN** is 1'b0, the read value of this bit is 0.

4.3.96.6 CFDVMISTSn.RFIF (n = Virtual Machine index = [0...n-1])(n = No. of Channels)

RXFIFO Interrupt Flag

This bit is set automatically when the RX FIFO Interrupt flag is set when the Interrupt is enabled.

This bit is cleared automatically when the RX FIFO Interrupt flag is cleared or the Interrupt enable is disabled.

This bit will be cleared automatically in GL_RESET mode.

When **CFDGFFIMC.FFIEN** is 1'b0, the read value of this bit is 0.

4.3.96.7 CFDVMISTSn.RFFIF (n = Virtual Machine index = [0...n-1])(n = No. of Channels)

RXFIFO Full Interrupt Flag

This bit is set automatically when the RX FIFO Full Interrupt flag is set when the Interrupt is enabled.

This bit is cleared automatically when the RX FIFO Full Interrupt flag is cleared or the Interrupt enable is disabled.

This bit will be cleared automatically in GL_RESET mode.

When **CFDGFFIMC.FFIEN** is 1'b0, the read value of this bit is 0.

4.3.96.8 CFDVMISTSn.CFRXINT (n = Virtual Machine index = [0...n-1])(n = No. of Channels)

COMFIFO RX Interrupt Flag

This bit is set automatically when the related COM RX FIFO Interrupt flag is set when the Interrupt is enabled.

This bit is cleared automatically when the related COM RX FIFO Interrupt flag is cleared or the Interrupt enable is disabled.

This bit will be cleared automatically in GL_RESET mode or related channel enter CH_RESET mode if the FIFO is configured in GW mode.

When **CFDGFFIMC.FFIEN** is 1'b0, the read value of this bit is 0.

4.3.96.9 CFDVMISTSn.CFOFRXINT (n = Virtual Machine index = [0...n-1])(n = No. of Channels)

COMFIFO One Frame RX Interrupt Flag

This bit is set automatically when the related COM RX FIFO One Frame Interrupt flag is set when the Interrupt is enabled.

This bit is cleared automatically when the related COM RX FIFO One frame Interrupt flag is cleared or the Interrupt enable is disabled.

This bit will be cleared automatically in GL_RESET mode or related channel enter CH_RESET mode if the FIFO is configured in GW mode.

When **CFDGFFIMC.FFIEN** is 1'b0, the read value of this bit is 0.

4.3.96.10 CFDVMISTSn.CFFIF (n = Virtual Machine index = [0...n-1])(n = No. of Channels)

COMFIFO Full Interrupt Flag

This bit is set automatically when the related COM RX FIFO Full Interrupt flag is set when the Interrupt is enabled.

This bit is cleared automatically when the related COM RX FIFO Full Interrupt flag is cleared or the Interrupt enable is disabled.

This bit will be cleared automatically in GL_RESET mode or related channel enter CH_RESET mode if the FIFO is configured in GW mode.

When **CFDGFFIMC.FFIEN** is 1'b0, the read value of this bit is 0.

4.3.96.11 CFDVMISTSn.TXQOFRXIF (n = Virtual Machine index = [0...n-1])(n = No. of Channels)

TXQ One Frame RX Interrupt Flag

This bit is set automatically when the related TXQ One Frame RX Interrupt flag is set when the Interrupt is enabled.

This bit is cleared automatically when the related TXQ One frame RX Interrupt flag is cleared or the Interrupt enable is disabled.

The bit is cleared if the related RS-CAN-FD channel enters CH_RESET mode.

When **CFDGFFIMC.FFIEN** is 1'b0, the read value of this bit is 0.

4.3.96.12 CFDVMISTSn.TXQFIF (n = Virtual Machine index = [0...n-1])(n = No. of Channels)

TXQ Full Interrupt Flag

This bit is set automatically when the related TXQ Full Interrupt flag is set when the Interrupt is enabled.
This bit is cleared automatically when the related TXQ Full Interrupt flag is cleared or the Interrupt enable is disabled.

The bit is cleared if the related RS-CAN-FD channel enters CH_RESET mode.

When **CFDGFFIMC.FFIEN** is 1'b0, the read value of this bit is 0.

4.3.96.13 CFDVMISTSn.RFMLT (n = Virtual Machine index = [0...n-1])(n = No. of Channels)

RXFIFO Message Lost

This bit is set automatically when the related RX FIFO Message Lost Error is detected.

This bit is cleared automatically when related RX FIFO Message Lost flags are cleared or the Interrupt enable is disabled.

This bit will be cleared automatically in GL_RESET mode.

When **CFDGFFIMC.FFIEN** is 1'b0, the read value of this bit is 0.

When **CFDGFFIMC.FFIEN** is 1'b1 and **CFDGVMEIS.FMLT** is 1'b1, this bit is valid.

4.3.96.14 CFDVMISTSn.CFMLT (n = Virtual Machine index = [0...n-1])(n = No. of Channels)

COMFIFO Message Lost

This bit is set automatically when the related COM RX/GW FIFO Message Lost Error is detected.

This bit is cleared automatically when related COM RX/GW FIFO Message Lost flags are cleared or the Interrupt enable is disabled.

This bit will be cleared automatically in GL_RESET mode or related channel enter CH_RESET mode if the FIFO is configured in GW mode.

When **CFDGFFIMC.FFIEN** is 1'b0, the read value of this bit is 0.

When **CFDGFFIMC.FFIEN** is 1'b1 and **CFDGVMEIS.FMLT** is 1'b1, this bit is valid.

4.3.96.15 CFDVMISTSn.CFMOW (n = Virtual Machine index = [0...n-1])(n = No. of Channels)

COMFIFO Message Overwrite

This bit is set automatically when the related COM GW FIFO Message Overwrite Error is detected.

This bit is cleared automatically when related COM GW FIFO Message Overwrite flags are cleared or the Interrupt enable is disabled.

This bit will be cleared automatically in GL_RESET mode or related channel enter CH_RESET mode if the FIFO is configured in GW mode.

When **CFDGFFIMC.FFIEN** is 1'b0, the read value of this bit is 0.

When **CFDGFFIMC.FFIEN** is 1'b1 and **CFDGVMEIS.CFMOW** is 1'b1, this bit is valid.

4.3.96.16 CFDVMISTSn.TXQMLT (n = Virtual Machine index = [0...n-1])(n = No. of Channels)

TXQ Message Lost

This bit is set automatically when the related TXQ Message Lost Error is detected.

This bit is cleared automatically when related TXQ Message Lost flags are cleared or the Interrupt enable is disabled.

The bit is cleared if the related RS-CAN-FD channel enters CH_RESET mode.

When **CFDGFFIMC.FFIEN** is 1'b0, the read value of this bit is 0.

When **CFDGFFIMC.FFIEN** is 1'b1 and **CFDGVMEIS.TXQMLT** is 1'b1, this bit is valid.

4.3.96.17 CFDVMISTSn.TXQMOW (n = Virtual Machine index = [0...n-1])(n = No. of Channels)

TXQ Message Overwrite

This bit is set automatically when the related TXQ Message Overwrite Error is detected.

This bit is cleared automatically when related TXQ Message Overwrite flags are cleared or the Interrupt enable is disabled.

The bit is cleared if the related RS-CAN-FD channel enters CH_RESET mode.

When **CFDGFFIMC.FFIEN** is 1'b0, the read value of this bit is 0.

When **CFDGFFIMC.FFIEN** is 1'b1 and **CFDGVMEIS.TXQMOW** is 1'b1, this bit is valid.

4.3.96.18 CFDVMIST_n.THLELT (n = Virtual Machine index = [0...n-1])(n = No. of Channels)

TX History List Entry Lost

This bit is set automatically when the related TX History List Entry Lost Interrupt Flag is set when the Interrupt is enabled.

This bit is cleared automatically when related TX History List Entry Lost Interrupt Flag is cleared or the Interrupt enable is disabled.

This bit will be cleared automatically in GL_RESET mode.

When **CFDGFFIMC.FFIEN** is 1'b0, the read value of this bit is 0.

When **CFDGFFIMC.FFIEN** is 1'b1 and **CFDGVMEIS.THLELT** is 1'b1, this bit is valid.

4.3.97 CFDGPFLECTR

Global Pretended Network Filter List Entry control Register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
	-	-	-	-	-	-	-	-
Value after reset	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
	-	-	-	-	-	-	-	-
Value after reset	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
	-	-	-	-	-	-	-	PFLDAE
Value after reset	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
	-	-				PFLPN[5:0]		
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:9]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[8]	PFLDAE	Pretended Network Filter List Data Access Enable	1'b0: Pretended Network Filter List Data access disabled 1'b1: Pretended Network Filter List Data access enabled	R/W
b[7:6]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[5:0]	PFLPN	Pretended Network Filter List Page Number	Pretended Network Filter List Page Number	R/W

4.3.97.1 CFDGPFLECTR.PFLPN

Pretended Network Filter List Page Number

These bits select the Page Number to access the desired RAM area of the Pretended Network Filter List.

One Pretended Network Filter List page consists of 4 Pretended Network Filter List entries.

Read/Write accesses to the Pretended Network Filter List can only be performed via a fixed window.

Users cannot write to this bit when RS-CAN-FD module is in GL_SLEEP mode.

Users should enter, only the values between 0 and 39 (7'h27) inclusive.

4.3.97.2 CFDGPFLECTR.PFLDAE

Pretended Network Filter List Data Access Enable

This bit prevents Pretended Network Filter List write access if cleared after configuration of the Pretended Network Filter List.

Users can read data from Pretended Network Filter List independent of the status of this bit.

Users cannot write to this bit when RS-CAN-FD module is in GL_SLEEP mode.

This bit should be set to enable write access to Pretended Network Filter List.

4.3.98 CFDGPFLCFGu

Global Pretended Network Filter List configuration Register u

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
	-	-			RNC((3-3)+(u*4))[5:0]			
Value after reset	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
	-	-			RNC((3-2)+(u*4))[5:0]			
Value after reset	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
	-	-			RNC((3-1)+(u*4))[5:0]			
Value after reset	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
	-	-			RNC((3-0)+(u*4))[5:0]			
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:30]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[29:24]	RNC((3-3)+(u*4))	Rule Number for Channel ((3 - 3) + (u * 4))	Number of rules dedicated to channel ((3-Address[1:0])+(u*4))	R/W
b[23:22]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[21:16]	RNC((3-2)+(u*4))	Rule Number for Channel ((3 - 2) + (u * 4))	Number of rules dedicated to channel ((3-Address[1:0])+(u*4))	R/W
b[15:14]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[13:8]	RNC((3-1)+(u*4))	Rule Number for Channel ((3 - 1) + (u * 4))	Number of rules dedicated to channel ((3-Address[1:0])+(u*4))	R/W
b[7:6]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[5:0]	RNC((3-0)+(u*4))	Rule Number for Channel ((3 - 0) + (u * 4))	Number of rules dedicated to channel ((3-Address[1:0])+(u*4))	R/W

(no_of_channels = no_of_RNs = 8)

(no_of_RNs_per_CFDGPFLCFG = 4)

(no_of_CFDGPFLCFGs = ceil(no_of_RNs / no_of_INTFs_per_CFDGPFLCFG) = ceil(8 / 4) = 2)

u = [0...no_of_CFDGPFLCFGs-1]

n = [0...no_of_channels-1]

(u) can be calculated from the desired Channel (n) Rule Number using the formula u = floor(n / 4)

Byte position can be calculated using the formula ((3 - n) + (u * 4))

e.g. for Channel 7 Rule Number we have:

n = 7, u = floor(7 / 4) = 1 and Byte position = ((3 - 7) + (1 * 4)) = 0

Therefore, Users should read the Byte 0 (First Byte) of **CFDGPFLCFG1**.

This register is used to define the number of Pretended Network Filter List Entries (called "Rules") applicable for channels 0 to 7 in the Pretended Network Filter List.

The total number of available entries in the Pretended Network Filter List is 20 * (n+1) (e.g. 160 for 8 CAN channels).

However, the filters can be allocated flexibly to the different channels depending on requirements as long as both of the following conditions are satisfied:

- the maximum number of Pretended Network Filter per channel is 32 and
- the total number of rules defined for all channels is not exceeding the number of available entries in the Pretended Network Filter List.

4.3.98.1 CFDGPFLCFG_u.RNC((3-n)+(u*4))

Rule Number for Channel ((3 - n) + (u * 4))

These bits define the number of rules in the Pretended Network Filter List for channel n.

Users can only write to this bit when RS-CAN-FD module is in GL_RESET mode.

4.3.99 CFDGPFLIDs

Global Pretended Network Filter List ID Registers s = [1...4]h

Address: Refer to section 4.1

	b31 GPFLIDE	b30 GPFLRTR	b29 GPFLLB	b28	b27	b26 GPFLID[28:24]	b25	b24
Value after reset	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
Value after reset	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31]	GPFLIDE	Global Pretended Network Filter List IDE Field	1'b0: Standard Identifier of Rule entry ID is valid for acceptance filtering 1'b1: Extended Identifier of Rule entry ID is valid for acceptance filtering	R/W
b[30]	GPFLRTR	Global Pretended Network Filter List Entry RTR Field	1'b0: Data Frame 1'b1: Remote Frame	R/W
b[29]	GPFLLB	Global Pretended Network Filter List Entry Loopback Configuration	1'b0: Global Pretended Network Filter List entry ID for acceptance filtering has attribute 'RX' 1'b1: Global Pretended Network Filter List entry ID for acceptance filtering has attribute 'TX'	R/W
b[28:0]	GPFLID	Global Pretended Network Filter List ID Field	ID part of the Global Pretended Network Filter List entry	R/W

(no_of_CFDGPFLIDs = 4)

(s = [1...no_of_CFDGPFLIDs])

These registers are used to configure the ID field of the Rule Entries in the Global Pretended Network Filter List.

4.3.99.1 CFDGPFLIDs.GPFLID

Global Pretended Network Filter List Entry ID Field

These bits represent the CAN Identifier (ID) field of each of the Global Pretended Network Filter List entry. Pretended Network filter process compares this field against the ID of a received CAN message.

For alignment of these bits in standard and extended frame format, see Section 4.4

Users cannot write to these bits when **CFDGPFLCTR.PFLDAE** bit is 1'b0.

Users should only write to these bits when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

4.3.99.2 CFDGPFLIDs.GPFLLB

Global Pretended Network Filter List Entry Loopback Configuration

This bit selects if the Global Pretended Network Filter List entry gets the attribute 'RX' or 'TX'. This attribute decides about the validity of the entry in the mirror mode case, loopback test mode case and during standard (non-loopback) reception. Please see Table 18.5 for detailed description of the validity of the Global Pretended Network Filter List entry depending on transmitter/receiver case, type of loopback mode and RX/TX attribute.

Users cannot write to these bits when **CFDGPFLCTR.PFLDAE** bit is 1'b0.

Users should only write to these bits when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

Users should only write to this bit 1'b1.

4.3.99.3 CFDGPFLIDs.GPFLRTR

Global Pretended Network Filter List Entry RTR Field

This bit allows the configuration of the specified frame format (Data Frame or Remote Frame) for each Global Pretended Network Filter List entry. For each Rule entry in a CAN channel the Pretended Network filter process compares this bit against the RTR bit of the received CAN message.

Users cannot write to these bits when **CFDGPFLCTR.PFLDAE** bit is 1'b0.

Users should only write to these bits when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

4.3.99.4 CFDGPFLIDs.GPFLIDE

Global Pretended Network Filter List Entry IDE Field

This bit allows the configuration of the ID format (Standard ID or Extended ID) for each of the Global Pretended Network Filter List entry. For each Rule entry of the related CAN channel the Pretended Network filter process compares this bit against the IDE bit of the received CAN message.

Users cannot write to these bits when **CFDGPFLCTR.PFLDAE** bit is 1'b0.

Users should only write to these bits when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

4.3.100 CFDGPFLMs

Global Pretended Network Filter List MASK Registers s = [1...4]h

Address: Refer to section

4.1

	b31	b30	b29	b28	b27	b26	b25	b24
	GPFLIDEM	GPFLRTRM	GPFLIFL1			GPFLIDM[28:24]		
Value after reset	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
	GPFLIDM[23:16]							
Value after reset	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
	GPFLIDM[15:8]							
Value after reset	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
	GPFLID[7:0]							
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31]	GPFLIDEM	Global Pretended Network Filter List IDE Mask	1'b0: IDE bit is not considered for ID matching 1'b1: IDE bit is considered for ID matching	R/W
b[30]	GPFLRTRM	Global Pretended Network Filter List Entry RTR Mask	1'b0: RTR bit is not considered for ID matching 1'b1: RTR bit is considered for ID matching	R/W
b[29]	GPFLIFL1	Global Pretended Network Filter List Information Label 1	Global Pretended Network Filter List Information Label bit1	R/W
b[28:0]	GPFLIDM	Global Pretended Network Filter List ID Mask Field	Global Pretended Network Filter List Mask field bits for ID field bits	R/W

The Global Rule Mask entry registers are used to configure the Mask field of each Rule Entries in the Global Pretended Network Filter List.

4.3.100.1 CFDGPFLMs.GPFLIDM

Global Pretended Network Filter List ID Mask Field

These bits are the filter mask bits for the related bits in the CAN Identifier field of each Global Pretended Network Filter List entry.

1'b0	Corresponding STD-ID / EXT-ID bit is not considered for ID matching
1'b1	Corresponding STD-ID / EXT-ID bit is considered for ID matching

Users cannot write to these bits when **CFDGPFLECTR.PFLDAE** bit is 1'b0.

Users should only write to these bits when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

4.3.100.2 CFDGPFLMs.GPFLIFL1

Global Pretended Network Filter List Information Label 1

These bits allow the configuration of a 2-bit Information label that will be attached to a received message accepted by the related Global Pretended Network Filter List entry.

This bit is a MSB bit of an information label.

Users cannot write to these bits when **CFDGPFLECTR.PFLDAE** bit is 1'b0.

Users should only write to these bits when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

This bit is stored in InfomationLabelField [1] (**CFDRMFSTS.RMIFL** [1], **CFDRFFDSTS.RFIFL** [1],

CFDCFFDCSTS.CFIFL [1] of the storage location of an incoming message.

This bit is stored in **CFDTHLACC1n.TIFL** [1] when **CFDTHLCCn.THLDGE=1** is set up using GW function.

4.3.100.3 CFDGPFLMs.GPFLRTRM

Global Pretended Network Filter List Entry RTR Mask

This bit allows the configuration of the RTR mask bit for each Global Pretended Network Filter List entry.

Users cannot write to these bits when **CFDGPFLECTR.PFLDAE** bit is 1'b0.

Users should only write to these bits when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

4.3.100.4 CFDGPFLMs.GPFLIDEM

Global Pretended Network Filter List IDE Mask

This bit allows the configuration of the IDE mask bit for each Global Pretended Network Filter List entry.

When IDE mask bit is 1'b0, then the ID comparison depends upon the received IDE bit.

If received IDE bit is 1'b0, then STD-ID comparison takes place.

If received IDE bit is 1'b1, then EXT-ID comparison takes place.

Users cannot write to these bits when **CFDGPFLECTR.PFLDAE** bit is 1'b0.

Users should only write to these bits when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

4.3.101 CFDGPFLP0s

Global Pretended Network Filter List Pointer 0 Registers s = [1...4]h

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
GPFLPTR[15:8]								
Value after reset	0	0	0	0	0	0	0	0
GPFLPTR[7:0]								
Value after reset	0	0	0	0	0	0	0	0
GPFLRMDP[6:0]								
Value after reset	0	0	0	0	0	0	0	0
GPFLIFL0 GPFLSRD2 GPFLSRD1 GPFLSRD0 GPFLDLC[3:0]								
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:16]	GPFLPTR	Global Pretended Network Filter List Pointer Field	Global Pretended Network Filter List Pointer	R/W
b[15]	GPFLRMV	Global Pretended Network Filter List RX Message Buffer Valid	1'b0: Global Pretended Network Filter List Single Message Buffer Direction Pointer is invalid 1'b1: Global Pretended Network Filter List Single Message Buffer Direction Pointer is valid	R/W
b[14:8]	GPFLRMDP	Global Pretended Network Filter List RX Message Buffer Direction Pointer	RX Message Buffer number for storage of received messages	R/W
b[7]	GPFLIFL0	Global Pretended Network Filter List Information Label bit0	Global Pretended Network Filter List Information Label bit0	R/W
b[6]	GPFLSRD2	Global Pretended Network Filter List Select Routing destination 2	1'b0: Routing target is CFIFO2 1'b1: Routing target is TX Queue 2 instead of CFIFO2	R/W
b[5]	GPFLSRD1	Global Pretended Network Filter List Select Routing destination 1	1'b0: Routing target is CFIFO1 1'b1: Routing target is TX Queue 1 instead of CFIFO1	R/W
b[4]	GPFLSRD0	Global Pretended Network Filter List Select Routing destination 0	1'b0: Routing target is CFIFO0 1'b1: Routing target is TX Queue 0 instead of CFIFO0	R/W
b[3:0]	GPFLDLC	Global Pretended Network Filter List DLC Field	Minimum no. of Data Bytes in a Data Frame required for its acceptance	R/W

The Global Pretended Network Filter List Pointer 0 registers are used to configure the DLC, SW Pointer, Single Message Buffer select and Message Buffer direction pointer for each Rule Entry in the Global Pretended Network Filter List.

4.3.101.1 CFDGPFLP0s.GPFLDLC

Global Pretended Network Filter List DLC Field

These bits allow the configuration of the minimum DLC (Data Length Code) value for a message to be accepted by the related Global Pretended Network Filter List entry (automatic DLC filter function). DLC filter process is only passed if the DLC value of the message accepted by a Global Pretended Network Filter List entry is equal or higher than the DLC value configured for this related Global Pretended Network Filter List entry.

Automatic DLC filter function is disabled for the corresponding Rule Entry when this field is set to 4'h0. Following binary values can be configured:

Format	DLC[3]	DLC[2]	DLC[1]	DLC[0]	Description
CAN and CAN-FD	0	0	0	0	DLC of received message = 0 or more (DLC Filter check is disabled)
CAN and CAN-FD	0	0	0	1	DLC of received message = 1 or more
CAN and CAN-FD	0	0	1	0	DLC of received message = 2 or more
CAN and CAN-FD	0	0	1	1	DLC of received message = 3 or more
CAN and CAN-FD	0	1	0	0	DLC of received message = 4 or more
CAN and CAN-FD	0	1	0	1	DLC of received message = 5 or more
CAN and CAN-FD	0	1	1	0	DLC of received message = 6 or more
CAN and CAN-FD	0	1	1	1	DLC of received message = 7 or more
CAN	1	x	x	x	DLC of received message = 8 or more
CAN-FD	1	0	0	0	DLC of received message = 8 or more
CAN-FD	1	0	0	1	DLC of received message = 12 or more
CAN-FD	1	0	1	0	DLC of received message = 16 or more
CAN-FD	1	0	1	1	DLC of received message = 20 or more
CAN-FD	1	1	0	0	DLC of received message = 24 or more
CAN-FD	1	1	0	1	DLC of received message = 32 or more
CAN-FD	1	1	1	0	DLC of received message = 48 or more
CAN-FD	1	1	1	1	DLC of received message = 64

Users cannot write to these bits when **CFDGPFLCTR.PFLDAE** bit is 1'b0.

Users should only write to these bits when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

4.3.101.2 CFDGPFLP0s.GPFLSRD0

Global Pretended Network Filter List Select Routing destination 0

This bit changes a copy destination to CFIFO0 or TXQ0 by routing.

If this bit is set as 1, the preset value of **CFDGPFLP1r.GPFLFDP** will choose TX Queue.

If this bit is cleared to 0, the preset value of **CFDGPFLP1r.GPFLFDP** will choose CommonFIFO.

Users cannot write to these bits when **CFDGPFLCTR.PFLDAE** bit is 1'b0.

Users should only write to these bits when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

4.3.101.3 CFDGPFLP0s.GPFLSRD1

Global Pretended Network Filter List Select Routing destination 1

This bit changes a copy destination to CFIFO1 or TXQ1 by routing.

If this bit is set as 1, the preset value of **CFDGPFLP1r.GPFLFDP** will choose TX Queue.

If this bit is cleared to 0, the preset value of **CFDGPFLP1r.GPFLFDP** will choose CommonFIFO.

Users cannot write to these bits when **CFDGPFLCTR.PFLDAE** bit is 1'b0.

Users should only write to these bits when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

4.3.101.4 CFDGPFLP0s.GPFLSRD2

Global Pretended Network Filter List Select Routing destination 2

This bit changes a copy destination to CFIFO2 or TXQ2 by routing.

If this bit is set as 1, the preset value of **CFDGPFLP1r.GPFLFDP** will choose TX Queue.

If this bit is cleared to 0, the preset value of **CFDGPFLP1r.GPFLFDP** will choose CommonFIFO.
Users cannot write to these bits when **CFDGPLECTR.PFLDAE** bit is 1'b0.
Users should only write to these bits when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

4.3.101.5 CFDGPFLP0s.GPFLIFL0

Global Pretended Network Filter List Information Label 0
These bits allow the configuration of a 2-bit Information label that will be attached to a received message accepted by the related Global Pretended Network Filter List entry.
This bit is a LSB bit of an information label.
Users cannot write to these bits when **CFDGPLECTR.PFLDAE** bit is 1'b0.
Users should only write to these bits when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.
This bit is stored in Information Label Field[0] (**CFDRMFDSTS.RMIFL[0]**, **CFDRFFDSTS.RFIFL[0]**, **CFDCFFDCSTS.CFIFL[0]**) of the storage location of an incoming message.
This bit is stored in **CFDTHLACC1n.TIFL[0]** when **CFDTHLCCn.THLDGE=1** is set up using GW function.

4.3.101.6 CFDGPFLP0s.GPFLRMDP

Global Pretended Network Filter List RX Message Buffer Direction Pointer
These bits allow the configuration of a single reception Message Buffer as the destination target for a received message that is passing the acceptance check of the related Global Pretended Network Filter List entry. The value entered is the single destination Message Buffer number.
Users cannot write to these bits when **CFDGPLECTR.PFLDAE** bit is 1'b0.
Users should only write to these bits when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.
CFDRMNBN.RRXMB[7:0] is the value entered in the RX Message Buffer Number Register to configure the number of RX Message Buffers.
The value to be entered in **CFDGPFLP0s.GPFLRMDP[6:0]** bits should only be between 7'h0 and (**CFDRMNBN.RRXMB[7:0]** - 8'h1).
If **CFDRMNBN.RRXMB[7:0]** = 8'h0, then the **CFDGPFLP0s.GPFLRMV** bit should be configured as 1'b0.

4.3.101.7 CFDGPFLP0s.GPFLRMV

Global Pretended Network Filter List RX Message Buffer Valid
This bit allows the enabling/disabling of a single reception Message Buffer as the target for a received message that is passing the acceptance check of the related Global Pretended Network Filter List entry.
Users cannot write to these bits when **CFDGPLECTR.PFLDAE** bit is 1'b0.
Users should only write to these bits when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

4.3.101.8 CFDGPFLP0s.GPFLPTR

Global Pretended Network Filter List Pointer Field
These bits allow the configuration of a 16-bit pointer that will be attached to a received message accepted by the related Global Pretended Network Filter List entry.
The Pointer will be added during message storage in the Message Buffer area and can be used by the application as support function.
The pointer information could be used for example to support PDU Identifier allocation for the received message in AUTOSAR systems.
Users cannot write to these bits when **CFDGPLECTR.PFLDAE** bit is 1'b0.
Users should only write to these bits when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

4.3.102 CFDGPFLP1s

Global Pretended Network Filter List Pointer 1 Registers s = [1...4]h

Address: Refer to section
4.1

	b31	b30	b29	b28	b27	b26	b25	b24
GPFLFDP[31:24]								
Value after reset	0	0	0	0	0	0	0	0
GPFLFDP[23:16]								
Value after reset	0	0	0	0	0	0	0	0
GPFLFDP[15:8]								
Value after reset	0	0	0	0	0	0	0	0
GPFLFDP[7:0]								
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:0]	GPFLFDP	Global Pretended Network Filter List FIFO Direction Pointer (GPFLFDP[((n+1)*3+7):0])	FIFO direction pointer bits for received message storage	R/W

The Global Pretended Network Filter List Pointer 1 registers are used to configure the FIFO direction pointer fields in each Rule Entry of the Global Pretended Network Filter List.

4.3.102.1 CFDGPFLP1s.GPFLFDP

Global Pretended Network Filter List FIFO Direction Pointer (**GPFLFDP[((n+1)*3+7):0]**)

These bits allow the configuration of FIFO Buffers as the target for a received message passing the acceptance check of the related Global Pretended Network Filter List entry. Each bit of the **CFDGPFLP1s.GPFLFDP[31:0]** is configuring a dedicated FIFO:

Bit	Value (Binary)	Function
b0	0	Disable RX FIFO 0 as target for reception
	1	Enable RX FIFO 0 as target for reception
b1	0	Disable RX FIFO 1 as target for reception
	1	Enable RX FIFO 1 as target for reception
b2	0	Disable RX FIFO 2 as target for reception
	1	Enable RX FIFO 2 as target for reception
b3	0	Disable RX FIFO 3 as target for reception
	1	Enable RX FIFO 3 as target for reception
b4	0	Disable RX FIFO 4 as target for reception
	1	Enable RX FIFO 4 as target for reception
b5	0	Disable RX FIFO 5 as target for reception
	1	Enable RX FIFO 5 as target for reception
b6	0	Disable RX FIFO 6 as target for reception
	1	Enable RX FIFO 6 as target for reception
b7	0	Disable RX FIFO 7 as target for reception

	1	Enable RX FIFO 7 as target for reception
b8	0	Disable Common FIFO 0 and Channel 0 TX Queue 0 as target for reception
	1	GPFLSRD0=0: Enable Common FIFO 0 as target for reception GPFLSRD0=1: Enable Channel 0 TX Queue 0 as target for reception
b9	0	Disable Common FIFO 1 and Channel 0 TX Queue 1 as target for reception
	1	GPFLSRD1=0: Enable Common FIFO 1 as target for reception GPFLSRD1=1: Enable Channel 0 TX Queue 1 as target for reception
b10	0	Disable Common FIFO 2 and Channel 0 TX Queue 2 as target for reception
	1	GPFLSRD2=0: Enable Common FIFO 2 as target for reception GPFLSRD2=1: Enable Channel 0 TX Queue 2 as target for reception
b11	0	Disable Common FIFO 3 and Channel 1 TX Queue 0 as target for reception
	1	GPFLSRD0=0: Enable Common FIFO 3 as target for reception GPFLSRD0=1: Enable Channel 1 TX Queue 0 as target for reception
b12	0	Disable Common FIFO 4 and Channel 1 TX Queue 1 as target for reception
	1	GPFLSRD1=0: Enable Common FIFO 4 as target for reception GPFLSRD1=1: Enable Channel 1 TX Queue 1 as target for reception
b13	0	Disable Common FIFO 5 and Channel 1 TX Queue 2 as target for reception
	1	GPFLSRD2=0: Enable Common FIFO 5 as target for reception GPFLSRD2=1: Enable Channel 1 TX Queue 2 as target for reception
b14	0	Disable Common FIFO 6 and Channel 2 TX Queue 0 as target for reception
	1	GPFLSRD0=0: Enable Common FIFO 6 as target for reception GPFLSRD0=1: Enable Channel 2 TX Queue 0 as target for reception
b15	0	Disable Common FIFO 7 and Channel 2 TX Queue 1 as target for reception
	1	GPFLSRD1=0: Enable Common FIFO 7 as target for reception GPFLSRD1=1: Enable Channel 2 TX Queue 1 as target for reception
b16	0	Disable Common FIFO 8 and Channel 2 TX Queue 2 as target for reception
	1	GPFLSRD2=0: Enable Common FIFO 8 as target for reception GPFLSRD2=1: Enable Channel 2 TX Queue 2 as target for reception
b17	0	Disable Common FIFO 9 and Channel 3 TX Queue 0 as target for reception
	1	GPFLSRD0=0: Enable Common FIFO 9 as target for reception GPFLSRD0=1: Enable Channel 3 TX Queue 0 as target for reception
b18	0	Disable Common FIFO 10 and Channel 3 TX Queue 1 as target for reception
	1	GPFLSRD1=0: Enable Common FIFO 10 as target for reception GPFLSRD1=1: Enable Channel 3 TX Queue 1 as target for reception
b19	0	Disable Common FIFO 11 and Channel 3 TX Queue 2 as target for reception
	1	GPFLSRD2=0: Enable Common FIFO 11 as target for reception GPFLSRD2=1: Enable Channel 3 TX Queue 2 as target for reception
b20	0	Disable Common FIFO 12 and Channel 4 TX Queue 0 as target for reception
	1	GPFLSRD0=0: Enable Common FIFO 12 as target for reception GPFLSRD0=1: Enable Channel 4 TX Queue 0 as target for reception
b21	0	Disable Common FIFO 13 and Channel 4 TX Queue 1 as target for reception
	1	GPFLSRD1=0: Enable Common FIFO 13 as target for reception GPFLSRD1=1: Enable Channel 4 TX Queue 1 as target for reception
b22	0	Disable Common FIFO 14 and Channel 4 TX Queue 2 as target for reception
	1	GPFLSRD2=0: Enable Common FIFO 14 as target for reception GPFLSRD2=1: Enable Channel 4 TX Queue 2 as target for reception

b23	0	Disable Common FIFO 15 and Channel 5 TX Queue 0 as target for reception
	1	GPFLSRD0=0: Enable Common FIFO 15 as target for reception GPFLSRD0=1: Enable Channel 5 TX Queue 0 as target for reception
b24	0	Disable Common FIFO 16 and Channel 5 TX Queue 1 as target for reception
	1	GPFLSRD1=0: Enable Common FIFO 16 as target for reception GPFLSRD1=1: Enable Channel 5 TX Queue 1 as target for reception
b25	0	Disable Common FIFO 17 and Channel 5 TX Queue 2 as target for reception
	1	GPFLSRD2=0: Enable Common FIFO 17 as target for reception GPFLSRD2=1: Enable Channel 5 TX Queue 2 as target for reception
b26	0	Disable Common FIFO 18 and Channel 6 TX Queue 0 as target for reception
	1	GPFLSRD0=0: Enable Common FIFO 18 as target for reception GPFLSRD0=1: Enable Channel 6 TX Queue 0 as target for reception
b27	0	Disable Common FIFO 19 and Channel 6 TX Queue 1 as target for reception
	1	GPFLSRD1=0: Enable Common FIFO 19 as target for reception GPFLSRD1=1: Enable Channel 6 TX Queue 1 as target for reception
b28	0	Disable Common FIFO 20 and Channel 6 TX Queue 2 as target for reception
	1	GPFLSRD2=0: Enable Common FIFO 20 as target for reception GPFLSRD2=1: Enable Channel 6 TX Queue 2 as target for reception
b29	0	Disable Common FIFO 21 and Channel 7 TX Queue 0 as target for reception
	1	GPFLSRD0=0: Enable Common FIFO 21 as target for reception GPFLSRD0=1: Enable Channel 7 TX Queue 0 as target for reception
b30	0	Disable Common FIFO 22 and Channel 7 TX Queue 1 as target for reception
	1	GPFLSRD1=0: Enable Common FIFO 22 as target for reception GPFLSRD1=1: Enable Channel 7 TX Queue 1 as target for reception
b31	0	Disable Common FIFO 23 and Channel 7 TX Queue 2 as target for reception
	1	GPFLSRD2=0: Enable Common FIFO 23 as target for reception GPFLSRD2=1: Enable Channel 7 TX Queue 2 as target for reception

Users cannot write to these bits when **CFDGPFLCTR.PLDAE** bit is 1'b0.

For storage in Common FIFO, target for reception can only be those Common FIFO Buffers that are configured as RX FIFO or GW FIFO.

For storage in TX Queue, when these TX Queue buffers of a target are in GW mode, it can do.

Users should only write to these bits when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

Users should only configure up to 8 destination FIFO Buffers or 7 destination FIFO Buffers plus one RX Message Buffer or 8 destination TX Queue Buffers or 7 destination TX Queue Buffers plus one RX Message Buffer.

Or a setup of a maximum of 8 destinations in all is possible at FIFO buffer and TX Queue buffer.

4.3.103 CFDGPFLPTs

Global Pretended Network Filter List Filter Type Registers s = [1...4]h

Address: Refer to section 4.1

	b31 Value after reset	GPFLANDOR	b30 0	GPFLRANG0	b29 0	GPFLOUT0	b28 0	-	b27 0	-	b26 0	-	b25 0	-	b24 0
	b23 Value after reset	-	b22 0	-	b21 0	-	b20 0	-	b19 0	b18 0	b17 0	b16 0			
	b15 Value after reset	-	b14 0	GPFLRANG1	b13 0	GPFLOUT1	b12 0	-	b11 0	-	b10 0	b9 0	b8 0		
	b7 Value after reset	-	b6 0	-	b5 0	-	b4 0	-	b3 0	b2 0	b1 0	b0 0			

Bit	Symbol	Bit name	Function	R/W
b[31]	GPFLANDOR	Global Pretended Network filter conditions of the filters 0 and 1	1'b0: Both of filters 0 and 1 are successful. 1'b1: One of the filter 0 or 1 is successful.	R/W
b[30]	GPFLRANG0	Global Pretended Network filter comparison conditions of the filter0	1'b0: payload data match filter 1'b1: upper / lower filter	R/W
b[29]	GPFLOUT0	Global Pretended Network filter conditions of upper / lower filter of the filter0	1'b0: Within the range of upper limit and lower limit 1'b1: Outside of the range of upper limit and lower limit	R/W
b[28:20]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[19:16]	GPFLOFFSET0	Global Pretended Network filter offset value of the filter0	offset value of the filter0 is specified.	R/W
b[15]	-	Reserved	This bit is read as 0b. The write value should be always 0b.	R/W
b[14]	GPFLRANG1	Global Pretended Network filter comparison conditions of the filter1	1'b0: payload data match filter 1'b1: upper / lower filter	R/W
b[13]	GPFLOUT1	Global Pretended Network filter conditions of upper / lower filter of the filter1	1'b0: Within the range of upper limit and lower limit 1'b1: Outside of the range of upper limit and lower limit	R/W
b[12:4]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[3:0]	GPFLOFFSET1	Global Pretended Network filter offset value of the filter1	offset value of the filter1 is specified.	R/W

4.3.103.1 CFDGPFLPTs.GPFLOFFSET1

Global Pretended Network filter offset value of the filter1

The offset value of the filter1 of Global Pretended Network filter is specified.

Offset can be specified per 4 bytes.

Users cannot write to these bits when **CFDGPFLCTR.PFLDAE** bit is 1'b0.

Users should only write to these bits when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

4.3.103.2 CFDGPFLPTs.GPFLOUT1

Global Pretended Network filter conditions of a upper / lower filter of the filter1

The conditions of upper / lower filter of the filter1.

Users cannot write to these bits when **CFDGPFLECTR.PFLDAE** bit is 1'b0.

Users should only write to these bits when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

4.3.103.3 CFDGPFLPTs.GPFLRANG1

Global Pretended Network filter comparison conditions of the filter1

The comparison conditions of the filter1.

Users cannot write to these bits when **CFDGPFLECTR.PFLDAE** bit is 1'b0.

Users should only write to these bits when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

4.3.103.4 CFDGPFLPTs.GPFLOFFSET0

Global Pretended Network filter offset value of the filter0

The offset value of the filter0 of Global Pretended Network filter is specified.

Offset can be specified per 4 bytes.

Users cannot write to these bits when **CFDGPFLECTR.PFLDAE** bit is 1'b0.

Users should only write to these bits when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

4.3.103.5 CFDGPFLPTs.GPFLOUT0

Global Pretended Network filter conditions of a upper / lower filter of the filter0

The conditions of upper / lower filter of the filter0.

Users cannot write to these bits when **CFDGPFLECTR.PFLDAE** bit is 1'b0.

Users should only write to these bits when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

4.3.103.6 CFDGPFLPTs.GPFLRANG0

Global Pretended Network filter comparison conditions of the filter0

The comparison conditions of the filter0.

Users cannot write to these bits when **CFDGPFLECTR.PFLDAE** bit is 1'b0.

Users should only write to these bits when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

4.3.103.7 CFDGPFLPTs.GPFLANDOR

Global Pretended Network filter conditions of the filters 0 and 1

This bit can set up the comparison conditions of two filters.

Users cannot write to these bits when **CFDGPFLECTR.PFLDAE** bit is 1'b0.

Users should only write to these bits when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

4.3.104 CFDGPFLPD0s

Global Pretended Network Filter List Payload Data 0 Registers s = [1...4]h

Address: Refer to section
4.1

	b31	b30	b29	b28	b27	b26	b25	b24
FDATA[31:24]								
Value after reset	0	0	0	0	0	0	0	0
FDATA[23:16]								
Value after reset	0	0	0	0	0	0	0	0
FDATA[15:8]								
Value after reset	0	0	0	0	0	0	0	0
FDATA[7:0]								
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:0]	FDATA	Pretended Network Filter List Filter data	Comparison data for payload filters	R/W

4.3.104.1 CFDGPFLPD0s.FDATA

Pretended Network Filter List Filter data

These bits set the payload filter data of the Pretended Network Filter List.

When **CFDGPFLPTr.GPFLRANG0** is 1'b0, these bits set up the payload match data of the filter0.

When **CFDGPFLPTr.GPFLRANG0** is 1'b1, these bits set up the upper-limit filter value of the filter0.

Users cannot write to these bits when **CFDGPFLCTR.PFLDAE** bit is 1'b0.

Users should only write to these bits when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

4.3.105 CFDGPFLPD1s

Global Pretended Network Filter List Payload Data 1 Registers s = [1...4]h

Address: Refer to section

4.1

	b31	b30	b29	b28	b27	b26	b25	b24
FDATA[31:24]								
Value after reset	0	0	0	0	0	0	0	0
FDATA[23:16]								
Value after reset	0	0	0	0	0	0	0	0
FDATA[15:8]								
Value after reset	0	0	0	0	0	0	0	0
FDATA[7:0]								
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:0]	FDATA	Pretended Network Filter List Filter data	Comparison data for payload filters	R/W

4.3.105.1 CFDGPFLPD1s.FDATA

Pretended Network Filter List Filter data

These bits set the payload filter data of the Pretended Network Filter List.

When **CFDGPFLPTr.GPFLRANG1** is 1'b0, these bits set up the payload match data of the filter1.

When **CFDGPFLPTr.GPFLRANG1** is 1'b1, these bits set up the upper-limit filter value of the filter1.

Users cannot write to these bits when **CFDGFLECTR.PFLDAE** bit is 1'b0.

Users should only write to these bits when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

4.3.106 CFDGPFLPM0s

Global Pretended Network Filter List Payload Mask 0 Registers s = [1...4]h

Address: Refer to section
4.1

	b31	b30	b29	b28	b27	b26	b25	b24
FMASK[31:24]								
Value after reset	0	0	0	0	0	0	0	0
FMASK[23:16]								
Value after reset	0	0	0	0	0	0	0	0
FMASK[15:8]								
Value after reset	0	0	0	0	0	0	0	0
FMASK[7:0]								
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:0]	FMASK	Pretended Network Filter List Filter data mask field	Global Pretended Network Filter List Mask field bits for payload data field bits	R/W

4.3.106.1 CFDGPFLPM0s.FMASK

Pretended Network Filter List Filter data mask field

These bits set the payload filter data mask of the Pretended Network Filter List.

When **CFDGPFLPTr.GPFLRANG0** is 1'b0, these bits set up the payload data mask of the filter0.

When **CFDGPFLPTr.GPFLRANG0** is 1'b1, these bits set up the lower-limit filter value of the filter0.

Users cannot write to these bits when **CFDGPFLCTR.PFLDAE** bit is 1'b0.

Users should only write to these bits when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

4.3.107 CFDGPFLPM1s

Global Pretended Network Filter List Payload Mask 1 Registers s = [1...4]h

Address: Refer to section
4.1

	b31	b30	b29	b28	b27	b26	b25	b24
FMASK[31:24]								
Value after reset	0	0	0	0	0	0	0	0
FMASK[23:16]								
Value after reset	0	0	0	0	0	0	0	0
FMASK[15:8]								
Value after reset	0	0	0	0	0	0	0	0
FMASK[7:0]								
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:0]	FMASK	Pretended Network Filter List Filter data mask field	Global Pretended Network Filter List Mask field bits for payload data field bits	R/W

4.3.107.1 CFDGPFLPM1s.FMASK

Pretended Network Filter List Filter data mask field

These bits set the payload filter data mask of the Pretended Network Filter List.

When **CFDGPFLPTr.GPFLRANG1** is 1'b0, these bits set up the payload data mask of the filter1.

When **CFDGPFLPTr.GPFLRANG1** is 1'b1, these bits set up the lower-limit filter value of the filter1.

Users cannot write to these bits when **CFDGPFLCTR.PFLDAE** bit is 1'b0.

Users should only write to these bits when the related RS-CAN-FD channel is in CH_RESET or CH_HALT mode.

4.3.108 CFDGAFLIGNENT

Global AFL Ignore Entry Register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
Value after reset	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
Value after reset	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset	0	0	0	0	0	0	0	0
	IRN[7:0]							

Bit	Symbol	Bit name	Function	R/W
b[31:19]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[18:16]	ICN	Ignore Channel Number	Define channel number which ignores an AFL entry	R/W
b[15:9]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[8:0]	IRN	Ignore Rule Number	Define rule number which ignores an AFL entry	R/W

4.3.108.1 CFDGAFLIGNENT.IRN

Ignore Rule Number

These bits define the rule number which updates an AFL entry.

Users should enter, only the values between 0 and 383 (9'h17F) inclusive.

The setting range of these bits depends on the combination of parameters. For details, refer to 5.8 software constraint.

Users should only write to these bits when **CFDGAFLIGNCTR.IREN** bit is 1'b0.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

4.3.108.2 CFDGAFLIGNENT.ICN

Ignore Channel Number

These bits define the channel number which updates an AFL entry.

Users should enter, only the values between 0 and 7 (3'h7) inclusive.

The setting range of these bits depends on the combination of parameters. For details, refer to 5.8 software constraint.

Users should only write to these bits when **CFDGAFLIGNCTR.IREN** bit is 1'b0.

Users cannot write to this bit when the RS-CAN-FD module is in GL_SLEEP mode.

4.3.109 CFDGAFLIGNCTR

Global AFL Ignore Control Register

Address: Refer to section 4.1

	b31	b30	b29	b28	b27	b26	b25	b24
Value after reset	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
Value after reset	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset	0	0	0	0	0	0	0	IREN

Bit	Symbol	Bit name	Function	R/W
b[31:16]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[15:8]	KEY	Key code	These bits control the right or wrong of rewriting of a IREN bit.	W
b[7:1]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[0]	IREN	Ignore Rule Enable	1'b0: AFL entry number does not ignore 1'b1: AFL entry number ignores	R/W

4.3.109.1 CFDGAFLIGNCTR.IREN

Ignore Rule Enable

When this bit is set, the entry number (selected by **CFDGAFLIGNENT** register) is ignored.
This bit is cleared automatically when RS-CAN-FD module enters GL_RESET mode.

4.3.109.2 CFDGAFLIGNCTR.KEY

Key code

When C4h is written in these bits, the write of a **CFDGAFLIGNCTR.IREN** bit becomes available.

Read value from these bits is always 8'h0.

Users should write a **CFDGAFLIGNCTR.IREN** bit and the **CFDGAFLIGNCTR.KEY** bit simultaneously.

4.4 Identifier Bits Alignment

Standard Identifier (11 bit) format: ID28 – ID18 is aligned to b10 – b0

Extended Identifier (29 bit) format: ID28 – ID0 is aligned to b28 – b0

For Standard Identifier format bits 11 to 28 (b11 – b28) should be 18'h0.

Standard Identifier (11 bit) format

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
IDE=0	RTR	-	0	0	0	0	0	0	0	0	0	0	0	0	0
b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
0	0	0	0	0	ID28	ID27	ID26	ID25	ID24	ID23	ID22	ID21	ID20	ID19	ID18

Extended Identifier (29 bit) format

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
IDE=1	RTR	-	ID28	ID27	ID26	ID25	ID24	ID23	ID22	ID21	ID20	ID19	ID18	ID17	ID16
b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
ID15	ID14	ID13	ID12	ID11	ID10	ID9	ID8	ID7	ID6	ID5	ID4	ID3	ID2	ID1	ID0

Figure 4.1 Bit alignment for Standard and Extended Identifier format

4.5 Message Buffer Component Structure

4.5.1 Start Addresses

The start address for each of the Message Buffer Components is calculated using the number of related Message Buffer Components and the number of channels.

no_of_channels	8
no_of_RMBCPs_per_channel	16
no_of_RFMBCPs	8
no_of_CFMBCPs_per_channel	3
no_of_TMBCPs_per_channel	64

The start addresses for each register in the Message Buffer Component are depicted in Figure 4.2.

b = Message Buffer Component Index	MBCP	Register	p	Regular Start Address $n = [0...no_of_channels-1]$	Debugger Start Address $n = [0...no_of_channels-1]$
[0...no_of_RMBCPs_per_channel-1]	RMBCPb[i]	RMID	x	2000h + b*0080h + n*800h	-
		RMPTR	x	2004h + b*0080h + n*800h	-
		RMFDSTS	x	2008h + b*0080h + n*800h	-
		RMDFP	[0...15]	200Ch + p*0004h + b*0080h + n*800h	-
[0...no_of_RFMBCPs-1]	RFMBCPb[i]	RFIDE	x	6000h + b*0080h	E000h + b*0080h
		RFPTRE	x	6004h + b*0080h	E004h + b*0080h
		RFFDSTSE	x	6008h + b*0080h	E008h + b*0080h
		RFDFP	[0...15]	600Ch + p*0004h + b*0080h	E00Ch + p*0004h + b*0080h
[0...no_of_CFMBCPs_per_channel-1]	CFMBCPb[i]	CFIDE	x	6400h + b*0080h + n*180h	E400h + b*0080h + n*180h
		CFPTRE	x	6404h + b*0080h + n*180h	E404h + b*0080h + n*180h
		CFFDCSTSE	x	6408h + b*0080h + n*180h	E408h + b*0080h + n*180h
		CFDFP	[0...15]	640Ch + p*0004h + b*0080h + n*180h	E40Ch + p*0004h + b*0080h + n*180h
[0...no_of_TMBCPs_per_channel-1]	TMBCPb[i]	TMID	x	10000h + b*0080h + n* 2000h	-
		TMPTR	x	10004h + b*0080h + n* 2000h	-
		TMFDCTR	x	10008h + b*0080h + n* 2000h	-
		TMDFP	[0...15]	1000Ch + p*0004h + b*0080h + n* 2000h	-

'-' means Not Applicable

Figure 4.2 Message Buffer Component Register Start Addresses

The message buffer configuration consists of 4 types of message buffer components, namely RX Message Buffer Component (**CFDRMBCPb[i]**), RX FIFO Access Message Buffer Component (**CFDRFMBCPb[i]**), Common FIFO Access Message Buffer Component (**CFDCFMBCPb[i]**) and TX Message Buffer Component (**CFDTMBCPb[i]**). Where b = the Message Buffer Component index that has a range that varies based on the type of message buffer component and i = channel index that has a range from 0 to n. For a summary of this configuration, refer to Figure 8.1. For a detailed description of the number of and the different types of message buffers, refer to Section 8.

As mentioned in Section 4, each message buffer component consists of the following registers: Identifier (ID), Pointer (PTR) and Data Field (DFP) where p = the Data Field register index that has a range that varies based on the type of message buffer component.

Rc is the message buffer component register where c = Message Buffer Component Register index that has a range that varies based on the type of message buffer component.

A description of the registers, their associated bits and their accessibility is shown below the summary and detailed figures of each component.

In each of the figures below, a cell that contains '-' means reserved. Same behaviour as the reserved bits for registers in Section 4.3.

4.5.2 CFDRMBCPb[i]

RX Message Buffer Component b

	b31	b30	b29	b28	b27	b26	b25	b24
Value after reset								
	0	0	0	0	0	0	0	0
Value after reset								
	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset								
	0	0	0	0	0	0	0	0
Value after reset								
	b15	b14	b13	b12	b11	b10	b9	b8
Value after reset								
	0	0	0	0	0	0	0	0
Value after reset								
	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset								
	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:0]	Rc	RX Message Buffer Component c	Refer to Figures 4.3, 4.4 and the descriptions that follow for a detailed description of each register and its related bits, contained within this message buffer component.	R

(no_of_channels = 8)

(i = 0 .. no_of_channels – 1)

(b = RX Message Buffer Component index = [0...no_of_CFDRMBCPs_per_channel-1])

(no_of_CFDRMBCPs_per_channel = No. of RX Message Buffer Components per Channel = 16)

Where the total number of CFDRMBCPs = no_of_CFDRMBCPs_per_channel * no_of_channels = 16 * 8 = 128

as shown in Figure 8.1

(c = RX Message Buffer Component Register index = [0...no_of_REGS_per_CFDRMBCP-1])

no_of_REGS_per_CFDRMBCP = No. of Registers per RX Message Buffer Component = 19

4.5.2.1 Rc

RX Message Buffer Component c

The RX Message Buffer Component is made up of the following registers: **CFDRMID**, **CFDRMPTR**, **CFDRMFSTS**, and **CFDRMDFp**. Refer to Figure 4.4 for details of how to interpret the structure of this buffer component and how to access the respective registers.

RX Message Buffer Component (RMBCP)	
Rc	
R0	RX Message Buffer (b) ID Registers CHn
R1	RX Message Buffer (b) Pointer Registers CHn
R2	RX Message Buffer (b) CAN-FD Status Registers CHn
R3	RX Message Buffer (b) Data Field 0 Registers CHn
R4	RX Message Buffer (b) Data Field 1 Registers CHn
R5	RX Message Buffer (b) Data Field 2 Registers CHn
R6	RX Message Buffer (b) Data Field 3 Registers CHn
R7	RX Message Buffer (b) Data Field 4 Registers CHn
R8	RX Message Buffer (b) Data Field 5 Registers CHn
R9	RX Message Buffer (b) Data Field 6 Registers CHn
R10	RX Message Buffer (b) Data Field 7 Registers CHn
R11	RX Message Buffer (b) Data Field 8 Registers CHn
R12	RX Message Buffer (b) Data Field 9 Registers CHn
R13	RX Message Buffer (b) Data Field 10 Registers CHn
R14	RX Message Buffer (b) Data Field 11 Registers CHn
R15	RX Message Buffer (b) Data Field 12 Registers CHn
R16	RX Message Buffer (b) Data Field 13 Registers CHn
R17	RX Message Buffer (b) Data Field 14 Registers CHn
R18	RX Message Buffer (b) Data Field 15 Registers CHn

Figure 4.3 RX Message Buffer Component Summary

RX Message Buffer Component (RMBCP)																																		
Rc	p	Symbol	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
R0	x	CFDRMID	RMIDE	RMIRTF	RMID																				RMTS									
R1	x	CFDRMPTR	RMDLC		-	-	-	-	-	-	-	-	-	-	-	-	RMFL		RMDFD							RMBSI								
R2	x	CFDRMFSTS	RMPTR										-	-	-	-	-	RMFL		-	-	-	-	RMDFD										
R3	0	CFDRMDFp	RMDB((p*q)+(q-1))					RMDB((p*q)+(q-2))					RMDB((p*q)+(q-3))					RMDB((p*q)+(q-4))					RMDB((p*q)+(q-5))					RMDB((p*q)+(q-6))						
R(4...18)	[1...15]	CFDRMDFp	RMDB((p*q)+(q-1))					RMDB((p*q)+(q-2))					RMDB((p*q)+(q-3))					RMDB((p*q)+(q-4))					RMDB((p*q)+(q-5))					RMDB((p*q)+(q-6))						

Figure 4.4 RX Message Buffer Component Detailed

4.5.3 CFDRMID

RX Message Buffer ID Registers

Address: Refer to Figure 4.2

	b31	b30	b29	b28	b27	b26	b25	b24	
	RMIDE	RMRTR	-			RMID[28:24]			
Value after reset	0	0	0	0	0	0	0	0	
	b23	b22	b21	b20	b19	b18	b17	b16	
Value after reset	0	0	0	0	0	0	0	0	
	b15	b14	b13	b12	b11	b10	b9	b8	
Value after reset	0	0	0	0	0	0	0	0	
	b7	b6	b5	b4	b3	b2	b1	b0	
Value after reset	0	0	0	0	0	0	0	0	
Bit	Symbol	Bit name	Function						R/W
b[31]	RMIDE	RX Message Buffer IDE Bit	1'b0: STD-ID is stored 1'b1: EXT-ID is stored						R
b[30]	RMRTR	RX Message Buffer RTR Bit	1'b0: Data Frame 1'b1: Remote Frame						R
b[29]	-	Reserved	This bit is read as 0b. The write value should be always 0b.						R/W
b[28:0]	RMID	RX Message Buffer ID Field	STD-ID / EXT-ID fields						R

The RX Message Buffer ID register stores the ID field, IDE bit and RTR bit of the received message.

4.5.3.1 CFDRMID.RMID

RX Message Buffer ID Field

These are the bits of the STD-ID / EXT-ID fields of the message stored in the RX Message Buffer. For alignment of these bits in standard and extended frame format, see Section 4.4

Refer to Section 4.5.1 for details of how to interpret the structure of this buffer component.

4.5.3.2 CFDRMID.RMRTR

RX Message Buffer RTR Bit

This bit shows whether a Data Frame or a Remote Frame was stored in the RX Message Buffer.

Note: There are no remote frames in CAN FD format. In case a CAN-FD frame was received the register reflects the state of the received value (RRS bit in FD frame format).

4.5.3.3 CFDRMID.RMIDE

RX Message Buffer IDE Bit

This bit shows whether message with Standard Identifier or Extended Identifier was stored in the RX Message Buffer.

4.5.4 CFDRMPTR

RX Message Buffer Pointer Registers

Address: Refer to Figure 4.2

	b31	b30	b29	b28	b27	b26	b25	b24	
	RMDLC[3:0]								
Value after reset	0	0	0	0	0	0	0	0	
	b23	b22	b21	b20	b19	b18	b17	b16	
Value after reset	-	-	-	-	-	-	-	-	
	0	0	0	0	0	0	0	0	
	b15	b14	b13	b12	b11	b10	b9	b8	
	RMTS[15:8]								
Value after reset	0	0	0	0	0	0	0	0	
	b7	b6	b5	b4	b3	b2	b1	b0	
	RMTS[7:0]								
Value after reset	0	0	0	0	0	0	0	0	
Bit	Symbol	Bit name	Function						R/W
b[31:28]	RMDLC	RX Message Buffer DLC Field	No. of Data Bytes received in a CAN Frame						R
b[27:16]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[15:0]	RMTS	RX Message Buffer Timestamp Field	Timestamp value stored for the message stored in the RX Message Buffer						R

The RX Message Buffer Pointer register stores the DLC and Timestamp fields for the received message.

4.5.4.1 CFDRMPTR.RMTS

RX Message Buffer Timestamp Field

The Timestamp value taken at the capture point as configured by **CFDGFD CFG.TSCCFG** of the received message is stored in these bits.

In the case of can_race_ts_en=1, the information as which **CFDRMPTR.RMTS** is inputted from can_race_ts is reflected.

b15	b14	b13	b12	b11	b10	b9	b8
0		CH[2:0]			TSIDCNT[8:5]		
b7	b6	b5	b4	b3	b2	b1	b0
	TSIDCNT[4:0]				0	0	0

4.5.4.2 CFDRMPTR.RMDLC

RX Message Buffer DLC Field

The number of Data Bytes that were received in the RX Message Buffer is stored in these bits.

Please refer to Table 5 in ISO 11898-1 (2015) Specification for details defining the number of Data Bytes that were received.

Note: The max. capacity of the buffer belongs to the **CFDRMNB.RMPLS**.

4.5.5 CFDRMFSTS

RX Message Buffer CAN-FD Status Register

Address: Refer to Figure 4.2

	b31	b30	b29	b28	b27	b26	b25	b24	
RMPTR[15:8]									
Value after reset	0	0	0	0	0	0	0	0	
RM PTR[7:0]									
Value after reset	0	0	0	0	0	0	0	0	
RMIFL[1:0]									
Value after reset	0	0	0	0	0	0	0	0	
RMFDF RMBRS RMESI									
Value after reset	0	0	0	0	0	0	0	0	
Bit	Symbol	Bit name	Function						R/W
b[31:16]	RM PTR	RX Message Buffer Pointer Field	RX Message Buffer Pointer						R
b[15:10]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[9:8]	RMIFL	RX Message Buffer Information label Field	RX Message Buffer Information Label						R
b[7:3]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[2]	RMFDF	CAN FD Format bit	1'b0: Non CAN-FD frame received 1'b1: CAN-FD frame received						R
b[1]	RMBRS	Bit Rate Switch bit	1'b0: CAN-FD frame received with no bit rate switch 1'b1: CAN-FD frame received with bit rate switch						R
b[0]	RMESI	Error State Indicator bit	1'b0: CAN-FD frame received from error active node 1'b1: CAN-FD frame received from error passive node						R

The RX Message Buffer CAN-FD Status register shows the status of the FDF, BRS and ESI bits, Pointer of the received CAN-FD frame.

4.5.5.1 CFDRMFSTS.RMESI

Error State Indicator bit

This bit is the same value as the ESI bit of the received CAN-FD frame.

When the received FDF bit is 1'b0, means a CAN2.0 frame is received, 1'b0 is always stored to this bit.

4.5.5.2 CFDRMFSTS.RMBRS

Bit Rate Switch bit

This bit is the same value as the BRS bit of the received CAN-FD frame.

When the received FDF bit is 1'b0, means a CAN2.0 frame is received, 1'b0 is always stored to this bit.

4.5.5.3 CFDRMFSTS.RMFDF

CAN FD Format bit

This bit is the same value as the FDF bit of the received CAN-FD frame.

4.5.5.4 CFDRMFSTS.RMIFL

RX Message Buffer Information label Field

The Information Label value from the related Global Acceptance Filter List entry is stored in these bits.

4.5.5.5 CFDRMFDSTS.RMPTR

RX Message Buffer Pointer Field

The Pointer value from the related Global Acceptance Filter List entry is stored in these bits.

4.5.6 CFDRMDFp

RX Message Buffer Data Field p Registers

Address: Refer to Figure 4.2

	b31	b30	b29	b28	b27	b26	b25	b24	
	RMDB((p*q)+(q-1))[7:0]								
Value after reset	0	0	0	0	0	0	0	0	
	b23	b22	b21	b20	b19	b18	b17	b16	
Value after reset	0	0	0	0	0	0	0	0	
	b15	b14	b13	b12	b11	b10	b9	b8	
Value after reset	0	0	0	0	0	0	0	0	
	b7	b6	b5	b4	b3	b2	b1	b0	
Value after reset	0	0	0	0	0	0	0	0	
Bit	Symbol	Bit name	Function						R/W
b[31:24]	RMDB((p*q)+(q-1))	RX Message Buffer Data Byte ((p*q)+(q-1))	RX Message Buffer Data Byte ((p*q)+(q-1))						R
b[23:16]	RMDB((p*q)+(q-2))	RX Message Buffer Data Byte ((p*q)+(q-2))	RX Message Buffer Data Byte ((p*q)+(q-2))						R
b[15:8]	RMDB((p*q)+(q-3))	RX Message Buffer Data Byte ((p*q)+(q-3))	RX Message Buffer Data Byte ((p*q)+(q-3))						R
b[7:0]	RMDB((p*q)+(q-4))	RX Message Buffer Data Byte ((p*q)+(q-4))	RX Message Buffer Data Byte ((p*q)+(q-4))						R

(p = Data Field Register index = [0...no_of_CFDRMDFs-1])

(no_of_CFDRMDFs = No. of RX Message Buffer Data Registers)

no_of_CFDRMDFs = 16

(q = No. of Data Bytes = 4)

The RX Message Buffer Data Field p registers store the Data Bytes DB((p*q)+(q-4)) to DB((p*q)+(q-1)) of the received message.

4.5.6.1 CFDRMDFp.RMDB((p*q)+(q-4))

RX Message Buffer Data Byte ((p*q)+(q-4))

Data Byte ((p*q)+(q-4)) of the Message stored in the RX Message Buffer.

Unused Data Bytes will be filled with 8'h0.

4.5.6.2 CFDRMDFp.RMDB((p*q)+(q-3))

RX Message Buffer Data Byte ((p*q)+(q-3))

Data Byte ((p*q)+(q-3)) of the Message stored in the RX Message Buffer.

Unused Data Bytes will be filled with 8'h0.

4.5.6.3 CFDRMDFp.RMDB((p*q)+(q-2))

RX Message Buffer Data Byte ((p*q)+(q-2))

Data Byte ((p*q)+(q-2)) of the Message stored in the RX Message Buffer.

Unused Data Bytes will be filled with 8'h0.

4.5.6.4 CFDRMDFp.RMDB((p*q)+(q-1))

RX Message Buffer Data Byte ((p*q)+(q-1))



Data Byte (($p \cdot q$) + (q - 1)) of the Message stored in the RX Message Buffer.

Unused Data Bytes will be filled with 8'h0.

4.5.7 CFDRFMBCPb[i]

RX FIFO Access Message Buffer Component b

	b31	b30	b29	b28	b27	b26	b25	b24
Value after reset								
	0	0	0	0	0	0	0	0
Value after reset								
	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset								
	0	0	0	0	0	0	0	0
Value after reset								
	b15	b14	b13	b12	b11	b10	b9	b8
Value after reset								
	0	0	0	0	0	0	0	0
Value after reset								
	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset								
	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:0]	Rc	RX FIFO Access Message Buffer Component c	Refer to Figures 4.5, 4.6 and the descriptions that follow for a detailed description of each register and its related bits, contained within this message buffer component.	R

(no_of_channels = 8)

(i = 0 .. no_of_channels – 1)

(b = RX FIFO Access Message Buffer Component index = [0...no_of_CFDRFMBCPs-1])

(no_of_CFDRFMBCPs = No. of RX FIFO Access Message Buffer Components = 8) as shown in Figure 8.1

(c = RX FIFO Access Message Buffer Component Register index = [0...no_of_REGS_per_CFDRFBCP-1])

no_of_REGS_per_CFDRFBCP = No. of Registers per RX FIFO Access Message Buffer Component = 19

4.5.7.1 CFDRFMBCPb.Rc

RX FIFO Access Message Buffer Component c

The RX FIFO Access Message Buffer Component is made up of the following registers: **CFDRFID**, **CFDRFPTR**, **CFDRFFDSTS**, and **CFDRFDFp**. Refer to Figure 4.6 for details of how to interpret the structure of this buffer component and how to access the respective registers.

RX FIFO Access Message Buffer Component (RFMBCP)	
Rc	
R0	RX FIFO Access ID Registers
R1	RX FIFO Access Pointer Register
R2	RX FIFO Access CAN-FD Status Registers
R3	RX FIFO Access Data Field 0 Registers
R4	RX FIFO Access Data Field 1 Registers
R5	RX FIFO Access Data Field 2 Registers
R6	RX FIFO Access Data Field 3 Registers
R7	RX FIFO Access Data Field 4 Registers
R8	RX FIFO Access Data Field 5 Registers
R9	RX FIFO Access Data Field 6 Registers
R10	RX FIFO Access Data Field 7 Registers
R11	RX FIFO Access Data Field 8 Registers
R12	RX FIFO Access Data Field 9 Registers
R13	RX FIFO Access Data Field 10 Registers
R14	RX FIFO Access Data Field 11 Registers
R15	RX FIFO Access Data Field 12 Registers
R16	RX FIFO Access Data Field 13 Registers
R17	RX FIFO Access Data Field 14 Registers
R18	RX FIFO Access Data Field 15 Registers
R[19...31]	-

Figure 4.5 RX FIFO Access Message Buffer Component Summary

RX FIFO Access Message Buffer Component (RFMBCP)			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
R0	x	CFDRFID	RFIDE	RFRTF																														
R1	x	CFDRFPTR	RFDLCL	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-				
R2	x	CFDRFFDSTS	RFPTR										-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	RFIFL	RFDF	RFBRG	RFESI			
R3	0	CFDRFDp	RFDB((p*q)+(q-1))					RFDB((p*q)+(q-2))					RFDB((p*q)+(q-3))					RFDB((p*q)+(q-4))					RFDB((p*q)+(q-1))					RFDB((p*q)+(q-2))						
R[4...18]	[1...15]	CFDRFDp	RFDB((p*q)+(q-1))					RFDB((p*q)+(q-2))					RFDB((p*q)+(q-3))					RFDB((p*q)+(q-4))					RFDB((p*q)+(q-1))					RFDB((p*q)+(q-2))						
R[19...31]	x	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	

Figure 4.6 RX FIFO Access Message Buffer Component Detailed

4.5.8 CFDRFID

RX FIFO Access ID Registers

Address: Refer to Figure 4.2

	b31	b30	b29	b28	b27	b26	b25	b24
	RFIDE	RFRTR	-			RFID[28:24]		
Value after reset	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
					RFID[23:16]			
Value after reset	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
					RFID[15:8]			
Value after reset	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
					RFID[7:0]			
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31]	RFIDE	RX FIFO Buffer IDE Bit	1'b0: STD-ID has been received 1'b1: EXT-ID has been received	R
b[30]	RFRTR	RX FIFO Buffer RTR Bit	1'b0: Data Frame 1'b1: Remote Frame	R
b[29]	-	Reserved	This bit is read as 0b. The write value should be always 0b.	R/W
b[28:0]	RFID	RX FIFO Buffer ID Field	STD-ID / EXT-ID fields	R

The RX FIFO Access ID registers stores the ID field, IDE bit and RTR bit of the message.

4.5.8.1 CFDRFID.RFID

RX FIFO Buffer ID Field

These are the bits of the STD-ID / EXT-ID fields of the message in the FIFO Buffer.

For alignment of these bits in standard and extended frame format, see Section 4.4

4.5.8.2 CFDRFID.RFRTR

RX FIFO Buffer RTR Bit

This bit shows whether a Data Frame or a Remote Frame was stored in the FIFO Buffer.

Note: There are no remote frames in CAN FD format. In case a CAN-FD frame was received the register reflects the state of the received value (RRS bit in FD frame format).

4.5.8.3 CFDRFID.RFIDE

RX FIFO Buffer IDE Bit

This bit shows whether message with Standard Identifier or Extended Identifier was received in the FIFO Buffer.

4.5.9 CFDRFPTR

RX FIFO Access Pointer Register

Address: Refer to Figure 4.2

	b31	b30	b29	b28	b27	b26	b25	b24
RFDLC[3:0]								
Value after reset	0	0	0	0	0	0	0	0
b23 b22 b21 b20 b19 b18 b17 b16								
Value after reset	0	0	0	0	0	0	0	0
b15 b14 b13 b12 b11 b10 b9 b8								
Value after reset	0	0	0	0	0	0	0	0
RFTS[15:8]								
Value after reset	0	0	0	0	0	0	0	0
b7 b6 b5 b4 b3 b2 b1 b0								
Value after reset	0	0	0	0	0	0	0	0
RFTS[7:0]								

Bit	Symbol	Bit name	Function	R/W
b[31:28]	RFDLC	RX FIFO Buffer DLC Field	No. of Data Bytes received in a CAN Frame	R
b[27:16]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[15:0]	RFTS	RX FIFO Timestamp Value	Timestamp value of the received CAN Frame	R

The FIFO Access Pointer registers store the DLC & Timestamp fields for the received message.

4.5.9.1 CFDRFPTR.RFTS

RX FIFO Timestamp Value

The Timestamp value taken at the capture point as configured by **CFDGFD CFG.TSCCFG** of the received message is stored in these bits.

In the case of can_race_ts_en=1, the information as which **CFDRFPTR.RFTS** is inputted from can_race_ts is reflected.

b15	b14	b13	b12	b11	b10	b9	b8
0		CH[2:0]		TSIDCNT[8:5]			
b7	b6	b5	b4	b3	b2	b1	b0
TSIDCNT[4:0]				0	0	0	0

4.5.9.2 CFDRFPTR.RFDLC

RX FIFO Buffer DLC Field

The number of Data Bytes that were received in the RX FIFO Buffer is stored in these bits.

Please refer to Table 5 in ISO 11898-1 (2015) Specification for details defining the number of Data Bytes that were received.

4.5.10 CFDRFFDSTS

RX FIFO Access CAN-FD Status Register

Address: Refer to Figure 4.2

	b31	b30	b29	b28	b27	b26	b25	b24	
	CFDRF PTR[16:8]								
Value after reset	0	0	0	0	0	0	0	0	
	b23	b22	b21	b20	b19	b18	b17	b16	
	CFDRF PTR[7:0]								
Value after reset	0	0	0	0	0	0	0	0	
	b15	b14	b13	b12	b11	b10	b9	b8	
	-	-	-	-	-	-	RFIFL[1:0]		
Value after reset	0	0	0	0	0	0	0	0	
	b7	b6	b5	b4	b3	b2	b1	b0	
	-	-	-	-	-	RFFDF	RFBRS	RFESI	
Value after reset	0	0	0	0	0	0	0	0	

Bit	Symbol	Bit name	Function	R/W
b[31:16]	CFDRF PTR	RX FIFO Buffer Pointer Field	FIFO Buffer Pointer	R
b[15:10]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[9:8]	RFIFL	RX FIFO Buffer Information label Field	RX FIFO Buffer Information Label	R
b[7:3]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[2]	RFFDF	CAN FD Format bit	1'b0: Non CAN-FD frame received 1'b1: CAN-FD frame received	R
b[1]	RFBRS	Bit Rate Switch bit	1'b0: CAN-FD frame received with no bit rate switch 1'b1: CAN-FD frame received with bit rate switch	R
b[0]	RFESI	Error State Indicator bit	1'b0: CAN-FD frame received from error active node 1'b1: CAN-FD frame received from error passive node	R

The RX FIFO Access CAN-FD Status register shows the status of the FDF, BRS and ESI bits, Pointer of the received CAN-FD frame.

4.5.10.1 CFDRFFDSTS.RFESI

Error State Indicator bit

This bit is the same value as the ESI bit of the received CAN-FD frame.

When the received FDF bit is 1'b0, means a CAN2.0 frame is received, 1'b0 is always stored to this bit.

4.5.10.2 CFDRFFDSTS.RFBRS

Bit Rate Switch bit

This bit is the same value as the BRS bit of the received CAN-FD frame.

When the received FDF bit is 1'b0, means a CAN2.0 frame is received, 1'b0 is always stored to this bit.

4.5.10.3 CFDRFFDSTS.RFFDF

CAN FD Format bit

This bit is the same value as the FDF bit of the received CAN-FD frame.

4.5.10.4 CFDRFFDSTS.RFIFL

RX FIFO Buffer Information Label Field

The Information label value from the related Global Acceptance Filter List entry is stored in these bits.

4.5.10.5 CFDRFFDSTS.CFDRF PTR

RX FIFO Buffer Pointer Field

The Pointer value from the related Global Acceptance Filter List entry is stored in these bits.

4.5.11 CFDRFDFp

RX FIFO Access Data Field p Registers

Address: Refer to Figure 4.2

	b31	b30	b29	b28	b27	b26	b25	b24
RFDB((p*q)+(q-1))[7:0]								
Value after reset	0	0	0	0	0	0	0	0
RFDB((p*q)+(q-2))[7:0]								
Value after reset	0	0	0	0	0	0	0	0
RFDB((p*q)+(q-3))[7:0]								
Value after reset	0	0	0	0	0	0	0	0
RFDB((p*q)+(q-4))[7:0]								
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:24]	RFDB((p*q)+(q-1))	RX FIFO Buffer Data Byte ((p*q)+(q-1))	FIFO Buffer Data Byte ((p*q)+(q-1))	R
b[23:16]	RFDB((p*q)+(q-2))	RX FIFO Buffer Data Byte ((p*q)+(q-2))	FIFO Buffer Data Byte ((p*q)+(q-2))	R
b[15:8]	RFDB((p*q)+(q-3))	RX FIFO Buffer Data Byte ((p*q)+(q-3))	FIFO Buffer Data Byte ((p*q)+(q-3))	R
b[7:0]	RFDB((p*q)+(q-4))	RX FIFO Buffer Data Byte ((p*q)+(q-4))	FIFO Buffer Data Byte ((p*q)+(q-4))	R

(p = Data Field Register index = [0...no_of_CFDRFDFs-1])

(no_of_CFDRFDFs = No. of RX FIFO Access Data Field Registers)

no_of_CFDRFDFs = 16

(q = No. of Data Bytes = 4)

The RX FIFO Access Data Field Registers store the Data Bytes DB((p*q)+(q-4)) to DB((p*q)+(q-1)) of the received message.

4.5.11.1 CFDRFDFp.RFDB((p*q)+(q-4))

RX FIFO Buffer Data Byte ((p*q)+(q-4))

Data Byte ((p*q)+(q-4)) of the Message present in the FIFO Buffer.

Unused Data Bytes will be filled with 8'h0, according to there configured data payload size **CFDRFCCa.RFPLS**.

4.5.11.2 CFDRFDFp.RFDB((p*q)+(q-3))

RX FIFO Buffer Data Byte ((p*q)+(q-3))

Data Byte ((p*q)+(q-3)) of the Message present in the FIFO Buffer.

Unused Data Bytes will be filled with 8'h0.

4.5.11.3 CFDRFDFp.RFDB((p*q)+(q-2))

RX FIFO Buffer Data Byte ((p*q)+(q-2))

Data Byte ((p*q)+(q-2)) of the Message present in the FIFO Buffer.

Unused Data Bytes will be filled with 8'h0.

4.5.11.4 CFDRFDFp.RFDB((p*q)+(q-1))

RX FIFO Buffer Data Byte ((p*q)+(q-1))

Data Byte ((p*q)+(q-1)) of the Message present in the FIFO Buffer.

Unused Data Bytes will be filled with 8'h0.

4.5.12 CFDCFMBCPb[i]

Common FIFO Access Message Buffer Component b

	b31	b30	b29	b28	b27	b26	b25	b24
Value after reset								
	0	0	0	0	0	0	0	0
Value after reset								
	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset								
	0	0	0	0	0	0	0	0
Value after reset								
	b15	b14	b13	b12	b11	b10	b9	b8
Value after reset								
	0	0	0	0	0	0	0	0
Value after reset								
	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset								
	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:0]	Rc	Common FIFO Access Message Buffer Component c	Refer to Figures 4.7, 4.8 and the descriptions that follow for a detailed description of each register and its related bits, contained within this message buffer component.	R

(no_of_channels = 8)

(i = 0 .. no_of_channels - 1)

(b = Common FIFO Message Buffer Component index = [0...no_of_CFDCFMBPs_per_channel-1])

(no_of_CFDCFMBPs_per_channel = No. of Common FIFO Message Buffer Components per Channel = 3)

Where the total number of CFDCFMBPs = no_of_CFDCFMBPs_per_channel * no_of_channels = 3 * 8 = 24

as shown in Figure 8.1

(c = Common FIFO Message Buffer Component Register index = [0...no_of_REGS_per_CFDCFBCP-1])
no_of_REGS_per_CFDCFBCP = No. of Registers per Common FIFO Message Buffer Component = 19

4.5.12.1 CFDCFMBCPb.Rc

Common FIFO Access Message Buffer Component c

The Common FIFO Access Message Buffer Component is made up of the following registers: **CFDCFID**, **CFDCF PTR**, **CFFDSTS**, and **CFDCFDFp**. Refer to Figure 4.8 for details of how to interpret the structure of this buffer component and how to access the respective registers.

Common FIFO Access Message Buffer Component (CFMBCP)	
Rc	
R0	Common FIFO Access ID Registers
R1	Common FIFO Access Pointer Register
R2	Common FIFO Access CAN-FD Status Registers
R3	Common FIFO Access Data Field 0 Registers
R4	Common FIFO Access Data Field 1 Registers
R5	Common FIFO Access Data Field 2 Registers
R6	Common FIFO Access Data Field 3 Registers
R7	Common FIFO Access Data Field 4 Registers
R8	Common FIFO Access Data Field 5 Registers
R9	Common FIFO Access Data Field 6 Registers
R10	Common FIFO Access Data Field 7 Registers
R11	Common FIFO Access Data Field 8 Registers
R12	Common FIFO Access Data Field 9 Registers
R13	Common FIFO Access Data Field 10 Registers
R14	Common FIFO Access Data Field 11 Registers
R15	Common FIFO Access Data Field 12 Registers
R16	Common FIFO Access Data Field 13 Registers
R17	Common FIFO Access Data Field 14 Registers
R18	Common FIFO Access Data Field 15 Registers
R[19...31]	-

Figure 4.7 Common FIFO Access Message Buffer Component Summary

Common FIFO Access Message Buffer Component (CFMBCP)			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
Rc	p	Symbol	CFIDE	CFTR	CFLEN																													
R0	x	CFDCFID	CFID																															
R1	x	CFDCFPT	CFDLC	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-				
R2	x	CFDCFFDCSTS	CFPTR										CFIFL	CFDF	CFBRS	CFESI				
R3	0	CFDCFDFp	CFDB((p*q)+(q-1))					CFDB((p*q)+(q-2))					CFDB((p*q)+(q-3))					CFDB((p*q)+(q-4))					CFDB((p*q)+(q-1))					CFDB((p*q)+(q-2))						
R[4...18]	[1...15]	CFDCFDFp	CFDB((p*q)+(q-1))					CFDB((p*q)+(q-2))					CFDB((p*q)+(q-3))					CFDB((p*q)+(q-4))					CFDB((p*q)+(q-1))					CFDB((p*q)+(q-2))						
R[19..31]	x	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-			

Figure 4.8 Common FIFO Access Message Buffer Component Detailed

4.5.13 CFDCFID

Common FIFO Access ID Registers

Address: Refer to Figure 4.2

	b31	b30	b29	b28	b27	b26	b25	b24
	CFIDE	CFRTR	THLEN			CFID[28:24]		
Value after reset	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
					CFID[23:16]			
Value after reset	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
				CFID[15:8]				
Value after reset	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
				CFID[7:0]				
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31]	CFIDE	Common FIFO Buffer IDE Bit	1'b0: STD-ID will be transmitted or has been received 1'b1: EXT-ID will be transmitted or has been received	R/W
b[30]	CFRTR	Common FIFO Buffer RTR	1'b0: Data Frame 1'b1: Remote Frame	R/W
b[29]	THLEN	THL Entry enable	TX FIFO Mode: 1'b0: Entry will not be stored in THL after successful TX. 1'b1: Entry will be stored in THL after successful TX. RX FIFO Mode: Reserved, this bit is read as 0	R/W
b[28:0]	CFID	Common FIFO Buffer ID Field	STD-ID / EXT-ID fields	R/W

The Common FIFO Access ID registers store the ID field, IDE bit and RTR bit of the message.

In TX mode, users can read data from the FIFO, only for the current entry based on the write pointer value, not for the other entries.

4.5.13.1 CFDCFID.CFID

Common FIFO Buffer ID Field

These are the bits of the STD-ID / EXT-ID fields of the message in the FIFO Buffer.

For alignment of these bits in standard and extended frame format, see Section 4.4

In TX mode, users can write and read from FIFO buffers.

In RX mode, users can only read data from FIFO buffers.

In GW mode, users cannot write data to the FIFO buffers.

4.5.13.2 CFDCFID.THLEN

THL Entry enable

This bit controls the storage of an entry corresponding to the transmitted message in the TX History list at the end of a successful transmission.

In TX mode, users can write and read from FIFO buffers.

In RX mode, users can only read data from FIFO buffers.

In GW mode, users cannot write data to the FIFO buffers.

4.5.13.3 CFDCFID.CFRTR

Common FIFO Buffer RTR Bit

This bit selects whether a Data Frame or a Remote Frame will be transmitted from or was received in the FIFO Buffer.

Note: There are no remote frames in CAN FD format. In case a CAN-FD frame was received (RX mode) the register reflects the state of the received value (RRS bit in FD frame format). In case of CAN-FD transmission (TX or GW mode **CFDCFID.CFFDF =1'b1**) the bit is always transmitted dominant (Data Frame).

In TX mode, users can write and read from FIFO buffers.

In RX mode, users can only read data from FIFO buffers.

In GW mode, users cannot write data to the FIFO buffers.

4.5.13.4 CFDCFID.CFIDE

Common FIFO Buffer IDE Bit

This bit selects whether a message with EXT-ID or STD-ID will be transmitted from or was received in the FIFO Buffer.

In TX mode, users can write and read from FIFO buffers.

In RX mode, users can only read data from FIFO buffers.

In GW mode, users cannot write data to the FIFO buffers.

4.5.14 CFDCF PTR

Common FIFO Access Pointer Registers

Address: Refer to Figure 4.2

	b31	b30	b29	b28	b27	b26	b25	b24
CFDLC[3:0]								
Value after reset	0	0	0	0	0	0	0	0
b23 b22 b21 b20 b19 b18 b17 b16								
Value after reset	-	-	-	-	-	-	-	-
	0	0	0	0	0	0	0	0
b15 b14 b13 b12 b11 b10 b9 b8								
Value after reset	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
CFTS[7:0]								
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:28]	CFDLC	Common FIFO Buffer DLC Field	No. of Data Bytes received in a CAN Frame, or to be transmitted in a CAN Frame	R/W
b[27:16]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[15:0]	CFTS	Common FIFO Timestamp Value	Timestamp value of the received CAN Frame (FIFO in RX Mode)	R/W

The Common FIFO Access Pointer registers store the DLC and Timestamp fields.

In TX mode, users can read data from the FIFO, only for the current entry based on the write pointer value, not for the other entries.

4.5.14.1 CFDCF PTR.CFTS

Common FIFO Timestamp Value

The Timestamp value taken at the capture point as configured by **CFDGFD CFG.TSCCFG** of the received message is stored in these bits (if FIFO is configured in RX mode).

In TX mode, users can write and read from FIFO buffers.

In RX mode, users can only read data from FIFO buffers.

In GW mode, users cannot write data to the FIFO buffers.

In the case of `can_race_ts_en=1`, the information as which **CFDCF PTR.CFTS** is inputted from `can_race_ts` is reflected.

b15	b14	b13	b12	b11	b10	b9	b8
0		CH[2:0]		TSIDCNT[8:5]			
b7	b6	b5	b4	b3	b2	b1	b0
TSIDCNT[4:0]				0	0	0	0

4.5.14.2 CFDCF PTR.CFDLC

Common FIFO Buffer DLC Field

The number of Data Bytes that were received in the FIFO Buffer or are to be transmitted, is stored in these bits.

Please refer to Table 5 in ISO 11898-1 (2015) Specification for details defining the number of Data Bytes

In TX mode, users can write and read from FIFO buffers.

In RX mode, users can only read data from FIFO buffers.

In GW mode, users cannot write data to the FIFO buffers.

Users cannot read data for the other entries in the FIFO when configured in TX mode.

4.5.15 CFDCFFDCSTS

Common FIFO Access CAN-FD Control/Status Register

Address: Refer to Figure 4.2

	b31	b30	b29	b28	b27	b26	b25	b24
CFPTR[15:8]								
Value after reset	0	0	0	0	0	0	0	0
CFPTR[7:0]								
Value after reset	0	0	0	0	0	0	0	0
CFIFL[1:0]								
Value after reset	-	-	-	-	-	-	0	0
CFFDF, CFBRS, CFESI								
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:16]	CFPTR	Common FIFO Buffer Pointer Field	FIFO Message Buffer Pointer	R/W
b[15:10]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[9:8]	CFIFL	COMMON FIFO Buffer Information label Field	COMMON FIFO Buffer Information Label	R/W
b[7:3]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W
b[2]	CFFDF	CAN FD Format bit	1'b0: Non CAN-FD frame received or to transmit 1'b1: CAN-FD frame received or to transmit	R/W
b[1]	CFBRS	Bit Rate Switch bit	1'b0: CAN-FD frame received or to transmit with no bit rate switch 1'b1: CAN-FD frame received or to transmit with bit rate switch	R/W
b[0]	CFESI	Error State Indicator bit	1'b0: CAN-FD frame received from or to transmit by error active node 1'b1: CAN-FD frame received from or to transmit by error passive node	R/W

The Common FIFO Access CAN-FD Status register shows the status of the FDF, BRS and ESI bits, Pointer of the received CAN-FD frame or the CAN-FD frame to transmit.

In TX mode, users can read data from the FIFO, only for the current entry based on the write pointer value, not for the other entries.

4.5.15.1 CFDCFFDCSTS.CFESI

Error State Indicator bit

In TX mode, users can write and read from FIFO buffers.

In RX mode, users can only read data from FIFO buffers.

In GW mode, users cannot write data to the FIFO buffers.

In RX or GW mode, this bit is updated with the ESI bit value of the CAN-FD frame when it has been received, indicating the error state of the transmitting node.

In RX or GW mode 1'b0 is stored to this bit when the received FDF bit is 1'b0, means a CAN 2.0 frame is received.

In TX mode, if the RS-CAN-FD module is not in error passive, then this bit equals the write value, else it is don't care and the bit is transmitted as 1'b1 on the CAN bus; indicating that this is an error passive node.

4.5.15.2 CFDCFFDCSTS.CFBRS

Bit Rate Switch bit

In TX mode, users can write and read from FIFO buffers.

In RX mode, users can only read data from FIFO buffers.

In GW mode, users cannot write data to the FIFO buffers.

In RX or GW mode, this bit is updated with the BRS bit value of the CAN-FD frame when it has been received, indicating whether there is a bit rate switch (1'b1) or not (1'b0) on the CAN-FD frame.

In RX or GW mode 1'b0 is stored to this bit when the received FDF bit is 1'b0, means a CAN 2.0 frame is received.

In TX mode, the RS-CAN-FD module will either transmit a 1'b0 to indicate no bit rate switch in the frame to be transmitted or a 1'b1 to indicate a bit rate switch in the frame to be transmitted.

4.5.15.3 CFDCFFDCSTS.CFFDF

CAN FD Format bit

In TX mode, users can write and read from FIFO buffers.

In RX mode, users can only read data from FIFO buffers.

In GW mode, users cannot write data to the FIFO buffers.

In RX or GW mode, this bit is updated with the FDF bit value of the CAN frame when it has been received, indicating whether it is a CAN 2.0 frame (1'b0) or a CAN-FD frame (1'b1).

In TX mode, the RS-CAN-FD module will either transmit a 1'b0 to indicate a CAN 2.0 frame is to be transmitted or a 1'b1 to indicate a CAN-FD frame is to be transmitted.

4.5.15.4 CFDCFFDCSTS.CFIFL

Common FIFO Buffer Information Label Field

If the Common FIFO is configured in TX Mode, the value programmed in **CFDCFFDCSTS.CFIFL[1:0]** will be stored together with further message information, to the TX History List after successful transmission of the message.

The Information label value from the related Global Acceptance Filter List entry is stored in these bits (if FIFO is configured in either RX or GW mode).

In TX mode, users can write and read from FIFO buffers.

In RX mode, users can only read data from FIFO buffers.

In GW mode, users cannot write data to the FIFO buffers.

4.5.15.5 CFDCFFDCSTS.CFPTR

Common FIFO Buffer Pointer Field

If the Common FIFO is configured in TX Mode, the value programmed in **CFDCFFDCSTS.CFPTR[15:0]** will be stored together with further message information, to the TX History List after successful transmission of the message.

The Pointer value from the related Global Acceptance Filter List entry is stored in these bits (if FIFO is configured in either RX or GW mode).

In TX mode, users can write and read from FIFO buffers.

In RX mode, users can only read data from FIFO buffers.

In GW mode, users cannot write data to the FIFO buffers.

4.5.16 CFDCFDFp

Common FIFO Access Data Field p Registers

Address: Refer to Figure 4.2

	b31	b30	b29	b28	b27	b26	b25	b24
	CFDB((p*q)+(q-1))[7:0]							
Value after reset	0	0	0	0	0	0	0	0
	b23	b22	b21	b20	b19	b18	b17	b16
	CFDB((p*q)+(q-2))[7:0]							
Value after reset	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8
	CFDB((p*q)+(q-3))[7:0]							
Value after reset	0	0	0	0	0	0	0	0
	b7	b6	b5	b4	b3	b2	b1	b0
	CFDB((p*q)+(q-4))[7:0]							
Value after reset	0	0	0	0	0	0	0	0
Bit	Symbol	Bit name	Function	R/W				
b[31:24]	CFDB((p*q)+(q-1))	Common FIFO Buffer Data Byte ((p*q)+(q-1))	FIFO Buffer Data Byte ((p*q)+(q-1))					
b[23:16]	CFDB((p*q)+(q-2))	Common FIFO Buffer Data Byte ((p*q)+(q-2))	FIFO Buffer Data Byte ((p*q)+(q-2))					
b[15:8]	CFDB((p*q)+(q-3))	Common FIFO Buffer Data Byte ((p*q)+(q-3))	FIFO Buffer Data Byte ((p*q)+(q-3))					
b[7:0]	CFDB((p*q)+(q-4))	Common FIFO Buffer Data Byte ((p*q)+(q-4))	FIFO Buffer Data Byte ((p*q)+(q-4))					

(p = Data Field Register index = [0...no_of_CFDCFDFs-1])

(no_of_CFDCFDFs = No. of Common FIFO Access Data Field Registers)

no_of_CFDCFDFs = 16

(q = No. of Data Bytes = 4)

The FIFO Access Data Field p registers store the Data Bytes DB((p*q)+(q-4)) to DB((p*q)+(q-1)) of the message.

In TX mode, users can read data from the FIFO, only for the current entry based on the write pointer value, not for the other entries.

4.5.16.1 CFDCFDFp.CFDB((p*q)+(q-4))

Common FIFO Buffer Data Byte ((p*q)+(q-4))

Data Byte ((p*q)+(q-4)) of the Message present in the FIFO Buffer.

In TX mode, users can write and read from FIFO buffers.

In RX mode, users can only read data from FIFO buffers.

In GW mode, users cannot write data to the FIFO buffers.

In RX or GW mode, unused Data Bytes will be filled with 8'h0, according to their configured data payload size **CFDCFCCd.CFPLS**.

4.5.16.2 CFDCFDFp.CFDB((p*q)+(q-3))

Common FIFO Buffer Data Byte ((p*q)+(q-3))

Data Byte ((p*q)+(q-3)) of the Message present in the FIFO Buffer.

In TX mode, users can write and read from FIFO buffers.

In RX mode, users can only read data from FIFO buffers.

In GW mode, users cannot write data to the FIFO buffers.

In RX or GW mode, unused Data Bytes will be filled with 8'h0, according to their configured data payload size **CFDCFCCd.CFPLS**.

4.5.16.3 CFDCFDFp.CFDB((p*q)+(q-2))

Common FIFO Buffer Data Byte ((p*q)+(q-2))

Data Byte ((p*q)+(q-2)) of the Message present in the FIFO Buffer.

In TX mode, users can write and read from FIFO buffers.

In RX mode, users can only read data from FIFO buffers.

In GW mode, users cannot write data to the FIFO buffers.

In RX or GW mode, unused Data Bytes will be filled with 8'h0, according to their configured data payload size **CFDCFCCd.CFPLS**.

4.5.16.4 CFDCFDFp.CFDB((p*q)+(q-1))

Common FIFO Buffer Data Byte ((p*q)+(q-1))

Data Byte ((p*q)+(q-1)) of the Message present in the FIFO Buffer.

In TX mode, users can write and read from FIFO buffers.

In RX mode, users can only read data from FIFO buffers.

In GW mode, users cannot write data to the FIFO buffers.

In RX or GW mode, unused Data Bytes will be filled with 8'h0, according to their configured data payload size **CFDCFCCd.CFPLS**.

4.5.17 CFDTMBCPb[i]

TX Message Buffer Component b

	b31	b30	b29	b28	b27	b26	b25	b24
Value after reset								
	0	0	0	0	0	0	0	0
Value after reset								
	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset								
	0	0	0	0	0	0	0	0
Value after reset								
	b15	b14	b13	b12	b11	b10	b9	b8
Value after reset								
	0	0	0	0	0	0	0	0
Value after reset								
	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset								
	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:0]	Rc	TX Message Buffer Component c	Refer to Figures 4.9, 4.10 and the descriptions that follow for a detailed description of each register and its related bits, contained within this message buffer component.	R

(no_of_channels = 8)

(i = 0 .. no_of_channels – 1)

(b = TX Message Buffer Component index = [0...no_of_CFDTMBCPs_per_channel-1])

(no_of_CFDTMBCPs_per_channel = No. of TX Message Buffer Components per Channel = 64)

Where the total number of CFDTMBCPs= no_of_CFDTMBCPs_per_channel * no_of_channels = 64 * 8 = 512

as shown in Figure 8.1

(c = TX Message Buffer Component Register index = [0...no_of_REGS_per_CFDTMBCP-1])

no_of_REGS_per_CFDTMBCP = No. of Registers per TX Message Buffer Component = 19

4.5.17.1 CFDTMBCPb.Rc

TX Message Buffer Component c

The TX Message Buffer Component is made up of the following registers: **CFDTMID**, **CFDTMPTR**, **CFDTMFCTR**, and **CFDTMDFp**. Refer to Figure 4.10 for details of how to interpret the structure of this buffer component and how to access the respective registers.

TX Message Buffer Component (TMBCP)	
Rc	
R0	TX Message Buffer (b) ID Registers CHn
R1	TX Message Buffer (b) Pointer Registers CHn
R2	TX Message Buffer (b) CAN-FD Status Registers CHn
R3	TX Message Buffer (b) Data Field 0 Registers CHn
R4	TX Message Buffer (b) Data Field 1 Registers CHn
R5	TX Message Buffer (b) Data Field 2 Registers CHn
R6	TX Message Buffer (b) Data Field 3 Registers CHn
R7	TX Message Buffer (b) Data Field 4 Registers CHn
R8	TX Message Buffer (b) Data Field 5 Registers CHn
R9	TX Message Buffer (b) Data Field 6 Registers CHn
R10	TX Message Buffer (b) Data Field 7 Registers CHn
R11	TX Message Buffer (b) Data Field 8 Registers CHn
R12	TX Message Buffer (b) Data Field 9 Registers CHn
R13	TX Message Buffer (b) Data Field 10 Registers CHn
R14	TX Message Buffer (b) Data Field 11 Registers CHn
R15	TX Message Buffer (b) Data Field 12 Registers CHn
R16	TX Message Buffer (b) Data Field 13 Registers CHn
R17	TX Message Buffer (b) Data Field 14 Registers CHn
R18	TX Message Buffer (b) Data Field 15 Registers CHn

Figure 4.9 TX Message Buffer Component Summary

		TX Message Buffer Component (TMBCP)																																
Rc	p	Symbol	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
R0	x	CFDTMID	TMIDE	TMRTR	THLEN																													
R1	x	CFDTMPTR	TMDLC		-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-		
R2	x	CFDTMFDFCTR	TMPTR												-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	TMIFL	TMDF	TMBRS	TMESI	
R3	0	CFDTMDFp	TMDB((p*q)+(q-1))				TMDB((p*q)+(q-2))				TMDB((p*q)+(q-3))				TMDB((p*q)+(q-4))				TMDB((p*q)+(q-1))				TMDB((p*q)+(q-2))				TMDB((p*q)+(q-3))				TMDB((p*q)+(q-4))			
R[4...18]	[1...15]	CFDTMDFp	TMDB((p*q)+(q-1))				TMDB((p*q)+(q-2))				TMDB((p*q)+(q-3))				TMDB((p*q)+(q-4))				TMDB((p*q)+(q-1))				TMDB((p*q)+(q-2))				TMDB((p*q)+(q-3))				TMDB((p*q)+(q-4))			

Figure 4.10 TX Message Buffer Component Detailed

4.5.18 CFDTMID

TX Message Buffer ID Registers

Address: Refer to Figure 4.2

	b31	b30	b29	b28	b27	b26	b25	b24
TMID[28:24]								
Value after reset	0	0	0	0	0	0	0	0
TMID[23:16]								
Value after reset	0	0	0	0	0	0	0	0
TMID[15:8]								
Value after reset	0	0	0	0	0	0	0	0
TMID[7:0]								
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31]	TMIDE	TX Message Buffer IDE Bit	1'b0: STD-ID will be transmitted 1'b1: EXT-ID will be transmitted	R/W
b[30]	TMRTR	TX Message Buffer RTR Bit	1'b0: Data Frame 1'b1: Remote Frame	R/W
b[29]	THLEN	Tx History List Entry	1'b0: Entry not stored in THL after successful TX 1'b1: Entry stored in THL after successful TX	R/W
b[28:0]	TMID	TX Message Buffer ID Field	STD-ID / EXT-ID fields	R/W

Each TX Message Buffer ID register is used to store the ID, IDE, RTR fields and history configuration of the message to be transmitted from the associated buffer.

4.5.18.1 CFDTMID.TMID

TX Message Buffer ID Field

These are the bits of the STD-ID / EXT-ID fields of the message stored in this TX MB.
For alignment of these bits in standard and extended frame format, see Section 4.4

Users should not write to these bits when corresponding channel is in CH_SLEEP.

4.5.18.2 CFDTMID.THLEN

Tx History List Entry

This bit controls the storage of an entry corresponding to the transmitted message in the TX History list at the end of a successful transmission.

Users should not write to these bits when corresponding channel is in CH_SLEEP.

4.5.18.3 CFDTMID.TMRTR

TX Message Buffer RTR Bit

This bit selects whether a Data Frame or a Remote Frame will be transmitted from this TX Message Buffer.

Note: There are no remote frames in CAN FD format. In case of CAN-FD transmission (**CFDTMFCTR.CFFDF =1'b1**) the bit is always transmitted dominant (Data Frame).

Users should not write to these bits when corresponding channel is in CH_SLEEP.

4.5.18.4 CFDTMID.TMIDE

TX Message Buffer IDE Bit

This bit selects whether a message with EXT-ID or STD-ID will be transmitted from this TX Message Buffer.

Users should not write to these bits when corresponding channel is in CH_SLEEP.

4.5.19 CFDTMPTR

TX Message Buffer Pointer Registers

Address: Refer to Figure 4.2

	b31	b30	b29	b28	b27	b26	b25	b24
TMDLC[3:0]								
Value after reset	0	0	0	0	0	0	0	0
b23 b22 b21 b20 b19 b18 b17 b16								
Value after reset	-	-	-	-	-	-	-	-
	0	0	0	0	0	0	0	0
b15 b14 b13 b12 b11 b10 b9 b8								
Value after reset	-	-	-	-	-	-	-	-
	0	0	0	0	0	0	0	0
b7 b6 b5 b4 b3 b2 b1 b0								
Value after reset	-	-	-	-	-	-	-	-
	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:28]	TMDLC	TX Message Buffer DLC Field	No. of Data Bytes to be transmitted in a CAN Frame	R/W
b[27:0]	-	Reserved	These bits are read as 0. The write value should be always 0.	R/W

Each TX Message Buffer Pointer register is used to store the DLC fields of the message to transmit from the associated buffer.

4.5.19.1 CFDTMPTR.TMDLC

TX Message Buffer DLC Field

These bits select the number of Data Bytes that will be transmitted from this TX Message Buffer if corresponding **TMRTR** bit is configured as 1'b0.

Please refer to Table 5 in ISO 11898-1 (2015) Specification for details defining the number of Data Bytes that will be transmitted.

Users should not write to these bits when corresponding channel is in CH_SLEEP.

4.5.20 CFDTMFDCTR

TX Message Buffer CAN-FD Control Register

Address: Refer to Figure 4.2

	b31	b30	b29	b28	b27	b26	b25	b24	
TMPTR[15:8]									
Value after reset	0	0	0	0	0	0	0	0	
b23 b22 b21 b20 b19 b18 b17 b16									
Value after reset	0	0	0	0	0	0	0	0	
b15 b14 b13 b12 b11 b10 b9 b8									
Value after reset	-	-	-	-	-	-	TMIFL[1:0]		
	0	0	0	0	0	0	0	0	
b7 b6 b5 b4 b3 b2 b1 b0									
Value after reset	-	-	-	-	-	TMFDF	TMBRS	TMESI	
	0	0	0	0	0	0	0	0	
Bit	Symbol	Bit name	Function						R/W
b[31:16]	TMPTR	TX Message Buffer Pointer Field	TX Message Buffer Pointer						R/W
b[15:10]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[9:8]	TMIFL	TX Message Buffer Information Label Field	TX Message Buffer Information Label						R/W
b[7:3]	-	Reserved	These bits are read as 0. The write value should be always 0.						R/W
b[2]	TMFDF	CAN FD Format bit	1'b0: Non CAN-FD frame to transmit 1'b1: CAN-FD frame to transmit						R/W
b[1]	TMBRS	Bit Rate Switch bit	1'b0: CAN-FD frame to transmit with no bit rate switch 1'b1: CAN-FD frame to transmit with bit rate switch						R/W
b[0]	TMESI	Error State Indicator bit	1'b0: CAN-FD frame to transmit by error active node 1'b1: CAN-FD frame to transmit by error passive node						R/W

The TX Message Buffer CAN-FD Control register shows the status of the FDF, BRS and ESI bits and Pointer fields of the CAN-FD frame to be transmitted.

4.5.20.1 CFDTMFDCTR.TMESI

Error State Indicator bit

If the channel is not in error passive, then this bit equals the write value, else it is don't care and the bit is transmitted as 1'b1 on the CAN bus; indicating that this is an error passive node.

Users should not write to these bits when corresponding channel is in CH_SLEEP.

4.5.20.2 CFDTMFDCTR.TMBRS

Bit Rate Switch bit

Users should not write to these bits when corresponding channel is in CH_SLEEP.

4.5.20.3 CFDTMFDCTR.TMFDF

CAN FD Format bit

Users should not write to these bits when corresponding channel is in CH_SLEEP.

4.5.20.4 CFDTMFDCTR.TMIFL

TX Message Buffer Information label Field

The Information label value is stored in these bits. It will be copied, together with further message information, in the TX History List after successful transmission of the message.

Users should not write to these bits when corresponding channel is in CH_SLEEP.

4.5.20.5 CFDTMFDCTR.TMPTR

TX Message Buffer Pointer Field

The Pointer value is stored in these bits. It will be copied, together with further message information, in the TX History List after successful transmission of the message.

Users should not write to these bits when corresponding channel is in CH_SLEEP.

4.5.21 CFDTMDFp

TX Message Buffer Data Field p Registers

Address: Refer to Figure 4.2

	b31	b30	b29	b28	b27	b26	b25	b24
TMDB((p*q)+(q-1))[7:0]								
Value after reset	0	0	0	0	0	0	0	0
TMDB((p*q)+(q-2))[7:0]								
Value after reset	0	0	0	0	0	0	0	0
TMDB((p*q)+(q-3))[7:0]								
Value after reset	0	0	0	0	0	0	0	0
TMDB((p*q)+(q-4))[7:0]								
Value after reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit name	Function	R/W
b[31:24]	TMDB((p*q)+(q-1))	TX Message Buffer Data Byte ((p*q)+(q-1))	TX Message Buffer Data Byte ((p*q)+(q-1))	R/W
b[23:16]	TMDB((p*q)+(q-2))	TX Message Buffer Data Byte ((p*q)+(q-2))	TX Message Buffer Data Byte ((p*q)+(q-2))	R/W
b[15:8]	TMDB((p*q)+(q-3))	TX Message Buffer Data Byte ((p*q)+(q-3))	TX Message Buffer Data Byte ((p*q)+(q-3))	R/W
b[7:0]	TMDB((p*q)+(q-4))	TX Message Buffer Data Byte ((p*q)+(q-4))	TX Message Buffer Data Byte ((p*q)+(q-4))	R/W

(p = Data Field Register index = [0...no_of_CFDTMDFs-1])

(no_of_CFDTMDFs = No. of TX Message Buffer Data Field Registers)

no_of_CFDTMDFs = 16

(q = No. of Data Bytes = 4)

Each TX Message Buffer Data Field p register is used to store the Data Bytes DB((p*q)+(q-4)) to DB((p*q)+(q-1)) of the message to transmit from the associated buffer.

4.5.21.1 CFDTMDFp.TMDB((p*q)+(q-4))

TX Message Buffer Data Byte ((p*q)+(q-4))

Data Byte ((p*q)+(q-4)) of the Message stored in the TX Message Buffer.

Users should not write to these bits when corresponding channel is in CH_SLEEP.

4.5.21.2 CFDTMDFp.TMDB((p*q)+(q-3))

TX Message Buffer Data Byte ((p*q)+(q-3))

Data Byte ((p*q)+(q-3)) of the Message stored in the TX Message Buffer.

Users should not write to these bits when corresponding channel is in CH_SLEEP.

4.5.21.3 CFDTMDFp.TMDB((p*q)+(q-2))

TX Message Buffer Data Byte ((p*q)+(q-2))

Data Byte ((p*q)+(q-2)) of the Message stored in the TX Message Buffer.

Users should not write to these bits when corresponding channel is in CH_SLEEP.

4.5.21.4 CFDTMDFp.TMDB((p*q)+(q-1))

TX Message Buffer Data Byte ((p*q)+(q-1))

Data Byte ((p*q)+(q-1)) of the Message stored in the TX Message Buffer.

Users should not write to these bits when corresponding channel is in CH_SLEEP.

5 Modes of operation

5.1 Overview

The modes of the RS-CAN-FD module can be classified into 2 groups:

Global modes
Channel modes

5.2 Global Modes

These modes are applicable for the complete RS-CAN-FD module and hence are called “global” modes. The global modes of the RS-CAN-FD module are:

Global Sleep Mode
Global Reset Mode
Global Halt Mode
Global Operation Mode

Figure 5.1 shows the possible transitions between the global modes.

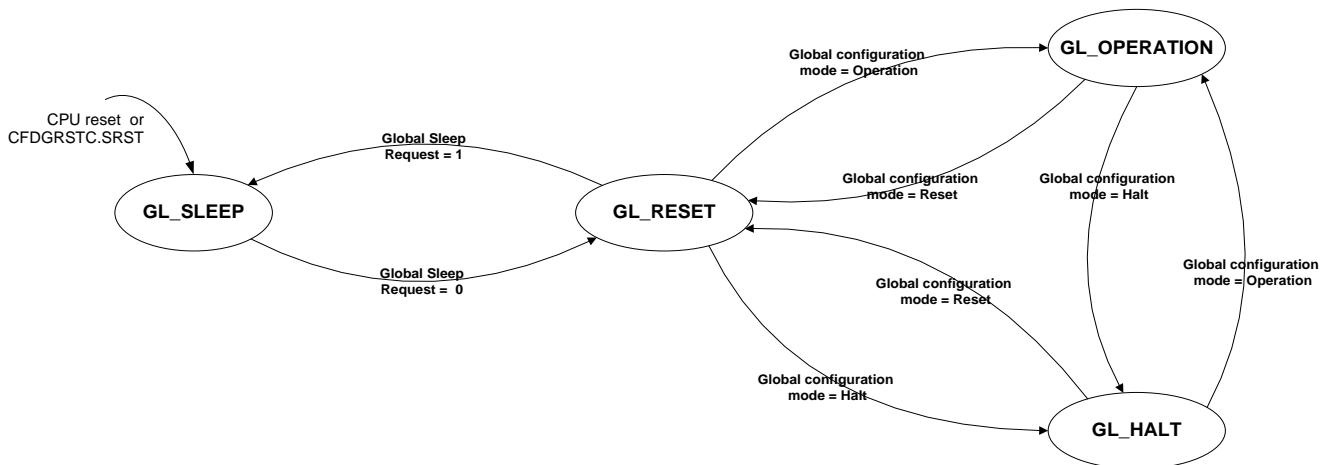


Figure 5.1 Transition between CAN Global Modes

Change in the Global mode can affect the Channel mode. Table 5.1 below shows the effect of a Global mode transition on the Channel mode.

target Global mode \ current Global mode	Sleep	Reset	Halt	Operation
Sleep	Ch-Sleep: keep Ch-Reset: N/A Ch-Halt: N/A Ch-Oper: N/A			
Reset	Ch-Sleep: keep Ch-Reset: -> Ch-Sleep Ch-Halt: N/A Ch-Oper: N/A		Ch-Sleep: keep Ch-Reset: keep Ch-Halt: N/A Ch-Oper: N/A	Ch-Sleep: keep Ch-Reset: keep Ch-Halt: N/A Ch-Oper: N/A
Halt		Ch-Sleep: keep Ch-Reset: keep Ch-Halt: -> Ch-Reset Ch-Oper: N/A		Ch-Sleep: keep Ch-Reset: keep Ch-Halt: keep Ch-Oper: N/A
Operation		Ch-Sleep: keep Ch-Reset: keep Ch-Halt: -> Ch-Reset Ch-Oper: -> Ch-Reset	Ch-Sleep: keep Ch-Reset: keep Ch-Halt: keep Ch-Oper: -> Ch-Halt	

Table 5.1 Possible CAN channel Modes versus Global Module Modes

5.2.1 Global Sleep Mode

After the release of the hardware reset or after setting and clearing a **CFDGRSTC.SRST** bit, the RS-CAN-FD module automatically enters Global Sleep Mode.

The RS-CAN-FD module will also enter this mode, when the Global Sleep Request bit is set while it is in Global Reset Mode.

This control bit cannot be set in Global Halt Mode or Global Operation Mode.

Setting the Global Sleep Request bit will set all Channel Sleep Request bits and force all channels into the Channel Sleep Mode.

Sleep Mode is used for power saving purpose. When RS-CAN-FD module is in Global Sleep Mode, only the clock for CPU write access to the Global Sleep Mode Request bit is active. All other clocks are stopped and all other functions of the RS-CAN-FD module are suspended.

Read access from all registers is still possible and all register values are preserved.

In Global Sleep mode RAM access is prohibition. Because, Logic which generates a RAM address does not operate.

After setting Global Sleep Request bit, it is necessary to confirm that the Global Sleep status has been updated indicating successful transition to Global Sleep Mode before the Global Sleep Request bit can be cleared again.

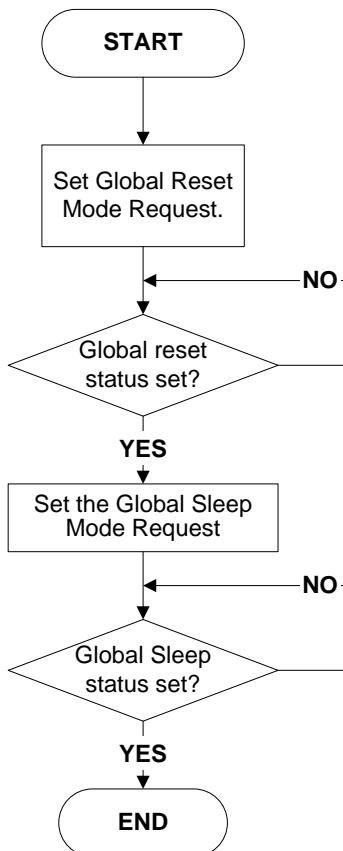


Figure 5.2 Procedure for entering global Sleep Mode

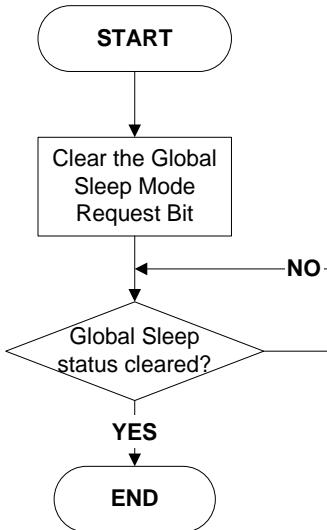


Figure 5.3 Procedure for exiting global Sleep Mode

5.2.2 Global Reset Mode

The RS-CAN-FD module enters this mode in the following ways:

Global Mode Control **CFDGCTR.GMDC** in the Global Control Register is configured for Global Reset Mode

while the RS-CAN-FD module is in Global Halt Mode or Global Operation Mode

Global Sleep Mode request bit is cleared while RS-CAN-FD module is in Global Sleep Mode

In Global Reset Mode, all RS-CAN-FD module functions are suspended and all status and flag registers are initialised.

Additionally all FIFOs and all channel TX Queues are disabled and transmission control bits are cleared.

Configuration registers (except the test mode registers) are not initialised in this mode to their MCU Reset values and the RS-CAN-FD module can be configured.

Refer to Section 5.4 for detailed description of the behaviour of all registers when transition to Global Reset Mode is performed.

Setting the global mode to Reset by setting the Global Mode Control bits **CFDGCTR.GMDC** in the Global Control Register to 2'b01 will set all Channel Mode Control bits **CFDCnCTR.CHMDC** in the Channel Control Registers to 2'b01 and force all channels into the Channel Reset Mode.

For channels that are already in Channel Reset Mode or Channel Sleep Mode this automatic transition is not performed (**CFDCnCTR.CHMDC** of related channel already set to 2'b01).

After setting Global Mode Control **CFDGCTR.GMDC** to Reset Mode it is necessary to confirm that the Reset Mode Status **CFDGSTS.GRSTSTS** in the Global Status Register has been updated indicating successful transition to Global Reset Mode before **CFDGCTR.GMDC** can be changed again.

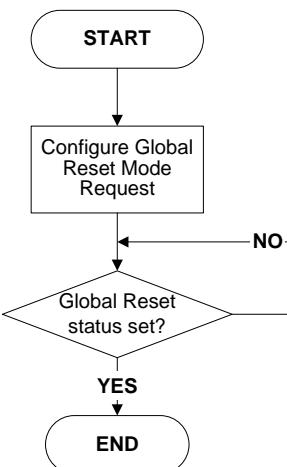


Figure 5.4 Procedure for entering global Reset Mode

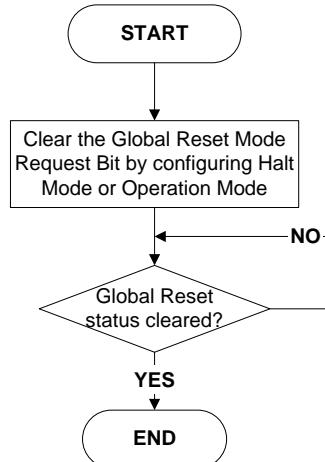


Figure 5.5 Procedure for exiting global Reset Mode

5.2.3 Global Halt Mode

The RS-CAN-FD module enters this mode in the following ways:

- Global Mode Control **CFDGCTR.GMDC** in the Global Control Register is configured for Global Halt Mode while the RS-CAN-FD module is in Global Reset Mode
 - the channels will be in either Channel Reset Mode or Channel Sleep Mode and will remain in this mode
 - Global Mode Control **CFDGCTR.GMDC** in the Global Control Register is configured for Global Halt Mode while the RS-CAN-FD module is in Global Operation Mode
 - all channels in Channel Reset Mode, Channel Halt Mode or Channel Sleep Mode will remain in this mode
 - all channels in Channel Operation Mode will transit to Channel Halt Mode
 - Global Halt Mode Status bit is set when all channels have left Channel Operation Mode
- If a transmission or reception is ongoing for a channel the transition to Channel Halt Mode is delayed until the completion of the communication.
- Similarly, if a channel is in Bus-Off, the full Bus-Off recovery sequence may be delayed depending on the channel configuration.

In the Global Halt Mode, all communications are suspended and RS-CAN-FD logic will not cause any change to status and flag registers (only when a channel is in the Bus-Off its REC and TEC values are cleared).

Also the test mode configuration and control registers are not initialised in this mode.

The Global Halt Mode should be used to configure global module test modes.

Refer to Section 5.4 for a detailed description of the behaviour of all registers when transition to Global Halt Mode is performed.

Setting the global mode to Halt by setting the Global Mode Control bits **CFDGCTR.GMDC** in the Global Control Register to 2'b10 will set all Channel Mode Control bits **CFDCnCTR.CHMDC** in the Channel Control Registers to 2'b10 for the channels that are in Channel Operation Mode and force these channels into the Channel Halt Mode.

For channels that are already in Channel Reset Mode, Channel Halt Mode or Channel Sleep Mode this automatic transition is not performed.

Therefore, the Global Halt Mode request can be used to shut down all CAN channel communications without loss of messages and disruption on the related CAN Bus (no interruption of reception/transmission processes on the channels).

After setting Global Mode Control **CFDGCTR.GMDC** to Halt Mode it is necessary to confirm that the Halt Mode status **CFDGSTS.GHLTSTS** in the Global Status Register has been updated indicating successful transition to Global Halt Mode. User should not do any other SFR setting until confirming **CFDGSTS.GHLTSTS** is set.

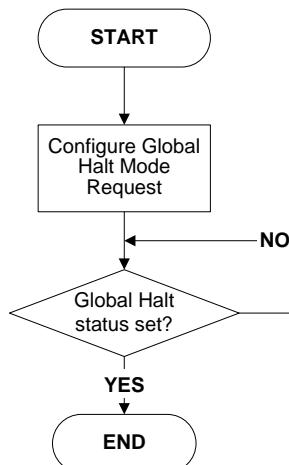


Figure 5.6 Procedure for entering global Halt Mode

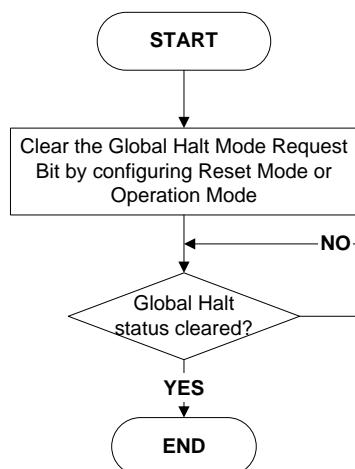


Figure 5.7 Procedure for exiting global Halt Mode

5.2.4 Global Operation Mode

The RS-CAN-FD module enters this mode when the Global Mode Configuration bits are set to Global Operation Mode.

CAN channels can only be set to Channel Operation Mode and start CAN communication when RS-CAN-FD is in Global Operation Mode.

After setting Global Mode Control **CFDGCTR.GMDC** to Global Operation Mode it is necessary to confirm that the Global Reset Mode status **CFDGSTS.GRSTSTS** and the Global Halt Mode status **CFDGSTS.GHLTSTS** in the Global Status Register have been cleared indicating successful transition to Global Operation Mode before **CFDGCTR.GMDC** can be changed again.

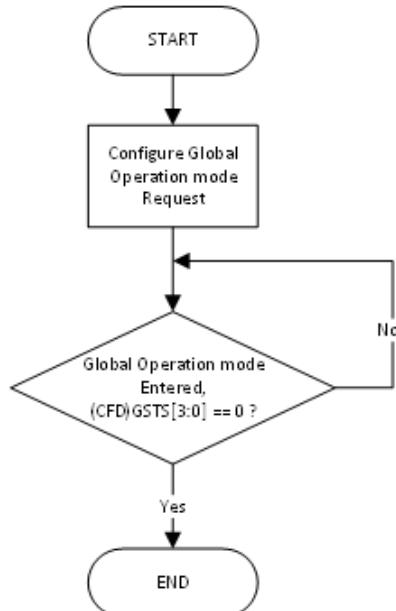


Figure 5.8 Procedure for entering global Operation Mode

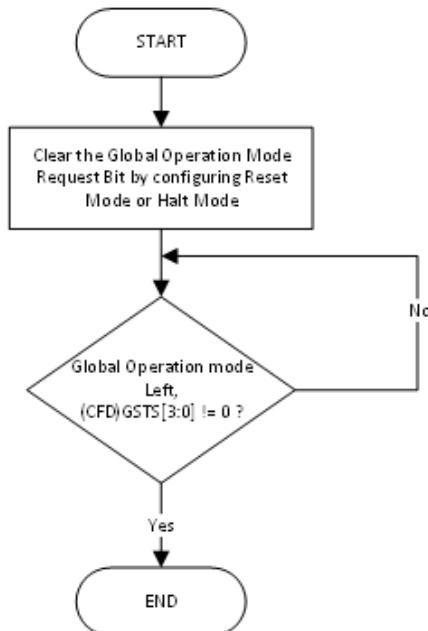


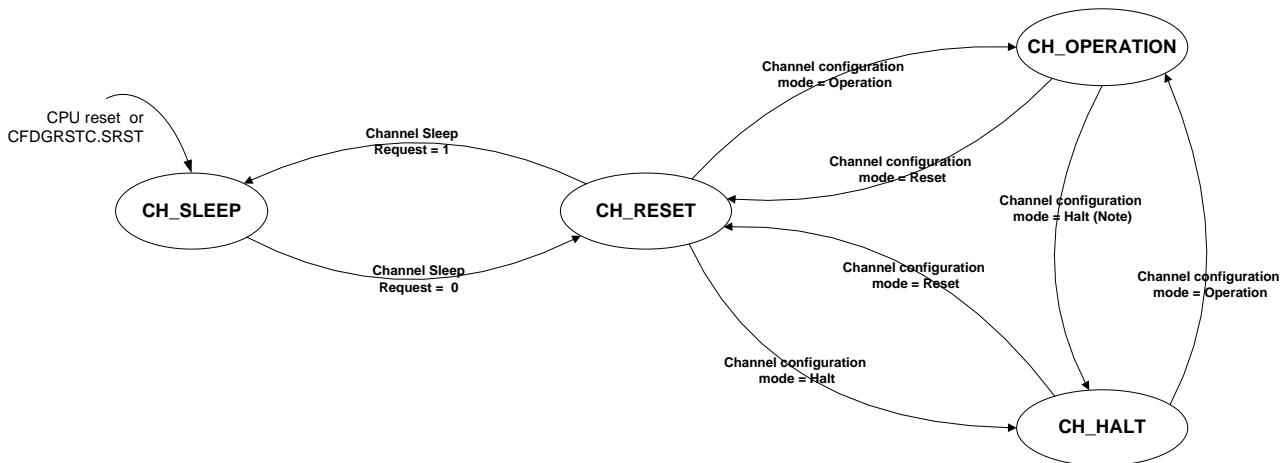
Figure 5.9 Procedure for exiting global Operation Mode

5.3 Channel Modes

Each CAN channel can be in one of the following four channel modes:

- Reset Mode
- Halt Mode
- Operation Mode
- Sleep Mode

Figure 5.10 shows the possible transitions between the channel modes.



Note: The transition timing depends on BOM setting in the Channel Configuration Register.
when BOM = 01b, the state transition timing is immediate after entering to BusOff state.
when BOM = 10b, the state transition timing is at the end of BusOff state.
when BOM = 11b, the state transition timing is matching with the setting of CHMDC to Halt mode.

Figure 5.10 Transition between CAN channel modes

5.3.1 CAN channel Sleep Mode

After the release of the hardware reset or after setting and clearing a **CFDGRSTC.SRST** bit, each CAN channel of the RS-CAN-FD module automatically enters Channel Sleep Mode.

Each CAN channel will also enter this mode, when the related Channel Sleep Mode Request bit is set while the CAN channel is in Channel Reset Mode.

This control bit should not be set in Channel Halt Mode or Channel Operation Mode.

Entering the CAN channel Sleep Mode instantly stops the clock supplied to the CAN channel unit and thereby reduces power consumption.

After setting Channel Sleep Mode request bit it is necessary to confirm that the Channel Sleep Mode Status has been updated indicating successful transition to Channel Sleep Mode before the Channel Sleep Mode request bit can be cleared again.

During Channel Sleep Mode, users cannot write to channel related registers. Read operation is still possible.

5.3.2 CAN channel Reset Mode

An RS-CAN-FD CAN channel enters this mode in the following ways:

Channel Mode Control **CFDCnCTR.CHMDC** in the Channel Control Registers is configured for Channel Reset Mode while the related CAN channel is in Channel Halt Mode or Channel Operation Mode
Channel Sleep Mode request bit is cleared while the related CAN channel is in Channel Sleep Mode
Global Mode Control **CFDGCTR.GMDC** is set to Global Reset Mode and CAN channel is not in Channel Sleep Mode or Channel Reset Mode

In Channel Reset Mode, all CAN channel status and flag registers are initialised.

Additionally all channel related transmission control bits are cleared and the channel related TX Queue is disabled.

Configuration registers (except the Channel Test Mode registers) are not initialised in this mode and the RS-

CAN-FD module CAN channel can be configured for communication.

Refer to Section 5.4 for detailed description of the behaviour of all registers when transition to Channel Reset Mode is performed.

After setting Channel Mode Control **CFDCnCTR.CHMDC** to Channel Reset Mode, it is necessary to confirm that the Reset Mode Status **CFDCnSTS.CRSTSTS** in the related Channel Status Registers has been updated indicating successful transition to Channel Reset Mode before the related **CFDCnCTR.CHMDC** can be changed again.

Refer to Table 5.2 below regarding the influence of transition to Channel Reset Mode while CAN communication is ongoing.

5.3.3 CAN channel Halt Mode

An RS-CAN-FD CAN channel enters this mode in the following ways:

Channel Mode Control **CFDCnCTR.CHMDC** in the Channel Control Registers is configured for Channel Halt Mode while the related CAN channel is in Channel Reset Mode or Channel Operation Mode

Global Mode Control **CFDGCTR.GMDC** is set to Global Halt Mode and CAN channel is in Channel Operation Mode

In Channel Halt Mode, all channel CAN communication is suspended but all status and flag registers remain unchanged during Channel Halt Mode entry (except for the Bus-Off case where REC and TEC values are cleared for this channel).

In addition, the channel test mode configuration and control registers are not initialised in this mode.

The Channel Halt Mode should be used to configure channel test modes.

Please refer to Section 5.4 for detailed description of the behaviour of all registers when transition to Channel Halt Mode is performed.

After setting Channel Mode Control **CFDCnCTR.CHMDC** to Channel Halt Mode it is necessary to confirm that the Halt Mode status **CFDCnSTS.CHLTSTS** in the related Channel Status Register has been updated indicating successful transition to Channel Halt Mode before the related **CFDCnCTR.CHMDC** can be changed again.

Refer to Table 5.2 regarding the influence of transition to Channel Halt Mode while CAN communication is ongoing.

State Mode	Receiver	Transmitter	Bus-Off
CAN channel Reset Mode (CFDCnCTR.CHMDC=01b)	The CAN channel transits to Channel Reset Mode without waiting for the completion of the ongoing reception. (Note 1)	The CAN channel transits to Channel Reset Mode without waiting for the completion of the ongoing transmission. (Note 1)	The CAN channel transits to Channel Reset Mode without waiting for the completion of the Bus-Off Recovery.
CAN channel Halt Mode (CFDCnCTR.CHMDC=10b)	CAN channel transits to Channel Halt Mode at the end of the ongoing reception or error (Note2).	CAN channel transits to Channel Halt Mode after completion of the ongoing transmission.	When CFDCnCTR.BOM is set to 00b, a Channel Halt Mode request will be accepted only after the completion of the full Bus-Off Recovery sequence. When CFDCnCTR.BOM is set to 10b, then, the CAN channel transits automatically to Channel Halt Mode after waiting for the completion of the Bus-Off Recovery. When CFDCnCTR.BOM is set to 01b, then, the CAN channel transits automatically to Channel Halt Mode without waiting for the completion of the Bus-Off Recovery. When CFDCnCTR.BOM is set to 11b, the CAN channel transits to Channel Halt Mode as soon as the Channel Halt Mode is requested (without waiting for the completion of the Bus-Off Recovery).

Table 5.2 Behaviour in CAN Reset / Halt Mode

Note 1: If the entry to Channel Reset Mode is required only at the end of an ongoing communication, then Channel Halt Mode can be requested first to prevent interruption of CAN communication by direct transition to Channel Reset Mode. After the CAN channel enters Channel Halt Mode the Channel Reset Mode can be requested.

Note 2: If CAN communication is locked at dominant level after an error flag, Application SW can detect this situation by monitoring the channel related BusLock Flag and resolve lock condition by setting the CAN channel to Channel Reset Mode.

5.3.4 CAN channel Operation Mode

The Channel Operation Mode is activated by setting the **CFDCnCTR.CHMDC** bits to 2'b00. If 11 consecutive recessive bits are detected after entering the CAN Operation Mode, then the **CFDCnSTS.COMSTS** bit is set and the CAN channel:

enables channel's communication functions allowing the channel to become an active node on the CAN network

releases the internal fault confinement logic including receive and transmit error counters

At this point, it can start transmission and reception of CAN messages.

Within the CAN channel Operation Mode, the channel may be in four different sub-modes, depending on which type of communication functions are performed (see Figure 5.11):

Channel idle: The CAN channel is neither receiving nor transmitting.

Channel receives: The channel is receiving a CAN message sent by another CAN node.

Channel transmits: The channel is transmitting a CAN message.

Note: The channel may receive its own message simultaneously when Self Test Mode is enabled.

Channel is in Bus-Off state: The CAN channel is cut-off from CAN Bus communication

After setting Channel Mode Control **CFDCnCTR.CHMDC** to Operation Mode, it is necessary to confirm that

the Channel Reset Mode Status **CFDCnSTS.CRSTSTS** and the Channel Halt Mode Status **CFDCnSTS.CHLTSTS** in the Channel Status Register have been updated indicating successful transition to Channel Operation Mode before the related **CFDCnCTR.CHMDC** can be changed again.

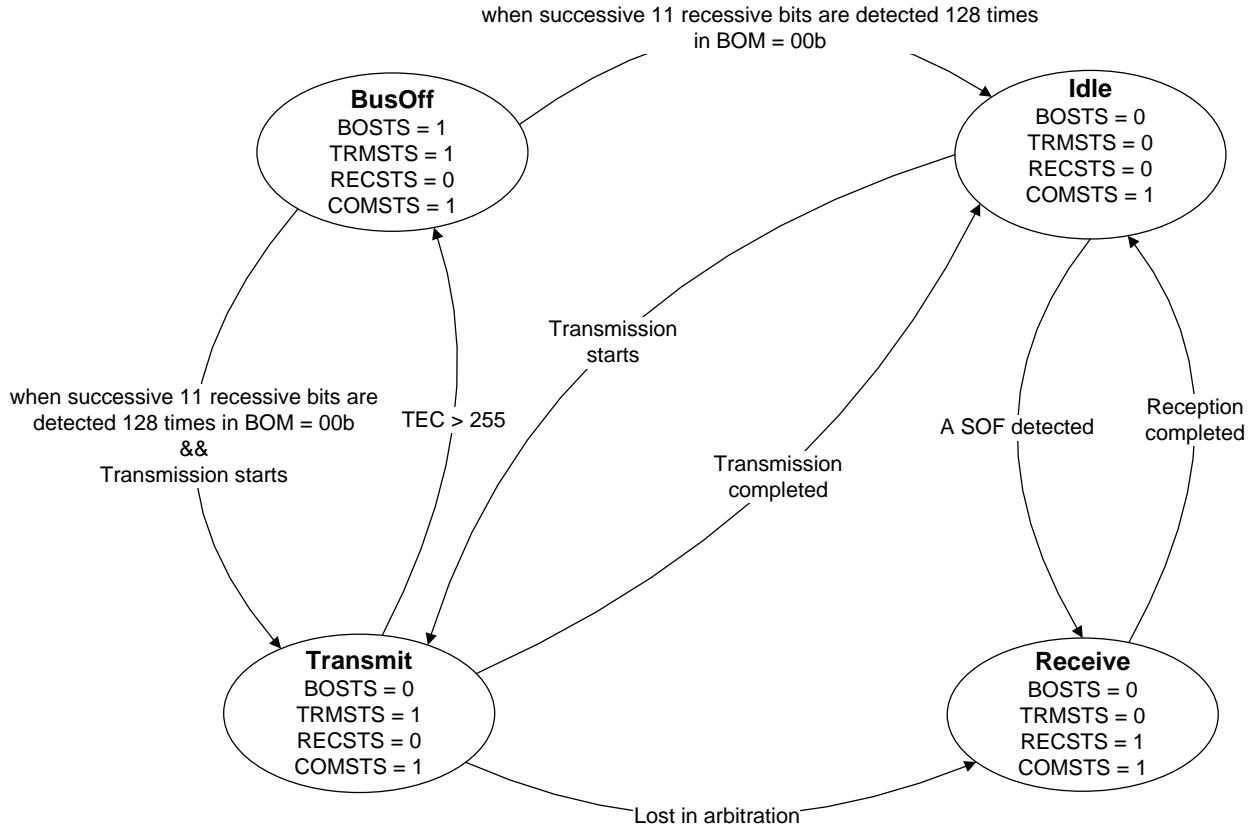


Figure 5.11 Sub-Modes of CAN channel Operation Mode (only BOM = 2'b00)

5.3.5 CAN channel Bus-Off State

The CAN channel Bus-Off state is entered according to the fault confinement rules of the CAN specification. Following modes for returning to the CAN channel Operation Mode from the Bus-Off state can be configured:

CFDCnCTR.BOM = 2'b00:

Bus-Off recovery is compliant to ISO 11898-1, namely the CAN channel re-enters CAN communication (Error Active state) after 11 consecutive recessive bits are detected 128 times. TEC and REC counters are initialised to zero. The Bus-Off Recovery Flag **CFDCnERFL.BORF** is set in this case.

CFDCnCTR.BOM = 2'b01:

The CAN channel changes the value of the **CFDCnCTR.CHMDC** bits within the CAN channel Control Register to 2'b10 and switches immediately to Channel Halt Mode automatically after entering the Bus-Off state. TEC and REC counters are initialised to zero. The Bus-Off Recovery Flag **CFDCnERFL.BORF** is not set in this case.

CFDCnCTR.BOM = 2'b10:

The CAN channel changes the value of the **CFDCnCTR.CHMDC** bits within the CAN channel Control Register to 2'b10 as soon as it reaches the Bus-Off state and enters Channel Halt Mode automatically after the CAN channel has completed the Bus-Off recovery sequence (i.e. after 11 consecutive recessive bits are detected 128 times). TEC and REC counters are initialised to zero. The Bus-Off Recovery Flag **CFDCnERFL.BORF** is set in this case.

CFDCnCTR.BOM = 2'b11:

Bus-Off recovery is initiated but CAN channel can enter immediately the Channel Halt Mode when still in Bus-Off state if a request is made to enter Channel Halt Mode.

TEC and REC counters are initialised to zero. In this case, the Bus-Off Recovery Flag **CFDCnERFL.BORF** is not set and TEC and REC counters are initialised to zero.

Without setting **CFDCnCTR.CHMDC [1:0]** =10B and when 11recessive bits is detected 128 times continuously, transition conditions become the same as **CFDCnCTR.BOM** = 2'b00.

Note, however, that if the recovery from Bus-Off occurs normally in this mode (i.e. after waiting for 128 sequences of 11 consecutive recessive bits), and no Halt request has been generated during this period, then the Bus-Off Recovery Flag **CFDCnERFL.BORF** is set.

In the case where Application SW writes into **CFDCnCTR.CHMDC** at the same time as the CAN channel is due to enter Halt Mode (at the start of Bus-Off when **CFDCnCTR.BOM** = 2'b01, or at the end of Bus-Off when **CFDCnCTR.BOM** = 2'b10) then the System SW request will have the highest priority.

Note that, in the above cases, the automatic setting of the **CFDCnCTR.CHMDC** to Channel Halt mode request is performed when the **CFDCnCTR.CHMDC** value is previously 2'b00 (Channel Operation Mode).

Additional it is possible to force the CAN channel to recover from the Bus-Off state by setting **CFDCnCTR.RTBO** to 1'b1. The error state changes from Bus-Off state to integrating state with a maximum delay of 1 CAN Bit time, and the CAN communication becomes possible again after 11 consecutive recessive bits are detected. The Bus-Off Recovery Flag is not set in this case.

TEC and REC counters are initialized to zero.

Before setting **CFDCnCTR.RTBO** to 1'b1, all pending transmissions from TX Message Buffers, TX Queues and/or Common FIFO in TX or GW Mode should be disabled.

The disable of the pending transmission Message Buffer, TX Queue or FIFO must be confirmed by the corresponding acknowledge flags.

For TX Message Buffer these are the Transmission Result Flags (**CFDTMSTSj.TMTRF**) for the TX Queue it is the TX Queue empty flag (**CFDTXQSTS_n.TXQEMP**) and for the FIFO it is the FIFO empty flag (**CFDCFSTS_d.CFEMP**).

The **CFDCnCTR.RTBO** bit should be used for Bus-Off recovery only when **CFDCnCTR.BOM** is set to 2'b00.

Setting this bit in any state (other than Bus-Off) will have no effect and the bit will be cleared immediately.

Table 5.3 summarises the setting of Bus-Off Entry Flag **CFDCnERFL.BOEF** and Bus-Off Recovery Flag **CFDCnERFL.BORF** for different configurations of **CFDCnCTR.BOM**.

BOM	BOEF bit set	BORF bit set
2'b00	Always (on entry to Bus-Off)	Always (on exit from Bus-Off)
2'b00 CFDCnCTR.RTBO set to '1'	Always (on entry to Bus-Off)	only if normal Bus-Off recovery occurs before System SW sets CFDCnCTR.RTBO to '1'
2'b01	Always (on entry to Bus-Off)	Never
2'b10	Always (on entry to Bus-Off)	Always (on exit from Bus-Off)
2'b11	Always (on entry to Bus-Off)	only if normal Bus-Off recovery occurs before System SW issues Halt request

Table 5.3 Bus-Off entry / recovery flag behaviour

To make an efficient SW procedure it is not mandatory to wait for BusOff recovery sequence end.

It is possible to do the transmission re-initialisation during the BusOff recovery. To do this following SW-flow is recommended to use, refer to Figure 5.12.

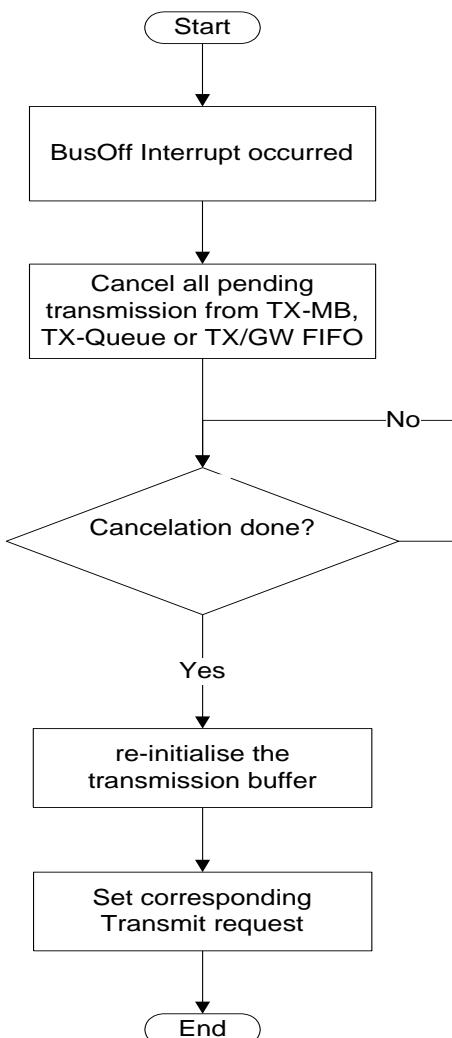


Figure 5.12 Transmission re-initialisation during Bus-Off

5.4 Global Mode – Channel Mode transition interactions

In the following, the interaction between global mode setting and channel mode setting is summarized:

Changing of Channel Mode Control **CFDCnCTR.CHMDC** in the Channel Control Registers does not have any influence on the Global Mode Control **CFDGCTR.GMDC**.

Changing of Global Mode Control **CFDGCTR.GMDC** is influencing the channel mode control in the way described in Table 5.4.

Global Mode Change	Channel Mode	Channel Mode Transition Action
Sleep → Reset	Sleep	channel remains in Sleep Mode
Sleep → Halt	- (global mode change not possible)	
Sleep → Operation	- (global mode change not possible)	
Reset → Sleep	Sleep	channels remains in Sleep Mode
	Reset	channel Sleep request bit is set automatically, channel transits to Sleep Mode
Reset → Halt	Sleep	channel remains in Sleep Mode
	Reset	channel remains in Reset Mode
Reset → Operation	Sleep	channel remains in Sleep Mode
	Reset	channel remains in Reset Mode
Halt → Sleep	- (global mode change not possible)	
Halt → Reset	Sleep	channel remains in Sleep Mode
	Reset	channel remains in Reset Mode
	Halt	channel mode control is set to Reset Mode, channel transits to Reset Mode
Halt → Operation	Sleep	channel remains in Sleep Mode
	Reset	channel remains in Reset Mode
	Halt	channel remains in Halt Mode
Operation → Sleep	- (global mode change not possible)	
Operation → Reset	Sleep	channel remains in Sleep Mode
	Reset	channel remains in Reset Mode
	Halt	channel mode control is set to Reset Mode, channel transits to Reset Mode
	Operation	channel mode control is set to Reset Mode, channel transits to Reset Mode
Operation → Halt	Sleep	channel remains in Sleep Mode
	Reset	channel remains in Reset Mode
	Halt	channel remains in Halt Mode
	Operation	channel mode control is set to Halt Mode, channel transits to Halt Mode after communication finished

Table 5.4 Global – Channel mode transition interaction

5.4.1 Global Mode change timing

The transition time for the Global mode changes are shown below.

From	To	max. transition time
GL_SLEEP	GL_RESET	3 peripheral clock cycle (Note 2)
GL_RESET	GL_SLEEP	3 peripheral clock cycle
GL_RESET	GL_HALT	10 peripheral clock cycle
GL_RESET	GL_OPERATION	10 peripheral clock cycle
GL_HALT	GL_RESET	2 CAN bit times (<i>1TQ + 16 peripheral clock cycle + 2 DLL clock cycle</i>)
GL_HALT	GL_OPERATION	3 peripheral clock cycle
GL_OPERATION	GL_RESET	2 CAN bit times (<i>1TQ + 16 peripheral clock cycle + 2 DLL clock cycle</i>)
GL_OPERATION	GL_HALT	3 CAN frames (<i>1 CAN Frame + 3424 peripheral clock cycle</i>) (Note 1,3)

Note 1: The given transition time is the time without any errors on the bus. In case of Error condition the transition time could lengthen to an uncalculated result. As well it could also come to stuck condition in case of locked RX lines or continues error conditions.

Note 2: Users should leave GL_SLEEP mode, only when **CFDGSTS.GRAMINIT** is cleared.

Note 3: TQ, CAN frame and CAN bits are related to the individual channels, for the max transition time the channel with the lowest Baud Rate has to be used.

5.4.2 Channel Mode change timing

The transition time for the Channel mode changes are shown below.

From	To	max. transition time
CH_SLEEP	CH_RESET	3 peripheral clock cycle
CH_RESET	CH_SLEEP	3 peripheral clock cycle
CH_RESET	CH_HALT	3 CAN bit times (<i>1 CAN Bits + 2TQ + 8 peripheral clock cycle + 2 DLL clock cycle</i>)
CH_RESET	CH_OPERATION	4 CAN bit times (<i>2 CAN Bit + 1 TSEG1 + 12 peripheral clock cycle + 2 DLL clock cycle</i>)
CH_HALT	CH_RESET	2 CAN bit times (<i>1TQ + 10 peripheral clock cycle + 2 DLL clock cycle</i>)
CH_HALT	CH_OPERATION	4 CAN bit times (<i>< 4 CAN bits</i>) (Note 3)
CH_OPERATION	CH_RESET	2 CAN bit times (<i>1TQ + 10 peripheral clock cycle + 2 DLL clock cycle</i>)
CH_OPERATION	CH_HALT	2 CAN frames (<i>1 CAN Frame + 13 CAN Bit</i>) (Note 1,2)

Note 1: The time specified for this transition does not include the case where channel enters Bus-Off state. In case of Bus-Off the timing depends upon the configuration of the **CFDCnCTR.BOM[1:0]** bits.

Note 2: The given transition time is the time without any errors on the bus. In case of Error condition the transition time could lengthen to an uncalculated result. As well it could also come to a stuck condition in case of locked RX lines or continues error conditions.

Note 3: In general if the Baudrate prescaler value **CFDCnNCFG.NBRP** is changed in CH_HALT then the transition time could be deviate from above. As the internal prescaler is a free running down counter to create the TQ clock. And new BRP value will earliest be captured when the counter reached the value zero.

5.4.3 Register behaviour in global/channel Modes

The following table shows the bit behaviour when RS-CAN-FD module changes state.

	Module	Register name	Bit	Symbol	Description	MCU_RESET	Software-RESET	G_CAN_Sleep	G_CAN_RESET	G_CAN_HALT	CH_CAN_Sleep	CH_CAN_RESET	CH_CAN_HALT
		1. Channel Registers											
CFDCnCFG	m_chsfr	Channel n Configuration Register											
			[9:0]	NBRP	Nominal Baudrate Prescaler	10'h00	10'h00	unch	unch	unch	unch	unch	unch
			[16:10]	NSJW	Nominal Synchronization Jump Width	7'b00000000	7'b00000000	unch	unch	unch	unch	unch	unch
			[24:17]	NTSEG1	Nominal Time Segment 1	8'b00000000	8'b00000000	unch	unch	unch	unch	unch	unch
			[31:25]	NTSEG2	Nominal Time Segment 2	7'b00000000	7'b00000000	unch	unch	unch	unch	unch	unch
CFDCnDCF	m_chsfr	Channel n Configuration Register											
			[7:0]	DBRP	Data Baudrate Prescaler	8'h00	8'h00	unch	unch	unch	unch	unch	unch
			[12:8]	DTSEG1	Data Time Segment 1	5'b00000	5'b00000	unch	unch	unch	unch	unch	unch
			[15:13]	-	reserved	-	-	-	-	-	-	-	-
			[19:16]	DTSEG2	Data Time Segment 2	4'b0000	4'b0000	unch	unch	unch	unch	unch	unch
			[23:20]	-	reserved	-	-	-	-	-	-	-	-
			[27:24]	DSJW	Data Synchronization Jump Width	4'b0000	4'b0000	unch	unch	unch	unch	unch	unch
			[31:28]	-	reserved	-	-	-	-	-	-	-	-
CFDCnCTR	m_chsfr	Channel n Control Register											
			[1:0]	CHMDC	Channel Mode Control	2'b01	2'b01	unch	2'b01	unch if channel in Sleep, Reset, Halt, otherwise bit will be set to 2'b10	unch	2'b01	2'b10
			2	CSLPR	Channel Sleep Request	1	1	1	unch	unch	unch	unch	unch
			3	RTBO	Return from Bus-Off	0	0	unch	unch	unch	unch	unch	unch
			[7:4]	-	reserved	-	-	-	-	-	-	-	-
			8	BEIE	Bus Error Interrupt Enable	0	0	unch	unch	unch	unch	unch	unch
			9	EWIE	Error Warning Interrupt Enable	0	0	unch	unch	unch	unch	unch	unch

			10	EPIE	Error Passive Interrupt Enable	0	0	unch	unch	unch	unch	unch	unch
			11	BOEIE	Bus-Off Entry Interrupt Enable	0	0	unch	unch	unch	unch	unch	unch
			12	BORIE	Bus-Off Recovery Interrupt Enable	0	0	unch	unch	unch	unch	unch	unch
			13	OLIE	Overload Interrupt Enable	0	0	unch	unch	unch	unch	unch	unch
			14	BLIE	Bus Lock Interrupt Enable	0	0	unch	unch	unch	unch	unch	unch
			15	ALIE	Arbitration Lost Interrupt Enable	0	0	unch	unch	unch	unch	unch	unch
			16	TAIE	Transmission Abortion Interrupt Enable	0	0	unch	unch	unch	unch	unch	unch
			17	EOCOIE	Error Occurrence Counter Overflow Interrupt Enable	0	0	unch	unch	unch	unch	unch	unch
			18	SOCOIE	Successful Occurrence Counter Overflow Interrupt Enable	0	0	unch	unch	unch	unch	unch	unch
			19	TDCVFIE	Transceiver Delay compensation Bit FIFO Msg Lost Interrupt enable	0	0	unch	unch	unch	unch	unch	unch
			20	-	reserved	-	-	-	-	-	-	-	-
			[22: 21]	BOM	Bus-Off Mode	2'b00	2'b00	unch	unch	unch	unch	unch	unch
			23	ERRD	Error Display	0	0	unch	unch	unch	unch	unch	unch
			24	CTME	Channel Test Mode Enable	0	0	unch	0	unch	unch	0	unch
			[26: 25]	CTMS	Channel Test Mode Select	2'b00	2'b00	unch	2'b00	unch	unch	2'b00	unch
			27	TRWE	TEC/REC Write Enable	0	0	unch	0	unch	unch	0	unch
			28	TRH	TEC/REC Hold	0	0	unch	0	unch	unch	0	unch
			29	TRR	TEC/REC Reset	0	0	unch	0	unch	unch	0	unch
			30	CRCT	CRC Test mode	0	0	unch	0	unch	unch	0	unch
			31	ROM	Restricted Operation Mode	0	0	unch	0	unch	unch	0	unch
CFDCnSTS	m_chsfr	Channel n Status Register											
			0	CRSTSTS	Channel RESET State	1	1	unch	1	unch	unch	1	0
			1	CHLTSTS	Channel HALT State	0	0	unch	0	unch if channel in Sleep, Reset , Halt, otherwise bit will be set to 1	unch	0	1
			2	CSLPSTS	Channel SLEEP State	1	1	1	unch	unch	1	unch	unch

			3	EPSTS	Error Passive Status	0	0	unch	0	unch	unch	0	unch
			4	BOSTS	Bus-Off Status	0	0	unch	0	0	unch	0	0
			5	TRMSTS	Transmit Status	0	0	unch	0	0	unch	0	0
			6	RECSTS	Receive Status	0	0	unch	0	0	unch	0	0
			7	COMSTS	Communication Status	0	0	unch	0	0	unch	0	0
			8	ESIF	Error State Indication Flag	0	0	unch	0	unch	unch	0	unch
			[15:9]	-	reserved	-	-	-	-	-	-	-	-
			[23:16]	REC	Reception Error Count	8'h00	8'h00	unch	8'h00	unch	unch	8'h00	unch
			[31:24]	TEC	Transmission Error Count	8'h00	8'h00	unch	8'h00	unch	unch	8'h00	unch
CFDCnERFL	m_chsfr	Channel n Error Flag Register											
			0	BEF	Bus Error Flag	0	0	unch	0	unch	unch	0	unch
			1	EWF	Error Warning Flag	0	0	unch	0	unch	unch	0	unch
			2	EPF	Error Passive Flag	0	0	unch	0	unch	unch	0	unch
			3	BOEF	Bus-Off Entry Flag	0	0	unch	0	unch	unch	0	unch
			4	BORF	Bus-Off Recovery Flag	0	0	unch	0	unch	unch	0	unch
			5	OVLF	Overload Flag	0	0	unch	0	unch	unch	0	unch
			6	BLF	Bus Lock Flag	0	0	unch	0	unch	unch	0	unch
			7	ALF	Arbitration Lost Flag	0	0	unch	0	unch	unch	0	unch
			8	SERR	Stuff Error	0	0	unch	0	unch	unch	0	unch
			9	FERR	Form Error	0	0	unch	0	unch	unch	0	unch
			10	AERR	Ack Error	0	0	unch	0	unch	unch	0	unch
			11	CERR	CRC Error	0	0	unch	0	unch	unch	0	unch
			12	B1ERR	Bit 1 Error	0	0	unch	0	unch	unch	0	unch
			13	B0ERR	Bit 0 Error	0	0	unch	0	unch	unch	0	unch
			14	ADERR	Ack Del Error	0	0	unch	0	unch	unch	0	unch
			15	-	reserved	-	-	-	-	-	-	-	-
			[30:16]	CRCREG	CRC register value	0	0	unch	0	unch	unch	0	unch
			31	-	reserved	-	-	-	-	-	-	-	-
CFDCnFDCFG	m_chsfr	Channel n CAN-FD Configuration Register	[2:0]	EOCCFG	Error Occurrence Counter Configuration	3'b000	3'b000	unch	unch	unch	unch	unch	unch
			[7:3]	-	reserved	-	-	-	-	-	-	-	-
			8	TDCOC	Transceiver Delay Compensation Offset Configuration	0	0	unch	unch	unch	unch	unch	unch
			9	TDCE	Transceiver Delay Compensation Enable	0	0	unch	unch	unch	unch	unch	unch
			10	ESIC	Error State Indication	0	0	unch	unch	unch	unch	unch	unch



					Configurati on									
		[11]	-	reserved	-	-	-	-	-	-	-	-	-	-
		[13: 12]	RPNMD	Pretended Network Filter Mode	0	0	unch							
		[15: 14]	-	reserved	-	-	-	-	-	-	-	-	-	-
		[23: 16]	TDCO	Transceiver Delay Compens ation Offset	0	0	unch							
		24	GWEN	CAN2.0 -> CAN FD GW Enable	0	0	unch							
		25	GWDF	GW FDF Configurati on	0	0	unch							
		26	GWBR	GW BRS Configurati on	0	0	unch							
		27	-	reserved	-	-	-	-	-	-	-	-	-	-
		28	FDOE	FD only enable	0	0	unch							
		29	REFE	RX edge filter enable	0	0	unch							
		30	CLOE	Classical CAN only enable	0	0	unch							
		31	CFDTE	CAN-FD Tolerance enable	0	0	unch							
CFDCnFDC TR	m_chsfr	Channel n CAN- FD Control Register	0	EOCCLR	Error Occurrenc e Counter Clear	0	0	unch	0	unch	unch	0	unch	
		1	SOCCLR	Successfu l Occurrenc e Counter Clear	0	0	unch	0	unch	unch	unch	0	unch	
		[15: 2]	-	reserved	-	-	-	-	-	-	-	-	-	-
		[17: 16]	PNMDC	Pretended Network Filter Mode Control	0	0	unch	0	unch	unch	0	unch		
		[23: 18]	-	reserved	-	-	-	-	-	-	-	-	-	-
		[31: 24]	KEY	Key code	-	-	-	-	-	-	-	-	-	-
CFDCnFDS TS	m_chsfr	Channel n CAN- FD Status Register	[7:0]	TDCR	Transceiver Delay Compens ation Result	0	0	unch	0	unch	unch	0	unch	
		8	EOCO	Error occurrence counter overflow	0	0	unch	0	unch	unch	0	unch		
		9	SOCO	Successfu l occurrenc e counter overflow	0	0	unch	0	unch	unch	0	unch		
		[11: 10]	-	reserved	-	-	-	-	-	-	-	-	-	-
		[13: 12]	PNSTS	Pretended Network Filter State	0	0	unch	0	unch	unch	0	unch		
		14	-	reserved	-	-	-	-	-	-	-	-	-	-
		15	TDCVF	Transceiver Delay compensa tion violation flag	0	0	unch	0	unch	unch	0	unch		
		[23: 16]	EOC	Error occurrence counter register	0	0	unch	0	unch	unch	0	unch		

			[31: 24]	SOC	Successfu l occurrenc e counter register	0	0	unch	0	unch	unch	0	unch
CFDCnFDC RC	m_chsfr	Channe l n CAN- FD CRC Register	[20: 0]	CRCREG	CRC Register value	0	0	unch	0	unch	unch	0	unch
			[23: 21]	-	reserved	-	-	-	-	-	-	-	-
			[27: 24]	SCNT	Stuff bit count	0	0	unch	0	unch	unch	0	unch
			[31: 28]	-	reserved	-	-	-	-	-	-	-	-
2. Global IP Registers													
CFDGIPV	m_comsf r	Global IP Version Register											
			[7:0]	IPV	IP Version Release number	8'h43	8'h43	8'h43	8'h43	8'h43	-	-	-
			[9:8]	IPT	IP Type	2'b01	2'b01	2'b01	2'b01	2'b01	-	-	-
			[15: 10]	-	reserved	-	-	-	-	-	-	-	-
			[29: 16]	PSI	Parameter Status Informatio n	14'h0000	14'h0000	14'h0 000	14'h0000	14'h0 000	-	-	-
			[31: 30]	-	reserved	-	-	-	-	-	-	-	-
3. Global Configuration / Control Registers													
CFDGCFG	m_comsf r	Global Configur ation Register											
			0	TPRI	Transmiss ion Priority	0	0	unch	unch	unch	-	-	-
			1	DCE	DLC Check Enable	0	0	unch	unch	unch	-	-	-
			2	DRE	DLC Replacem ent Enable	0	0	unch	unch	unch	-	-	-
			3	MME	Mirror Mode Enable	0	0	unch	unch	unch	-	-	-
			4	DCS	PLL By- Pass	0	0	unch	unch	unch	-	-	-
			5	CMPOC	CAN-FD message Payload overflow configurati on	0	0	unch	unch	unch	-	-	-
			[7:6]	-	reserved	-	-	-	-	-	-	-	-
			[11: 8]	TSP	Timestamp Prescaler	4'b0000	4'b0000	unch	unch	unch	-	-	-
			12	TSSS	Timestamp Source Select	0	0	unch	unch	unch	-	-	-
			[15: 13]	TSBTCS	Timestamp Bit Time Channel Select	3'b000	3'b000	unch	unch	unch	-	-	-
			[31: 16]	ITRPCP	Interval Timer Reference Clock Prescaler	16'h0000	16'h0000	unch	unch	unch	-	-	-
CFDGCTR	m_comsf r	Global Control Register											
			[1:0]	GMDC	Global Mode Control	2'b01	2'b01	unch	unch	unch	-	-	-
			2	GSLPR	Global Sleep Request	1	1	unch	unch	unch	unch	unch	unch
			[7:3]	-	reserved	-	-	-	-	-	-	-	-
			8	DEIE	DLC Error Interrupt Enable	0	0	unch	unch	unch	-	-	-

			9	MEIE	Message Lost Error Interrupt Enable	0	0	unch	unch	unch	-	-	-
			10	THLEIE	TX History List Entry Lost Interrupt Enable	0	0	unch	unch	unch	-	-	-
			11	CMPOFIE	CAN-FD message payload overflow Flag Interrupt enable	0	0	unch	unch	unch	-	-	-
			12	QOWEIE	TXQ Message overwrite Error Interrupt Enable	0	0	unch	unch	unch	-	-	-
			13	-	reserved	-	-	-	-	-	-	-	-
			14	QMEIE	TXQ Message lost Error Interrupt Enable	0	0	unch	unch	unch	-	-	-
			15	MOWEIE	Message overwrite Error Interrupt Enable	0	0	unch	unch	unch	-	-	-
			16	TSRST	TS Reset	0	0	unch	unch	unch	-	-	-
			17	TSWR	<i>Timestamp p Write</i>	0	0	unch	0	unch	-	-	-
			[31: 18]	-	reserved	-	-	-	-	-	-	-	-
<hr/>													
CFDGFDG G	m_comsf r	Global FD configuration register											
		0	RPED	Protocol exception state disable	0	0	unch	unch	unch	-	-	-	-
		[7:1]	-	reserved	-	-	-	-	-	-	-	-	-
		[9:8]	TSCCFG	Time stamp capture configuration	0	0	unch	unch	unch	-	-	-	-
		[31: 10]	-	reserved	-	-	-	-	-	-	-	-	-
<hr/>													
CFDGRC CFG	m_comsf r	Global CRC configuration register											
		0	NIE	<i>Non ISO enable</i>	0	0	unch	unch	unch	-	-	-	-
		[31: 1]	-	reserved	-	-	-	-	-	-	-	-	-
4. Global Status Registers													
CFDGSTS	m_comsf r	Global Status Register											
		0	GRSTSTS	Global RESET Status	1	1	unch	1	0	-	-	-	-
		1	GHLTSTS	Global HALT Status	0	0	unch	0	1	-	-	-	-
		2	GSLPSTS	Global SLEEP Status	1	1	1	unch	unch	-	-	-	-
		3	GRAMINIT	Global RAM Initialisation Status	1	1	1->0	0	0	-	-	-	-
		[31: 4]	-	reserved	-	-	-	-	-	-	-	-	-
CFDGERFL	m_comsf r	Global Error											

		Flag Register											
		0	DEF	DLC Error Flag	0	0	unch	0	unch	-	-	-	-
		1	MES	Message Lost Error Status	0	0	unch	0	unch	-	-	-	-
		2	THLES	TX History List Entry Lost Error Status	0	0	unch	0	unch	-	-	-	-
		3	CMPOF	CAN-FD message payload overflow Flag	0	0	unch	0	unch	-	-	-	-
		4	QOWES	TXQ Message overwrite Error Status	0	0	unch	0	unch	-	-	-	-
		5	OTBMLTSTS	OTB FIFO Msg Lost Status	0	0	unch	0	unch	-	-	-	-
		6	QMES	TXQ Message Lost Error Status	0	0	unch	0	unch	-	-	-	-
		7	MOWES	Message overwrite Error Status	0	0	unch	0	unch	-	-	-	-
		[15: 8]	RXSFAILn	RX SCAN Fail Channel n	0	0	unch	0	unch	-	-	-	-
		[23: 16]	EEFn	ECC Error Flag for Channel n	0	0	unch	0	unch	-	-	-	-
		[31: 24]	-	reserved	-	-	-	-	-	-	-	-	-
CFDGTS	m_timestamp	Global Timestamp Counter Register											
		[15: 0]	TS	Timestamp Value	16'h0000	16'h0000	unch	16'h0000	unch	-	-	-	-
		[31: 16]	-	reserved	-	-	-	-	-	-	-	-	-
5. Global Acceptance Filter List Configuration Registers													
CFDGAFLE CTR	m_acsfr	Global Acceptance Filter List Entry Control Register											
		[6:0]	AFLPN	Acceptance Filter List Page Number	7'b00000000	7'b00000000	unch	unch	unch	-	-	-	-
		7	-	reserved	-	-	-	-	-	-	-	-	-
		8	AFLDAE	Acceptance Filter List Access Enable Data	0	0	unch	unch	unch	-	-	-	-
		[31: 9]	-	reserved	-	-	-	-	-	-	-	-	-
CFDGAFLC FG0	m_acsfr	Global Acceptance Filter List Configuration 0 Register											
		[8:0]	RNC1	Rule Number for Channel 1	9'h000	9'h000	unch	unch	unch	-	-	-	-
		[15: 9]	-	reserved	-	-	-	-	-	-	-	-	-
		[24: 16]	RNC0	Rule Number for Channel 0	9'h000	9'h000	unch	unch	unch	-	-	-	-

			[31: 25]	-	reserved	-	-	-	-	-	-	-	-	-
CFDGAFLC FG1	m_acsfr	Global Acceptance Filter List Configuration 1 Register												
			[8:0]	RNC3	Rule Number for Channel 3	9'h000	9'h000	unch	unch	unch	-	-	-	-
			[15: 9]	-	reserved	-	-	-	-	-	-	-	-	-
			[24: 16]	RNC2	Rule Number for Channel 2	9'h000	9'h000	unch	unch	unch	-	-	-	-
			[31: 25]	-	reserved	-	-	-	-	-	-	-	-	-
CFDGAFLC FG2	m_acsfr	Global Acceptance Filter List Configuration 2 Register												
			[8:0]	RNC5	Rule Number for Channel 5	9'h000	9'h000	unch	unch	unch	-	-	-	-
			[15: 9]	-	reserved	-	-	-	-	-	-	-	-	-
			[24: 16]	RNC4	Rule Number for Channel 4	9'h000	9'h000	unch	unch	unch	-	-	-	-
			[31: 25]	-	reserved	-	-	-	-	-	-	-	-	-
CFDGAFLC FG3	m_acsfr	Global Acceptance Filter List Configuration 3 Register												
			[8:0]	RNC7	Rule Number for Channel 7	9'h000	9'h000	unch	unch	unch	-	-	-	-
			[15: 9]	-	reserved	-	-	-	-	-	-	-	-	-
			[24: 16]	RNC6	Rule Number for Channel 6	9'h000	9'h000	unch	unch	unch	-	-	-	-
			[31: 25]	-	reserved	-	-	-	-	-	-	-	-	-
6. RX Mailbox Registers														
CFDRMNB	m_acsfr	RX Mailbox Number Register												
			[7:0]	NRXMB	Number of RX MB	8'h00	8'h00	unch	unch	unch	-	-	-	-
			[10: 8]	RMPLS	Reception Message Buffer Payload Data Size	3'b000	3'b000	unch	unch	unch	-	-	-	-
			[31: 11]	-	reserved	-	-	-	-	-	-	-	-	-
CFDRMNDt	m_acsfr	RX-Mailbox NewData Registers												
			[31: 0]	RMNS	RX Mailboxes Newdata Flag	32'h0000 0000	32'h0000 0000	unch	32'h0000 0000	unch	-	-	-	-

7. RX FIFO Registers											
CFDRFCCa	m_rxifou nit	RX FIFO Configur ation / Control Register s [7:0]									
		0	RFE	RX FIFO Enable	0	0	unch	0	unch	-	-
		1	RFIE	RX FIFO Interrupt Enable	0	0	unch	unch	unch	-	-
		[3:2]	-	reserved	-	-	-	-	-	-	-
		[6:4]	RFPLS	Rx FIFO Payload Data Size configuration	3'b000	3'b000	unch	unch	unch	-	-
		7	-	reserved	-	-	-	-	-	-	-
		[10: 8]	RFDC	RX FIFO Depth Configuration	3'b000	3'b000	unch	unch	unch	-	-
		11	-	reserved	-	-	-	-	-	-	-
		12	RFIM	RX FIFO Interrupt Mode	0	0	unch	unch	unch	-	-
		[15: 13]	RFIGCV	RX FIFO Interrupt Generation Counter Value	3'b000	3'b000	unch	unch	unch	-	-
		16	RFFIE	RX FIFO Full interrupt Enable	0	0	unch	unch	unch	-	-
		[23: 17]	-	reserved	-	-	-	-	-	-	-
		24	MEIE	Message lost Error Interrupt Enable for FFI Mode	0	0	0 : normal mdoe unch : FFI mode	0 : normal mdoe unch : FFI mode	0 : normal mdoe unch : FFI mode	-	-
		[27: 25]	-	reserved	-	-	-	-	-	-	-
		28	DMAE	DMA Transfer Enable for RXFIFO a for FFI Mode	0	0	0 : normal mdoe unch : FFI mode	0	0 : normal mdoe unch : FFI mode	-	-
		[31: 29]	-	reserved	-	-	-	-	-	-	-
CFDRFSTS a	m_rxifou nit	RX FIFO Status Register s [7:0]									
		0	RFEMP	RX FIFO Empty	1	1	unch	1	unch	-	-
		1	RFULL	RX FIFO Full	0	0	unch	0	unch	-	-
		2	RFMLT	RX FIFO Msg Lost Flag	0	0	unch	0	unch	-	-
		3	RFIF	RX FIFO Interrupt Flag	0	0	unch	0	unch	-	-
		[7:4]	-	reserved	-	-	-	-	-	-	-
		[15: 8]	RFMC	RX FIFO Message Count	8'h00	8'h00	unch	8'h00	unch	-	-
		16	RFFIF	RX FIFO Full Interrupt Flag	0	0	unch	0	unch	-	-
		[27: 17]	-	reserved	-	-	-	-	-	-	-
		28	RFDMASTS	DMA Transfer Status for RX FIFO a for FFI Mode	0	0	0 : normal mdoe unch : FFI mode	0	0 : normal mdoe unch : FFI mode	-	-
		[31: 29]	-	reserved	-	-	-	-	-	-	-

CFDRFPCT Ra	m_rxifou nit	RX FIFO Pointer Control Register [7:0]										
		[7:0]	RFPC	RX FIFO Pointer Control	-	-	-	-	-	-	-	-
		[31: 8]	-	reserved	-	-	-	-	-	-	-	-
8. Common FIFO Registers												
CFDCFCCd	m_comfif ounit	Commo n FIFO Configur ation / Control Register s [23:0]										
		0	CFE	Common FIFO Enable	0	0	unch	0	unch	unch	0: TX / GW FIFO unch: RX FIFO	unch
		1	CFRXIE	Common FIFO Interrupt Enable for RX Mode	0	0	unch	unch	unch	unch	unch	unch
		2	CFTXIE	Common FIFO Interrupt Enable for TX Mode	0	0	unch	unch	unch	unch	unch	unch
		3	-	reserved	-	-	-	-	-	-	-	-
		[6:4]	CFPLS	Common FIFO Payload Data Size configurati on	3'b000	3'b000	unch	unch	unch	-	-	-
		7	-	reserved	-	-	-	-	-	-	-	-
		[9:8]	CFM	Common FIFO Mode	2'b00	2'b00	unch	unch	unch	unch	unch	unch
		10	CFITSS	Common FIFO Interval Timer Source Select	0	0	unch	unch	unch	unch	unch	unch
		11	CFITR	Common FIFO Interval Timer Reference Clock Resolutio n	0	0	unch	unch	unch	unch	unch	unch
		12	CFIM	Common FIFO Interrupt Mode	0	0	unch	unch	unch	unch	unch	unch
		[15: 13]	CFIGCV	Common FIFO Interrupt Generatio n Counter Value	3'b000	3'b000	unch	unch	unch	unch	unch	unch
		[20: 16]	CFTML	Common FIFO TX- Mailbox Link	5'b00000	5'b00000	unch	unch	unch	unch	unch	unch
		[23: 21]	CFDC	Common FIFO Depth Configurat ion	3'b000	3'b000	unch	unch	unch	unch	unch	unch
		[31: 24]	CFITT	Common FIFO Interval Transmiss ion Time	0	0	unch	unch	unch	unch	unch	unch
CFDCFCE d	m_comfif ounit	Commo n FIFO Configur ation / Control Register s2 [23:0]										

		0	CFFIE	COMMON FIFO Full interrupt Enable	0	0	unch	unch	unch	unch	unch	unch
		1	CFOFRXIE	COMMON FIFO One Frame Reception Interrupt Enable	0	0	unch	unch	unch	unch	unch	unch
		2	CFOFTXIE	COMMON FIFO One Frame Transmission Interrupt Enable	0	0	unch	unch	unch	unch	unch	unch
		[7:3]	-	reserved	-	-	-	-	-	-	-	-
		8	CFMOWM	Common FIFO message overwrite mode	0	0	unch	unch	unch	unch	unch	unch
		[15:9]	-	reserved	-	-	-	-	-	-	-	-
		16	CFBME	COMMON FIFO Buffering Mode Enable	0	0	unch	unch	unch	unch	unch	unch
		[23:17]	-	reserved	-	-	-	-	-	-	-	-
		24	MEIE	Message lost Error Interrupt Enable for FFI Mode	0	0	0 : normal mdoe unch : FFI mode					
		25	MOWEIE	GW FIFO Message overwrite Error Interrupt Enable for FFI Mode	0	0	0 : normal mdoe unch : FFI mode					
		[27:26]	-	reserved	-	-	-	-	-	-	-	-
		28	DMAE	DMA Transfer Enable for Common FIFO a for FFI Mode	0	0	0 : normal mdoe unch : FFI mode	0	0 : normal mdoe unch : FFI mode			
		[31:29]	-	reserved	-	-	-	-	-	-	-	-
CFDCFSTSd	m_comfifounit	Common FIFO status registers [23:0]										
		0	CFEMP	Common FIFO Empty	1	1	unch	1	unch	-	1: TX / GW FIFO unch: RX FIFO	-
		1	CFULL	Common FIFO Full	0	0	unch	0	unch	-	0: TX / GW FIFO unch: RX FIFO	-
		2	CFMLT	Common FIFO Msg Lost	0	0	unch	0	unch	-	0: TX / GW FIFO unch: RX FIFO	-
		3	CFRXIF	Common reception FIFO Interrupt Flag	0	0	unch	0	unch	-	0: TX / GW FIFO unch: RX FIFO	-
		4	CFTXIF	Common transmit FIFO Interrupt Flag	0	0	unch	0	unch	-	0: TX / GW FIFO unch: RX FIFO	-
		[7:5]	-	reserved	-	-	-	-	-	-	-	-
		[15:8]	CFMC	Common FIFO Message Count	8'h00	8'h00	unch	8'h00	unch	-	0: TX / GW FIFO unch: RX FIFO	-

			16	CFFIF	Common FIFO Full Interrupt Flag	0	0	unch	0	unch	-	0: TX / GW FIFO unch: RX FIFO	-
			17	CFOFRXIF	Common FIFO One Frame Reception	0	0	unch	0	unch	-	0: TX / GW FIFO unch: RX FIFO	-
			18	CFOFTXIF	Common FIFO One Frame Transmission	0	0	unch	0	unch	-	0: TX / GW FIFO unch: RX FIFO	-
		[23: 19]	-	reserved	-	-	-	-	-	-	-	-	-
			24	CFMOW	Common FIFO message overwrite	0	0	unch	0	unch	-	0: TX / GW FIFO unch: RX FIFO	-
		[27: 25]	-	reserved	-	-	-	-	-	-	-	-	-
			28	CFDMASTS	DMA Transfer Status for Common FIFO a for FFI Mode	0	0	0 : norm al mdoe unch : FFI mode	0	0 : norm al mdoe unch : FFI mode	-	-	-
		[31: 29]	-	reserved	-	-	-	-	-	-	-	-	-
CFDCFPCT Rd	m_comfif ounit	Comm o n FIFO Pointer Control Register [23:0]											
			[7:0]	CFPC	Common FIFO Pointer Control	-	-	-	-	-	-	-	-
			[31: 8]	-	reserved	-	-	-	-	-	-	-	-
9. FIFO Status Support Registers													
CFDFESTS	m_acsfr	FIFO Empty Status Register											
			[7:0]	RFxEMP	RX FIFO Empty Status	8'hff	8'hff	unch	8'hff	unch	-	-	-
			[31: 8]	CFxEMP	Common FIFO Empty Status	3'b111 * n	3'b111 * n	unch	3'b111 * n	unch	-	mirror of COM FIFO Status	-
CFDFFSTS	m_acsfr	FIFO Full Status Register											
			[7:0]	RFxFLL	RX FIFO Full Status	32'h0000 0000	32'h0000 0000	unch	8'h00	unch	-	-	-
			[31: 8]	CFxFLL	Common FIFO Full Status			unch	24'h0000 00	unch	-	mirror of COM FIFO Status	-
CFDFFFSTS	m_acsfr	FIFO FDC level Full Status Register											
			[7:0]	RFxFFLL	RX FIFO FDC level full Status	32'h0000 0000	32'h0000 0000	unch	8'h00	unch	-	-	-
			[31: 8]	CFxFFLL	COMMON FIFO FDC level full Status			unch	24'h0000 00	unch	-	mirror of COM FIFO Status	-
CFDFMSTS	m_acsfr	FIFO Msg Lost Status Register											
			[7:0]	RFxMLT	RX FIFO Msg Lost Status	32h0000 0000	32h0000 0000	unch	8'h00	unch	-	-	-



			[31: 8]	CFxMLT	Common FIFO Msg Lost Status			unch	24'h0000 00	unch	-	-	-	-
CFDCFMO WSTS	m_acsfr	Common FIFO Message OverWrite Status												
			[23: 0]	CFxMOW	Common FIFO [x] Message overwrite status	24'h0000 00	24'h0000 00	unch	24'h0000 00	unch	-	-	-	-
			[31: 24]	-	reserved	-	-	-	-	-	-	-	-	-
CFDRFISTS	m_acsfr	RX FIFO Interrupt Flag Status Register												
			[7:0]	RFxIF	RX FIFO Interrupt Flag Status	8'h00	8'h00	unch	8'h00	unch	-	-	-	-
			[15: 8]	-	reserved	-	-	-	-	-	-	-	-	-
			[23: 16]	RFxFFLL	RX FIFO Full Interrupt Flag Status	8'h00	8'h00	unch	8'h00	unch	-	-	-	-
			[31: 24]	-	reserved	-	-	-	-	-	-	-	-	-
CFDCFRISTS	m_acsfr	COM FIFO RX Interrupt Flag Status Register												
			[23: 0]	CFxRXIF	RX Common FIFO Interrupt Flag Status	24'h0000 00	24'h0000 00	unch	24'h0000 00	unch	-	-	-	-
			[31: 24]	-	reserved	-	-	-	-	-	-	-	-	-
CFDCFTISTS	m_acsfr	COM FIFO TX Interrupt Flag Status Register												
			[23: 0]	CFxTXIF	TX Common FIFO Interrupt Flag Status	24'h0000 00	24'h0000 00	unch	24'h0000 00	unch	-	-	-	-
			[31: 24]	-	reserved	-	-	-	-	-	-	-	-	-
CFDCFOFRISTS	m_acsfr	Common FIFO One Frame RX Interrupt Flag Status												
			[23: 0]	CFxOFRXIF	Common FIFO [x] One Frame RX Interrupt Flag Status	24'h0000 00	24'h0000 00	unch	24'h0000 00	unch	-	-	-	-
			[31: 24]	-	reserved	-	-	-	-	-	-	-	-	-

CFDCFOFTISTS	m_acsfr	Common FIFO One Frame TX Interrupt Flag Status											
			[23:0]	CFxOFTXIF	Common FIFO [x] One Frame TX Interrupt Flag Status	24'h0000 00	24'h0000 00	unch	24'h0000 00	unch	-	-	-
			[31:24]	-	reserved	-	-	-	-	-	-	-	-
10. Special CAN-FD register													
CFDCDTCT	m_dmaif	DMA Transfer Control Register											
			[7:0]	RFDMAEe	DMA Transfer Enable for RXFIFO e	16'h0000	16'h0000	unch : normal mdoe 8'h00 : FFI mode	8'h00	unch : normal mdoe 8'h00 : FFI mode	unch : normal mdoe 8'h00 : FFI mode	unch : normal mdoe 8'h00 : FFI mode	unch : normal mdoe 8'h00 : FFI mode
			[15:8]	CFDMAEn	DMA Transfer Enable for Common FIFO n			unch : normal mdoe 8'h00 : FFI mode	8'h00	unch : normal mdoe 8'h00 : FFI mode	unch : normal mdoe 8'h00 : FFI mode	unch : normal mdoe 8'h00 : FFI mode	unch : normal mdoe 8'h00 : FFI mode
			[31:16]	-	reserved	-	-	-	-	-	-	-	-
CFDCDTSTS	m_dmaif	DMA Transfer Status Register											
			[7:0]	RFDMAStse	DMA Transfer Status for RX FIFO e	16'h0000	16'h0000	unch	8'h00	unch	-	-	-
			[15:8]	CFDMAStsn	DMA Transfer Status only for Common FIFO 0 of channel n			unch	8'h00	unch	-	-	-
			[31:16]	-	Reserved	-	-	-	-	-	-	-	-
CFDCDTTC	m_dmaif	DMA TX Transfer Control Register											
			[7:0]	TQ0DMAEn	DMA TX Transfer Enable for TXQ0 n	8'h00	8'h00	unch : normal mdoe 8'h00 : FFI mode	8'h00	unch : normal mdoe 8'h00 : FFI mode	unch : normal mdoe 8'h00 : FFI mode	unch : normal mdoe 8'h00 : FFI mode	unch : normal mdoe 8'h00 : FFI mode
			[15:8]	TQ3DMAEn	DMA TX Transfer Enable for TXQ3 n	8'h00	8'h00	unch : normal mdoe 8'h00 : FFI mode	8'h00	unch : normal mdoe 8'h00 : FFI mode	unch : normal mdoe 8'h00 : FFI mode	unch : normal mdoe 8'h00 : FFI mode	unch : normal mdoe 8'h00 : FFI mode
			[23:16]	CFDMAEn	DMA Transfer Enable for Common FIFO n	8'h00	8'h00	unch : normal mdoe 8'h00 : FFI mode	8'h00	unch : normal mdoe 8'h00 : FFI mode	unch : normal mdoe 8'h00 : FFI mode	unch : normal mdoe 8'h00 : FFI mode	unch : normal mdoe 8'h00 : FFI mode
			[31:24]	-	reserved	-	-	-	-	-	-	-	-

CFDCDTTS TS	m_dmaif	DMA TX Transfer Status Register											
		[7:0]	TQ0DMAST Sn	DMA Transfer Status for TXQ0 n	8'h00	8'h00	unch	8'h00	unch	-	-	-	
		[15: 8]	TQ3DMAST Sn	DMA Transfer Status for TXQ3 n	8'h00	8'h00	unch	8'h00	unch	-	-	-	
		[23: 16]	CFDMAST n	DMA Transfer Status only for Common FIFO 0 of channel n	8'h00	8'h00	unch	8'h00	unch	-	-	-	
		[31: 24]	-	reserved	-	-	-	-	-	-	-	-	
11. TX Mailbox Registers													
CFDTMCI	m_mbctrl	TX Mailbox Control Register s i											
		0	TMTR	TX Mailbox Transmiss ion Request	0	0	unch	0	unch	unch	0	unch	
		1	TMTAR	TX Mailbox Transmiss ion Abortion Request	0	0	unch	0	unch	unch	0	unch	
		2	TMOM	TX Mailbox One-shot mode	0	0	unch	0	unch	unch	0	unch	
		[7:3]	-	reserved	-	-	-	-	-	-	-	-	
CFDTMSTS j	m_mbctrl	TX Mailbox Status Register s j											
		0	TMTSTS	TX Mailbox Transmiss ion Status	0	0	unch	0	0	unch	0	0	
		[2:1]	TMTRF	TX Mailbox Transmiss ion Result Flag	2'b00	2'b00	unch	2'b00	unch	unch	2'b00	unch	
		3	TMTRM	TX Mailbox Transmiss ion Request	0	0	unch	0	unch	unch	0	unch	
		4	TMTARM	TX Mailbox Transmiss ion Abortion Request	0	0	unch	0	unch	unch	0	unch	
		[7:5]	-	reserved	-	-	-	-	-	-	-	-	
CFDTMTRS TSf	m_mbctrl	TX Mailbox es Transmi ssion Request Status Register s											
		[31: 0]	CFDTMTRS TS	TX Mailbox Transmiss ion Request Status	32'h0000 0000	32'h0000 0000	unch	32'h0000 0000	unch	unch	will be cleared partly by channel reset	unch	

CFDTMTAR STSf	m_mbctrl	TX Mailbox es Transmission Abortion Request Status Registers											
			[31: 0]	CFDTMTAR STS	TX Mailbox Transmission Abortion Request Status	32'h0000 0000	32'h0000 0000	unch	32'h0000 0000	unch	unch	will be cleared partly by channel reset	unch
<hr/>													
CFDTMTCS TSf	m_mbctrl	TX Mailbox es Transmission Completion Status Registers											
			[31: 0]	CFDTMTCS TS	TX Mailbox Transmission Completion Status	32'h0000 0000	32'h0000 0000	unch	32'h0000 0000	unch	unch	will be cleared partly by channel reset	unch
<hr/>													
CFDTMTAS TSf	m_mbctrl	TX Mailbox es Transmission Abortion Status Registers											
			[31: 0]	CFDTMTAS TS	TX Mailbox Transmission Abortion Status	32'h0000 0000	32'h0000 0000	unch	32'h0000 0000	unch	unch	will be cleared partly by channel reset	unch
<hr/>													
CFDTMIECf	m_mbctrl	TX Mailbox es Interrupt Enable Configuration Registers											
			[31: 0]	TMIE	TX Mailbox Interrupt Enable	32'h0000 0000	32'h0000 0000	unch	unch	unch	unch	unch	unch
12. TX Queue Registers													
CFDTXQCC 0n	m_mbctrl	TX Queue0 Configuration / Control Register n											
			0	TXQE	TX Queue Enable	0	0	unch	0	unch	unch	0	unch
			1	TXQGWE	TX Queue Gateway Mode Enable	0	0	unch	unch	unch	unch	unch	unch
			2	TXQOWE	TX Queue Overwrite Mode Enable	0	0	unch	unch	unch	unch	unch	unch
			[4:3]	-	reserved	-	-	-	-	-	-	-	-
			5	TXQTXIE	TX Queue TX Interrupt Enable	0	0	unch	unch	unch	unch	unch	unch
			6	-	reserved	-	-	-	-	-	-	-	-
			7	TXQIM	TX Queue Interrupt Mode	0	0	unch	unch	unch	unch	unch	unch

			[12: 8]	TXQDC	TX Queue Depth Configuration	5'b00000	5'b00000	unch	unch	unch	unch	unch	unch
			[15: 13]	-	reserved	-	-	-	-	-	-	-	-
			16	TXQFIE	TXQ Full interrupt Enable	0	0	unch	unch	unch	unch	unch	unch
			17	TXQOFRXIE	TXQ One Frame Reception Interrupt Enable	0	0	unch	unch	unch	unch	unch	unch
			18	TXQOFTXIE	TXQ One Frame Transmission Interrupt Enable	0	0	unch	unch	unch	unch	unch	unch
			[23: 19]	-	reserved	-	-	-	-	-	-	-	-
			24	QMEIE	TXQ Message lost Error Interrupt Enable for FFI Mode	0	0	0 : normal mdoe unch : FFI mode					
			25	QOWEIE	TXQ Message overwrite Error Interrupt Enable for FFI Mode	0	0	0 : normal mdoe unch : FFI mode					
			[27: 26]	-	reserved	-	-	-	-	-	-	-	-
			28	DMAE	DMA TX Transfer Enable for TXQ for FFI Mode	0	0	0 : normal mdoe unch : FFI mode	0	0 : normal mdoe unch : FFI mode			
			[31: 29]	-	reserved	-	-	-	-	-	-	-	-
<hr/>													
CFDTXQCC 1n	m_mbctrl	TX Queue1 Configuration / Control Register n											
		0	TXQE	TX Queue Enable	0	0	unch	0	unch	unch	0	unch	
		1	TXQGWE	TX Queue Gateway Mode Enable	0	0	unch	unch	unch	unch	unch	unch	
		2	TXQOWE	TX Queue Overwrite Mode Enable	0	0	unch	unch	unch	unch	unch	unch	
		[4:3]	-	reserved	-	-	-	-	-	-	-	-	
		5	TXQTXIE	TX Queue TX Interrupt Enable	0	0	unch	unch	unch	unch	unch	unch	
		6	-	reserved	-	-	-	-	-	-	-	-	
		7	TXQIM	TX Queue Interrupt Mode	0	0	unch	unch	unch	unch	unch	unch	
		[12: 8]	TXQDC	TX Queue Depth Configuration	5'b00000	5'b00000	unch	unch	unch	unch	unch	unch	
		[15: 13]	-	reserved	-	-	-	-	-	-	-	-	
		16	TXQFIE	TXQ Full interrupt Enable	0	0	unch	unch	unch	unch	unch	unch	
		17	TXQOFRXIE	TXQ One Frame Reception Interrupt Enable	0	0	unch	unch	unch	unch	unch	unch	
		18	TXQOFTXIE	TXQ One Frame Transmission	0	0	unch	unch	unch	unch	unch	unch	

					Interrupt Enable									
		[23: 19]	-	reserved	-	-	-	-	-	-	-	-	-	-
		24	QMEIE	TXQ Message lost Error Interrupt Enable for FFI Mode	0	0	0 : normal mdoe unch : FFI mode							
		25	QOWEIE	TXQ Message overwrite Error Interrupt Enable for FFI Mode	0	0	0 : normal mdoe unch : FFI mode							
		[31: 26]	-	reserved	-	-	-	-	-	-	-	-	-	-
<hr/>														
CFDTXQCC 2n	m_mbctrl	TX Queue2 Configuration / Control Register n												
		0	TXQE	TX Queue Enable	0	0	unch	0	unch	unch	0	unch		
		1	TXQGWE	TX Queue Gateway Mode Enable	0	0	unch							
		2	TXQOWE	TX Queue Overwrite Mode Enable	0	0	unch							
		[4:3]	-	reserved	-	-	-	-	-	-	-	-	-	
		5	TXQTXIE	TX Queue TX Interrupt Enable	0	0	unch							
		6	-	reserved	-	-	-	-	-	-	-	-	-	
		7	TXQIM	TX Queue Interrupt Mode	0	0	unch							
		[12: 8]	TXQDC	TX Queue Depth Configuration	5'b00000	5'b00000	unch							
		[15: 13]	-	reserved	-	-	-	-	-	-	-	-	-	
		16	TXQFIE	TXQ Full interrupt Enable	0	0	unch							
		17	TXQOFRXIE	TXQ One Frame Reception Interrupt Enable	0	0	unch							
		18	TXQOFTXIE	TXQ One Frame Transmission Interrupt Enable	0	0	unch							
		[23: 19]	-	reserved	-	-	-	-	-	-	-	-	-	
		24	QMEIE	TXQ Message lost Error Interrupt Enable for FFI Mode	0	0	0 : normal mdoe unch : FFI mode							
		25	QOWEIE	TXQ Message overwrite Error Interrupt Enable for FFI Mode	0	0	0 : normal mdoe unch : FFI mode							
		[31: 26]	-	reserved	-	-	-	-	-	-	-	-	-	
<hr/>														
CFDTXQCC 3n	m_mbctrl	TX Queue3 Configuration / Control												

		Register n											
		0	TXQE	TX Queue Enable	0	0	unch	0	unch	unch	0	unch	
		1	-	reserved	-	-	-	-	-	-	-	-	
		2	TXQOWE	TX Queue Overwrite Mode Enable	0	0	unch	unch	unch	unch	unch	unch	
		[4:3]	-	reserved	-	-	-	-	-	-	-	-	
		5	TXQTXIE	TX Queue TX Interrupt Enable	0	0	unch	unch	unch	unch	unch	unch	
		6	-	reserved	-	-	-	-	-	-	-	-	
		7	TXQIM	TX Queue Interrupt Mode	0	0	unch	unch	unch	unch	unch	unch	
		[12: 8]	TXQDC	TX Queue Depth Configuration	5'b00000	5'b00000	unch	unch	unch	unch	unch	unch	
		[17: 13]	-	reserved	-	-	-	-	-	-	-	-	
		18	TXQOFTXIE	TXQ One Frame Transmission Interrupt Enable	0	0	unch	unch	unch	unch	unch	unch	
		[31: 19]	-	reserved	-	-	-	-	-	-	-	-	
		[24: 19]	-	reserved	-	-	-	-	-	-	-	-	
		25	QOWEIE	TXQ Message overwrite Error Interrupt Enable for FFI Mode	0	0	0 : normal mdoe unch : FFI mode						
		[27: 26]	-	reserved	-	-	-	-	-	-	-	-	
		28	DMAE	DMA TX Transfer Enable for TXQ for FFI Mode	0	0	0 : normal mdoe unch : FFI mode	0	0 : normal mdoe unch : FFI mode				
		[31: 29]	-	reserved	-	-	-	-	-	-	-	-	
<hr/>													
CFDTXQSTS0n	m_mbctrl	TX Queue0 Status Register n											
		0	TXQEMP	TX Queue Empty	1	1	unch	1	unch	unch	1	unch	
		1	TXQFLL	TX Queue Full	0	0	unch	0	unch	unch	0	unch	
		2	TXQTXIF	TX Queue TX Interrupt Flag	0	0	unch	0	unch	unch	0	unch	
		[7:3]	-	reserved	-	-	-	-	-	-	-	-	
		[13: 8]	TXQMC	CHn TX Queue Message counter	5'h00	5'h00	unch	5'h00	unch	unch	5'h00	unch	
		[15: 14]	-	reserved	-	-	-	-	-	-	-	-	
		16	TXQFIF	TXQ Full Interrupt Flag	0	0	unch	0	unch	unch	0	unch	
		17	TXQOFRXIF	TXQ One Frame Reception Interrupt Flag	0	0	unch	0	unch	unch	0	unch	
		18	TXQOFTXIF	TXQ One Frame Transmission Interrupt Flag	0	0	unch	0	unch	unch	0	unch	

			19	TXQMLT	TXQ Message Lost	0	0	unch	0	unch	unch	0	unch
			20	TXQMOW	TXQ message overwrite	0	0	unch	0	unch	unch	0	unch
			[27: 21]	-	reserved	-	-	-	-	-	-	-	-
			28	TXQDMASTS	DMA TX Transfer Status for TXQ0	0	0	0 : normal mdoe unch : FFI mode	0	0 : normal mdoe unch : FFI mode			
			[31: 29]	-	reserved	-	-	-	-	-	-	-	-
<hr/>													
CFDTXQST S1n	m_mbctrl	TX Queue1 Status Register n											
		0	TXQEMP	TX Queue Empty	1	1	unch	1	unch	unch	1	unch	
		1	TXQFLL	TX Queue Full	0	0	unch	0	unch	unch	0	unch	
		2	TXQTXIF	TX Queue TX Interrupt Flag	0	0	unch	0	unch	unch	0	unch	
		[7:3]	-	reserved	-	-	-	-	-	-	-	-	
		[13: 8]	TXQMC	CHn TX Queue Message counter	5'h00	5'h00	unch	5'h00	unch	unch	5'h00	unch	
		[15: 14]	-	reserved	-	-	-	-	-	-	-	-	
		16	TXQFIF	TXQ Full Interrupt Flag	0	0	unch	0	unch	unch	0	unch	
		17	TXQOFRXIF	TXQ One Frame Reception Interrupt Flag	0	0	unch	0	unch	unch	0	unch	
		18	TXQOFTXIF	TXQ One Frame Transmission Interrupt Flag	0	0	unch	0	unch	unch	0	unch	
		19	TXQMLT	TXQ Message Lost	0	0	unch	0	unch	unch	0	unch	
		20	TXQMOW	TXQ message overwrite	0	0	unch	0	unch	unch	0	unch	
		[31: 21]	-	reserved	-	-	-	-	-	-	-	-	
<hr/>													
CFDTXQST S2n	m_mbctrl	TX Queue2 Status Register n											
		0	TXQEMP	TX Queue Empty	1	1	unch	1	unch	unch	1	unch	
		1	TXQFLL	TX Queue Full	0	0	unch	0	unch	unch	0	unch	
		2	TXQTXIF	TX Queue TX Interrupt Flag	0	0	unch	0	unch	unch	0	unch	
		[7:3]	-	reserved	-	-	-	-	-	-	-	-	
		[13: 8]	TXQMC	CHn TX Queue Message counter	5'h00	5'h00	unch	5'h00	unch	unch	5'h00	unch	
		[15: 14]	-	reserved	-	-	-	-	-	-	-	-	
		16	TXQFIF	TXQ Full Interrupt Flag	0	0	unch	0	unch	unch	0	unch	
		17	TXQOFRXIF	TXQ One Frame Reception Interrupt Flag	0	0	unch	0	unch	unch	0	unch	
		18	TXQOFTXIF	TXQ One Frame Transmission	0	0	unch	0	unch	unch	0	unch	

					ion Interrupt Flag								
			19	TXQMLT	TXQ Message Lost	0	0	unch	0	unch	unch	0	unch
			20	TXQMOW	TXQ message overwrite	0	0	unch	0	unch	unch	0	unch
			[31: 21]	-	reserved	-	-	-	-	-	-	-	-
CFDTXQST S3n	m_mbctrl	TX Queue3 Status Register n											
		0	TXQEMP	TX Queue Empty	1	1	unch	1	unch	unch	1	unch	
		1	TXQFLL	TX Queue Full	0	0	unch	0	unch	unch	0	unch	
		2	TXQTXIF	TX Queue TX Interrupt Flag	0	0	unch	0	unch	unch	0	unch	
		[7:3]	-	reserved	-	-	-	-	-	-	-	-	-
		[13: 8]	TXQMC	CHn TX Queue Message counter	5'h00	5'h00	unch	5'h00	unch	unch	5'h00	unch	
		[17: 14]	-	reserved	-	-	-	-	-	-	-	-	
		18	TXQOFTXIF	TXQ One Frame Transmiss ion Interrupt Flag	0	0	unch	0	unch	unch	0	unch	
		19	-	reserved	-	-	-	-	-	-	-	-	
		20	TXQMOW	TXQ message overwrite	0	0	unch	0	unch	unch	0	unch	
		[27: 21]	-	reserved	-	-	-	-	-	-	-	-	
		28	TXQDMAST S	DMA TX Transfer Status for TXQ3	0	0	0 : norm al mdoe unch : FFI mode	0	0 : norm al mdoe unch : FFI mode	0 : norm al mdoe unch : FFI mode	0 : normal mdoe unch : FFI mode	0 : norm al mdoe unch : FFI mode	
		[31: 29]	-	reserved	-	-	-	-	-	-	-	-	
CFDTXQPC TR0n	m_mbctrl	TX Queue0 Pointer Control Register n											
		[7:0]	TXQPC	Channel n TX Queue Pointer Control	-	-	-	-	-	-	-	-	
		[31: 8]	-	reserved	-	-	-	-	-	-	-	-	
CFDTXQPC TR1n	m_mbctrl	TX Queue1 Pointer Control Register n											
		[7:0]	TXQPC	Channel n TX Queue Pointer Control	-	-	-	-	-	-	-	-	
		[31: 8]	-	reserved	-	-	-	-	-	-	-	-	
CFDTXQPC TR2n	m_mbctrl	TX Queue2 Pointer Control Register n											
		[7:0]	TXQPC	Channel n TX Queue Pointer Control	-	-	-	-	-	-	-	-	

			[31: 8]	-	reserved	-	-	-	-	-	-	-	-	-
CFDTXQPC TR3n	m_mbctrl	TX Queue3 Pointer Control Register n												
			[7:0]	TXQPC	Channel n TX Queue Pointer Control	-	-	-	-	-	-	-	-	-
			[31: 8]	-	reserved	-	-	-	-	-	-	-	-	-
CFDTXQES TS	m_mbctrl	TX Queue Empty Status Status Register s												
			[31: 0]	TXQEMP	TXQ empty Status	32'hFFFF FFFF	32'hFFFF FFFF	-	-	-	unch	32'hFFFF FFFF	unch	
CFDTXQFI STS	m_mbctrl	TX Queue Full Interrupt Status Status Register s												
			[2:0]	TXQFULL[2: 0]	TXQ Full Interrupt Status	3'b000	3'b000	-	-	-	unch	3'b000	unch	
			3	-	reserved	-	-	-	-	-	-	-	-	-
			[6:4]	TXQFULL[6: 4]	TXQ Full Interrupt Status	3'b000	3'b000	-	-	-	unch	3'b000	unch	
			7	-	reserved	-	-	-	-	-	-	-	-	-
			[10: 8]	TXQFULL[1 0:8]	TXQ Full Interrupt Status	3'b000	3'b000	-	-	-	unch	3'b000	unch	
			11	-	reserved	-	-	-	-	-	-	-	-	-
			[14: 12]	TXQFULL[1 4:12]	TXQ Full Interrupt Status	3'b000	3'b000	-	-	-	unch	3'b000	unch	
			15	-	reserved	-	-	-	-	-	-	-	-	-
			[18: 16]	TXQFULL[1 8:16]	TXQ Full Interrupt Status	3'b000	3'b000	-	-	-	unch	3'b000	unch	
			19	-	reserved	-	-	-	-	-	-	-	-	-
			[22: 20]	TXQFULL[2 2:20]	TXQ Full Interrupt Status	3'b000	3'b000	-	-	-	unch	3'b000	unch	
			23	-	reserved	-	-	-	-	-	-	-	-	-
			[26: 24]	TXQFULL[2 6:24]	TXQ Full Interrupt Status	3'b000	3'b000	-	-	-	unch	3'b000	unch	
			27	-	reserved	-	-	-	-	-	-	-	-	-
			[30: 28]	TXQFULL[3 0:28]	TXQ Full Interrupt Status	3'b000	3'b000	-	-	-	unch	3'b000	unch	
			31	-	reserved	-	-	-	-	-	-	-	-	-
CFDTXQMS TS	m_mbctrl	TX Queue Message lost Status Status Register s												
			[2:0]	TXQML[2:0]	TXQ message lost Status	3'b000	3'b000	-	-	-	unch	3'b000	unch	
			3	-	reserved	-	-	-	-	-	-	-	-	-
			[6:4]	TXQML[6:4]	TXQ message lost Status	3'b000	3'b000	-	-	-	unch	3'b000	unch	
			7	-	reserved	-	-	-	-	-	-	-	-	-

			[10:8]	TXQML[10:8]	TXQ message lost Status	3'b000	3'b000	-	-	-	-	unch	3'b000	unch
			11	-	reserved	-	-	-	-	-	-	-	-	-
			[14:12]	TXQML[14:12]	TXQ message lost Status	3'b000	3'b000	-	-	-	-	unch	3'b000	unch
			15	-	reserved	-	-	-	-	-	-	-	-	-
			[18:16]	TXQML[18:16]	TXQ message lost Status	3'b000	3'b000	-	-	-	-	unch	3'b000	unch
			19	-	reserved	-	-	-	-	-	-	-	-	-
			[22:20]	TXQML[22:20]	TXQ message lost Status	3'b000	3'b000	-	-	-	-	unch	3'b000	unch
			23	-	reserved	-	-	-	-	-	-	-	-	-
			[26:24]	TXQML[26:24]	TXQ message lost Status	3'b000	3'b000	-	-	-	-	unch	3'b000	unch
			27	-	reserved	-	-	-	-	-	-	-	-	-
			[30:28]	TXQML[30:28]	TXQ Full Interrupt Status	3'b000	3'b000	-	-	-	-	unch	3'b000	unch
			31	-	reserved	-	-	-	-	-	-	-	-	-
<hr/>														
CFDTXQO WSTS	m_mbctrl	TX Queue Message Overwrite Status Register												
			[31:0]	TXQOW	TXQ message overwrite Status	32'h0000 0000	32'h0000 0000	-	-	-	-	unch	32'h0000 0000	unch
<hr/>														
CFDTXQIS TS	m_mbctrl	TX Queue Interrupt Status Register												
			[31:0]	TXQISF	TXQ Interrupt Status Flag	32'h0000 0000	32'h0000 0000	-	-	-	-	unch	32'h0000 0000	unch
<hr/>														
CFDTXQOF TISTS	m_mbctrl	TX Queue One Frame TX Interrupt Status Register												
			[31:0]	TXQOFTIF	TXQ One Frame Tx Interrupt Status Flag	32'h0000 0000	32'h0000 0000	-	-	-	-	unch	32'h0000 0000	unch
<hr/>														
CFDTXQOF RISTS	m_mbctrl	TX Queue One Frame RX Interrupt Status Register												
			[2:0]	TXQOFRISF [2:0]	TXQ One Frame RX Interrupt Status Flag	3'b000	3'b000	-	-	-	-	unch	3'b000	unch
			3	-	reserved	-	-	-	-	-	-	-	-	-
			[6:4]	TXQOFRISF [6:4]	TXQ One Frame RX Interrupt Status Flag	3'b000	3'b000	-	-	-	-	unch	3'b000	unch
			7	-	reserved	-	-	-	-	-	-	-	-	-

			[10:8]	TXQOFRISF [10:8]	TXQ One Frame RX Interrupt Status Flag	3'b000	3'b000	-	-	-	-	unch	3'b000	unch
			11	-	reserved	-	-	-	-	-	-	-	-	-
			[14:12]	TXQOFRISF [14:12]	TXQ One Frame RX Interrupt Status Flag	3'b000	3'b000	-	-	-	-	unch	3'b000	unch
			15	-	reserved	-	-	-	-	-	-	-	-	-
			[18:16]	TXQOFRISF [18:16]	TXQ One Frame RX Interrupt Status Flag	3'b000	3'b000	-	-	-	-	unch	3'b000	unch
			19	-	reserved	-	-	-	-	-	-	-	-	-
			[22:20]	TXQOFRISF [22:20]	TXQ One Frame RX Interrupt Status Flag	3'b000	3'b000	-	-	-	-	unch	3'b000	unch
			23	-	reserved	-	-	-	-	-	-	-	-	-
			[26:24]	TXQOFRISF [26:24]	TXQ One Frame RX Interrupt Status Flag	3'b000	3'b000	-	-	-	-	unch	3'b000	unch
			27	-	reserved	-	-	-	-	-	-	-	-	-
			[30:28]	TXQOFRISF [30:28]	TXQ One Frame RX Interrupt Status Flag	3'b000	3'b000	-	-	-	-	unch	3'b000	unch
			31	-	reserved	-	-	-	-	-	-	-	-	-
CFDTXQFS TS	m_mbctrl	TX Queue Full Status Registers												
			[31:0]	TXQFSF	TXQ Full Status	32'h0000 0000	32'h0000 0000	-	-	-	-	unch	32'h0000 0000	unch
13. TX History List Registers														
CFDTHLCC n	m_thfifo unit	Channel n TX History List Configuration / Control Register												
			0	THLE	TX History List Enable	0	0	unch	0	unch	unch	0 : normal mode unch : FFI mode	unch	unch
			[7:1]	-	reserved	-	-	-	-	-	-	-	-	-
			8	THLIE	TX History List Interrupt Enable	0	0	unch	unch	unch	unch	unch	unch	unch
			9	THLIM	TX History List Interrupt Mode	0	0	unch	unch	unch	unch	unch	unch	unch
			10	THLDTE	TX History List Dedicated TX Enable	0	0	unch	unch	unch	unch	unch	unch	unch
			11	THLDGE	TX History List Dedicated GW Enable	0	0	unch	unch	unch	unch	unch	unch	unch
			[23:12]	-	reserved	-	-	-	-	-	-	-	-	-
			24	THLEIE	TX History List Entry Lost Interrupt Enable	0	0	unch	unch	unch	unch	unch	unch	unch
			[31:25]	-	reserved	-	-	-	-	-	-	-	-	-

CFDTHLST Sn	m_thlfifo unit	Channel n TX History List Status Register											
			0	THLEMP	TX History List Empty	1	1	unch	1	unch	unch	1 : normal mdoe unch : FFI mode	unch
			1	THLFLL	TX History List Full	0	0	unch	0	unch	unch	0 : normal mdoe unch : FFI mode	unch
			2	THLELT	TX History List Entry Lost Flag	0	0	unch	0	unch	unch	0 : normal mdoe unch : FFI mode	unch
			3	THLIF	TX History List Interrupt Flag	0	0	unch	0	unch	unch	0 : normal mdoe unch : FFI mode	unch
			[7:4]	-	reserved	-	-	-	-	-	-	-	-
			[13: 8]	THLMC	TX History List Message Count	6'h00	6'h00	unch	6'h00	unch	-	6'h00 : normal mdoe unch : FFI mode	-
			[31: 14]	-	reserved	-	-	-	-	-	-	-	-
CFDTHLPC TR	m_thlfifo unit	CHn TX History List pointer control register											
			[7:0]	THLPC	TX History List Pointer Control	-	-	-	-	-	-	-	-
			[31: 8]	-	reserved	-	-	-	-	-	-	-	-
14. TX Interrupt Status Register													
CFDGINT STSv	m_interr upt	Global Interrupt Status Register v											
			0	TSIFO	TX Successfu l Transmiss ion Interrupt Flag Channel 0	0	0	unch	0	unch	unch	0	unch
			1	TAIFO	TX Abortion Interrupt Flag Channel 0	0	0	unch	0	unch	unch	0	unch
			2	TQIFO	TX Queue Interrupt Flag Channel 0	0	0	unch : norm al mdoe 0 : FFI mode	0	unch : norm al mdoe 0 : FFI mode	unch : norm al mdoe 0 : FFI mode	0	unch : norm al mdoe 0 : FFI mode
			3	CFTIFO	COM FIFO TX/GW Mode Interrupt Flag Channel 0	0	0	unch : norm al mdoe 0 : FFI mode	0	unch : norm al mdoe 0 : FFI mode	unch : norm al mdoe 0 : FFI mode	0	unch : norm al mdoe 0 : FFI mode
			4	THIFO	TX History List Interrupt Channel 0	0	0	unch : norm al mdoe 0 : FFI mode	0	unch : norm al mdoe 0 : FFI mode	unch : norm al mdoe 0 : FFI mode	0	unch : norm al mdoe 0 : FFI mode

			5	TQOFIFO	TX Queue One Frame Transmission Interrupt Flag Channel 0	0	0	unch : normal mdoe 0 : FFI mode	0	unch : normal mdoe 0 : FFI mode	unch : normal mdoe 0 : FFI mode	0	unch : normal mdoe 0 : FFI mode
			6	CFOTIF0	COM FIFO One Frame Transmission Interrupt Channel 0	0	0	unch : normal mdoe 0 : FFI mode	0	unch : normal mdoe 0 : FFI mode	unch : normal mdoe 0 : FFI mode	0	unch : normal mdoe 0 : FFI mode
			7	-	reserved	-	-	-	-	-	-	-	-
			8	TSIF1	TX Successful Transmission Interrupt Flag Channel 1	0	0	unch	0	unch	unch	0	unch
			9	TAIF1	TX Abortion Interrupt Flag Channel 1	0	0	unch	0	unch	unch	0	unch
			10	TQIF1	TX Queue Interrupt Flag Channel 1	0	0	unch : normal mdoe 0 : FFI mode	0	unch : normal mdoe 0 : FFI mode	unch : normal mdoe 0 : FFI mode	0	unch : normal mdoe 0 : FFI mode
			11	CFTIF1	COM FIFO TX/GW Mode Interrupt Flag Channel 1	0	0	unch : normal mdoe 0 : FFI mode	0	unch : normal mdoe 0 : FFI mode	unch : normal mdoe 0 : FFI mode	0	unch : normal mdoe 0 : FFI mode
			12	THIF1	TX History List Interrupt Channel 1	0	0	unch : normal mdoe 0 : FFI mode	0	unch : normal mdoe 0 : FFI mode	unch : normal mdoe 0 : FFI mode	0	unch : normal mdoe 0 : FFI mode
			13	TQOFIF1	TX Queue One Frame Transmission Interrupt Flag Channel 1	0	0	unch : normal mdoe 0 : FFI mode	0	unch : normal mdoe 0 : FFI mode	unch : normal mdoe 0 : FFI mode	0	unch : normal mdoe 0 : FFI mode
			14	CFOTIF1	COM FIFO One Frame Transmission Interrupt Channel 1	0	0	unch : normal mdoe 0 : FFI mode	0	unch : normal mdoe 0 : FFI mode	unch : normal mdoe 0 : FFI mode	0	unch : normal mdoe 0 : FFI mode
			15	-	reserved	-	-	-	-	-	-	-	-
			16	TSIF2	TX Successful Transmission Interrupt Flag Channel 2	0	0	unch	0	unch	unch	0	unch
			17	TAIF2	TX Abortion Interrupt Flag Channel 2	0	0	unch	0	unch	unch	0	unch
			18	TQIF2	TX Queue Interrupt Flag Channel 2	0	0	unch : normal mdoe 0 : FFI mode	0	unch : normal mdoe 0 : FFI mode	unch : normal mdoe 0 : FFI mode	0	unch : normal mdoe 0 : FFI mode

			19	CFTIF2	COM FIFO TX/GW Mode Interrupt Flag Channel 2	0	0	unch : norm al mdoe 0 : FFI mode	0	unch : norm al mdoe 0 : FFI mode	unch : norm al mdoe 0 : FFI mode	0	unch : norm al mdoe 0 : FFI mode
			20	THIF2	TX History List Interrupt Channel 2	0	0	unch : norm al mdoe 0 : FFI mode	0	unch : norm al mdoe 0 : FFI mode	unch : norm al mdoe 0 : FFI mode	0	unch : norm al mdoe 0 : FFI mode
			21	TQOEIF2	TX Queue One Frame Transmission Interrupt Flag Channel 2	0	0	unch : norm al mdoe 0 : FFI mode	0	unch : norm al mdoe 0 : FFI mode	unch : norm al mdoe 0 : FFI mode	0	unch : norm al mdoe 0 : FFI mode
			22	CFOTIF2	COM FIFO One Frame Transmission Interrupt Channel 2	0	0	unch : norm al mdoe 0 : FFI mode	0	unch : norm al mdoe 0 : FFI mode	unch : norm al mdoe 0 : FFI mode	0	unch : norm al mdoe 0 : FFI mode
			23	-	reserved	-	-	-	-	-	-	-	-
			24	TSIF3	TX Successful Transmission Interrupt Flag Channel 3	0	0	unch	0	unch	unch	0	unch
			25	TAIF3	TX Abortion Interrupt Flag Channel 3	0	0	unch	0	unch	unch	0	unch
			26	TQIF3	TX Queue Interrupt Flag Channel 3	0	0	unch : norm al mdoe 0 : FFI mode	0	unch : norm al mdoe 0 : FFI mode	unch : norm al mdoe 0 : FFI mode	0	unch : norm al mdoe 0 : FFI mode
			27	CFTIF3	COM FIFO TX/GW Mode Interrupt Flag Channel 3	0	0	unch : norm al mdoe 0 : FFI mode	0	unch : norm al mdoe 0 : FFI mode	unch : norm al mdoe 0 : FFI mode	0	unch : norm al mdoe 0 : FFI mode
			28	THIF3	TX History List Interrupt Channel 3	0	0	unch : norm al mdoe 0 : FFI mode	0	unch : norm al mdoe 0 : FFI mode	unch : norm al mdoe 0 : FFI mode	0	unch : norm al mdoe 0 : FFI mode
			29	TQOEIF3	TX Queue One Frame Transmission Interrupt Flag Channel 3	0	0	unch : norm al mdoe 0 : FFI mode	0	unch : norm al mdoe 0 : FFI mode	unch : norm al mdoe 0 : FFI mode	0	unch : norm al mdoe 0 : FFI mode
			30	CFOTIF3	COM FIFO One Frame Transmission Interrupt Channel 3	0	0	unch : norm al mdoe 0 : FFI mode	0	unch : norm al mdoe 0 : FFI mode	unch : norm al mdoe 0 : FFI mode	0	unch : norm al mdoe 0 : FFI mode
			31	-	reserved	-	-	-	-	-	-	-	-
15. Global Test / Evaluation / Diag. Control Register													
CFDTSTC FG	m_test	Global Test Configuration Register											

			[7:0]	CnICBCE	Channel n Internal CAN Bus Communication Test Mode Enable	8'b00000000	8'b00000000	unch	8'b00000000	unch	-	-	-
			[15:8]	-	reserved	-	-	-	-	-	-	-	-
			[25:16]	RTMPS	RAM Test Mode Page Select	9'b00000000	9'b00000000	unch	9'b00000000	unch	-	-	-
			[30:26]	-	reserved	-	-	-	-	-	-	-	-
			31	PNFS	Pretended Network Filter List RAM Select	0	0	unch	0	unch	-	-	-
<hr/>													
CFDGSTCTR	m_test	Global Test Control Register											
			0	ICBCTME	Internal CAN Bus Communication Test Mode Enable	0	0	unch	0	unch	-	-	-
			1	-	reserved	-	-	-	-	-	-	-	-
			2	RTME	RAM Test Mode Enable	0	0	unch	0	unch	-	-	-
			[31:3]	-	reserved	-	-	-	-	-	-	-	-
<hr/>													
CFDGLOCKK	m_test	Global Lock Key Register											
			[15:0]	LOCK	Lock Key	-	-	-	-	-	-	-	-
			[31:16]	-	reserved	-	-	-	-	-	-	-	-
<hr/>													
CFDGLOTB	m_otbfifounit	OTB FIFO Configuration / Status Registers											
			0	OTBFE	OTB FIFO Enable	1	1	unch	1	unch	-	-	-
			[7:1]	-	reserved	-	-	-	-	-	-	-	-
			8	OTBEMP	OTB FIFO Empty	1	1	unch	1	unch	-	-	-
			9	OTBFLL	OTB FIFO Full	0	0	unch	0	unch	-	-	-
			10	OTBMLT	OTB FIFO Msg Lost Flag	0	0	unch	0	unch	-	-	-
			[15:11]	OTBMC	OTB FIFO Message Count	5'h00	5'h00	unch	5'h00	unch	-	-	-
			[31:16]	-	reserved	-	-	-	-	-	-	-	-
16. Bus load counter Register													
CFDCnBLS TS	m_busloadad	BUS Load counter Register											
			[2:0]	-	reserved	-	-	-	-	-	-	-	-
			[31:3]	BLC	BUS load counter status	29'h00000000	29'h00000000	-	-	-	-	-	-
<hr/>													
CFDCnBLC T	m_busloadad	BUS Load counter control Register											
			0	BLCE	BUS load counter enable	0	0	unch	0	-	unch	0	-

			[7: 1]	-	reserved	-	-	-	-	-	-	-	-	-
			8	BLCLD	BUS load counter load	-	-	-	-	-	-	-	-	-
			[31: 9]	-	reserved	-	-	-	-	-	-	-	-	-
17. Flexible CAN mode Register														
CFDGFCMC	m_comsf r	Global Flexible CAN mode Configur ation Register												
			0	FLXC0	Flexible CAN mode between Channel 0 and Channel 1	0	0	unch	unch	unch	-	-	-	-
			1	FLXC1	Flexible CAN mode between Channel 2 and Channel 3	0	0	unch	unch	unch	-	-	-	-
			2	FLXC2	Flexible CAN mode between Channel 4 and Channel 5	0	0	unch	unch	unch	-	-	-	-
			3	FLXC3	Flexible CAN mode between Channel 6 and Channel 7	0	0	unch	unch	unch	-	-	-	-
			[31: 4]	-	reserved	-	-	-	-	-	-	-	-	-
18. Flexible transmission buffer assignment														
CFDGFTBAC	m_bus_if	Global Flexible transmis sion buffer assignm ent Configur ation Register												
			[3:0]	FLXMB0	Flexible transmissi on buffer assignme nt between Channel 0 and Channel 1	0	0	unch	unch	unch	-	-	-	-
			[7:4]	-	reserved	-	-	-	-	-	-	-	-	-
			[11: 8]	FLXMB1	Flexible transmissi on buffer assignme nt between Channel 2 and Channel 3	0	0	unch	unch	unch	-	-	-	-
			[15: 12]	-	reserved	-	-	-	-	-	-	-	-	-
			[19: 16]	FLXMB2	Flexible transmissi on buffer assignme nt between Channel 4 and Channel 5	0	0	unch	unch	unch	-	-	-	-
			[23: 20]	-	reserved	-	-	-	-	-	-	-	-	-
			[27: 24]	FLXMB3	Flexible transmissi on buffer assignme nt between Channel 6	0	0	unch	unch	unch	-	-	-	-

					and Channel 7									
			[31: 28]	-	reserved	-	-	-	-	-	-	-	-	-
19. BUS Interface Register														
CFDGBISC	m_bus_if	Global Bus Interface Select Configuration Register												
			0	IFSW	Select BUS Interface APB or R-ACE	0	unch	unch	unch	unch	-	-	-	-
			[7:1]	-	reserved	-	-	-	-	-	-	-	-	-
			[15: 8]	KEY	Key code	-	-	-	-	-	-	-	-	-
			[31: 16]	-	reserved	-	-	-	-	-	-	-	-	-
20. Reset Control Register														
CFDGRSTC	m_bus_if	Global SW reset Register												
			0	SRST	SW reset	0	unch	unch	unch	unch	-	-	-	-
			[7:1]	-	reserved	-	-	-	-	-	-	-	-	-
			[15: 8]	KEY	Key code	-	-	-	-	-	-	-	-	-
			[31: 16]	-	reserved	-	-	-	-	-	-	-	-	-
21. RX Interrupt Status Register (COMMON FIFO RX-GW mode & TXQ GW mode)														
CFDGRINT STS _n	m_interr upt	Global RX Interrupt Status Register n												
			[2:0]	QFIF	TXQ Full Interrupt Flag Channel n	0	0	unch : normal mdoe 0 : FFI mode	0	unch : normal mdoe 0 : FFI mode	unch : normal mdoe 0 : FFI mode	0	unch : normal mdoe 0 : FFI mode	
			[3]	-	reserved	-	-	-	-	-	-	-	-	-
			[5:4]	BQFIF	Borrowed TXQ Full Interrupt Flag Channel n	0	0	unch : normal mdoe 0 : FFI mode	0	unch : normal mdoe 0 : FFI mode	unch : normal mdoe 0 : FFI mode	0	unch : normal mdoe 0 : FFI mode	
			[7:6]	-	reserved	-	-	-	-	-	-	-	-	-
			[10: 8]	QOFRIF	TXQ One Frame RX Interrupt Flag Channel n	0	0	unch : normal mdoe 0 : FFI mode	0	unch : normal mdoe 0 : FFI mode	unch : normal mdoe 0 : FFI mode	0	unch : normal mdoe 0 : FFI mode	
			[11]	-	reserved	-	-	-	-	-	-	-	-	-
			[13: 12]	BQOFRIF	Borrowed TXQ One Frame RX Interrupt Flag Channel n	0	0	unch : normal mdoe 0 : FFI mode	0	unch : normal mdoe 0 : FFI mode	unch : normal mdoe 0 : FFI mode	0	unch : normal mdoe 0 : FFI mode	
			[15: 14]	-	reserved	-	-	-	-	-	-	-	-	-
			[18: 16]	CFRIF	Common FIFO RX Interrupt Flag Channel n	0	0	unch : normal mdoe 0 : FFI mode	0	unch : normal mdoe 0 : FFI mode	unch : normal mdoe 0 : FFI mode	0	unch : normal mdoe 0 : FFI mode	

			[23: 19]	-	reserved	-	-	-	-	-	-	-	-	-
			[26: 24]	CFRFIF	Common FIFO One Frame FDC level Full Interrupt Flag Channel n	0	0	unch : normal mdoe 0 : FFI mode	0	unch : normal mdoe 0 : FFI mode	unch : normal mdoe 0 : FFI mode	0	unch : normal mdoe 0 : FFI mode	
			27	-	reserved	-	-	-	-	-	-	-	-	-
			[30: 28]	CFOFRIF	Common FIFO One Frame RX Interrupt Flag Channel n	0	0	unch : normal mdoe 0 : FFI mode	0	unch : normal mdoe 0 : FFI mode	unch : normal mdoe 0 : FFI mode	0	unch : normal mdoe 0 : FFI mode	
			31	-	reserved	-	-	-	-	-	-	-	-	-
22. Global AFL Ignore register (Entry & Control)														
CFDGAFIIGNENT	m_acsfr	Global RX Interrupt Status Register n												
			[8:0]	IRN	Ignore Rule Number	0	0	unch	unch	unch	-	-	-	-
			[15: 9]	-	reserved	-	-	-	-	-	-	-	-	-
			[30: 16]	ICN	Ignore Channel Number	0	0	unch	unch	unch	-	-	-	-
			[31: 19]	-	reserved	-	-	-	-	-	-	-	-	-
CFDGAFIIGNCTR	m_acsfr	Global AFL Ignore Control Register												
			0	IREN	Ignore Rule Enable	0	0	unch	unch	unch	-	-	-	-
			[7:1]	-	reserved	-	-	-	-	-	-	-	-	-
			[15: 8]	KEY	These bits control the right or wrong of rewriting of a IREN bit.	0	0	unch	unch	unch	-	-	-	-
			[31: 16]	-	reserved	-	-	-	-	-	-	-	-	-
23. Global Virtual Machine register														
CFDGFFIMC	m_bus_if	Global Virtual Machine Mode configuration Register												
			0	FFIEN	FFI Mode Enable	0	0	unch	unch	unch	-	-	-	-
			[7:1]	-	reserved	-	-	-	-	-	-	-	-	-
			[15: 8]	KEY	These bits control the right or wrong of rewriting of FFIEN bit.	0	0	unch	unch	unch	-	-	-	-
			[31: 16]	-	reserved	-	-	-	-	-	-	-	-	-
CFDGVMES	m_bus_if	Global Virtual Machine Error Interrupt Select Register												

			0	FMLT	Selection of the output destination of FIFO message lost interrupt	0	0	unch	unch	unch	-	-	-
			1	TXQMLT	Selection of the output destination of TXQ message lost interrupt	0	0	unch	unch	unch	-	-	-
			2	THLELT	Selection of the output destination of THL entry lost interrupt	0	0	unch	unch	unch	-	-	-
			3	-	reserved	-	-	-	-	-	-	-	-
			4	CFMOW	Selection of the output destination of COMFIFO message overwrite interrupt	0	0	unch	unch	unch	-	-	-
			5	TXQOW	Selection of the output destination of TXQ message overwrite interrupt	0	0	unch	unch	unch	-	-	-
			[31: 6]	-	reserved	-	-	-	-	-	-	-	-
CFDVMCF Gn	m_bus_if	Global Virtual Machine Common FIFO TXQ configuration Register											
		[3:0]	TXQ0VMN	VM number for TXQ0	0	0	unch	unch	unch	-	-	-	-
		[7:4]	TXQ1VMN	VM number for TXQ1	0	0	unch	unch	unch	-	-	-	-
		[11: 8]	TXQ2VMN	VM number for TXQ2	0	0	unch	unch	unch	-	-	-	-
		[15: 12]	TXQ3VMN	VM number for TXQ3	0	0	unch	unch	unch	-	-	-	-
		[19: 16]	CF0VMN	VM number for COMFIFO 0	0	0	unch	unch	unch	-	-	-	-
		[23: 20]	CF1VMN	VM number for COMFIFO 1	0	0	unch	unch	unch	-	-	-	-
		[27: 24]	CF2VMN	VM number for COMFIFO 2	0	0	unch	unch	unch	-	-	-	-
		[31: 28]	-	reserved	-	-	-	-	-	-	-	-	-
CFDVMRFC FG	m_bus_if	Global Virtual Machine RX FIFO configuration											

		ation Register											
		[3:0]	VMN0	VM number for RXFIFO0	0	0	unch	unch	unch	-	-	-	
		[7:4]	VMN1	VM number for RXFIFO1	0	0	unch	unch	unch	-	-	-	
		[11:8]	VMN2	VM number for RXFIFO2	0	0	unch	unch	unch	-	-	-	
		[15:12]	VMN3	VM number for RXFIFO3	0	0	unch	unch	unch	-	-	-	
		[19:16]	VMN4	VM number for RXFIFO4	0	0	unch	unch	unch	-	-	-	
		[23:20]	VMN5	VM number for RXFIFO5	0	0	unch	unch	unch	-	-	-	
		[27:24]	VMN6	VM number for RXFIFO6	0	0	unch	unch	unch	-	-	-	
		[31:28]	VMN7	VM number for RXFIFO7	0	0	unch	unch	unch	-	-	-	
CFDVMISt Sn	m_bus_if	Virtual Machine Interrupt Status Register											
		0	CFTXINT	COM FIFO TX Interrupt Flag	0	0	unch	unch	unch	-	-	-	
		1	CFOFTXINT	COM FIFO One Frame TX Interrupt Flag	0	0	unch	unch	unch	-	-	-	
		2	TXQTXIF	TXQ TX Interrupt Flag	0	0	unch	unch	unch	-	-	-	
		3	TXQOFTXIF	TXQ One Frame TX Interrupt Flag	0	0	unch	unch	unch	-	-	-	
		4	THLIF	TX History List entry Interrupt Flag	0	0	unch	unch	unch	-	-	-	
		[7:5]	-	reserved	-	-	-	-	-	-	-	-	
		8	RFIF	RXFIFO Interrupt Flag	0	0	unch	unch	unch	-	-	-	
		9	RFFIF	RXFIFO Full Interrupt Flag	0	0	unch	unch	unch	-	-	-	
		10	CFRXINT	COM FIFO RX Interrupt Flag	0	0	unch	unch	unch	-	-	-	
		11	CFOFRXINT	COM FIFO One Frame RX Interrupt Flag	0	0	unch	unch	unch	-	-	-	
		12	CFFIF	COM FIFO Full Interrupt Flag	0	0	unch	unch	unch	-	-	-	
		13	TXQOFRXIF	TXQ One Frame RX Interrupt Flag	0	0	unch	unch	unch	-	-	-	
		14	TXQFIF	TXQ Full Interrupt Flag	0	0	unch	unch	unch	-	-	-	
		15	-	reserved	-	-	-	-	-	-	-	-	

			16	RFMLT	RXFIFO Message Lost	0	0	unch	unch	unch	-	-	-
			17	CFMLT	COMFIFO Message Lost	0	0	unch	unch	unch	-	-	-
			18	CFMOW	COMFIFO Message Overwrite	0	0	unch	unch	unch	-	-	-
			19	TXQMLT	TXQ Message Lost	0	0	unch	unch	unch	-	-	-
			20	TXQMOW	TXQ Message Overwrite	0	0	unch	unch	unch	-	-	-
			21	THLELT	TX History List Entry Lost	0	0	unch	unch	unch	-	-	-
			[31: 22]	-	reserved	-	-	-	-	-	-	-	-
24. Pretended Network Filter (Entry & Control)													
CFDGPFLC CTR	m_pnacs fr	Pretend ed Network Filter List Entry control Register											
			[5:0]	PFLPN	Pretended Network Filter List Page Number	0	0	unch	unch	unch	-	-	-
			[7:6]	-	reserved	-	-	-	-	-	-	-	-
			8	PFLDAE	Pretended Network Filter List Data Access Enable	0	0	unch	unch	unch	-	-	-
			[31: 9]	-	reserved	-	-	-	-	-	-	-	-
CFDGPFLC FG0	m_pnacs fr	Pretend ed Network Filter List Entry Configuration Register 0											
			[5:0]	RNC3	Rule Number for Channel 3	0	0	unch	unch	unch	-	-	-
			[7:6]	-	reserved	-	-	-	-	-	-	-	-
			[13: 8]	RNC2	Rule Number for Channel 2	0	0	unch	unch	unch	-	-	-
			[15: 14]	-	reserved	-	-	-	-	-	-	-	-
			[21: 16]	RNC1	Rule Number for Channel 1	0	0	unch	unch	unch	-	-	-
			[23: 22]	-	reserved	-	-	-	-	-	-	-	-
			[29: 24]	RNC0	Rule Number for Channel 0	0	0	unch	unch	unch	-	-	-
			[31: 30]	-	reserved	-	-	-	-	-	-	-	-
CFDGPFLC FG1	m_pnacs fr	Pretend ed Network Filter List Entry Configuration Register 1											

			[5:0]	RNC7	Rule Number for Channel 7	0	0	unch	unch	unch	-	-	-
			[7:6]	-	reserved	-	-	-	-	-	-	-	-
			[13:8]	RNC6	Rule Number for Channel 6	0	0	unch	unch	unch	-	-	-
			[15:14]	-	reserved	-	-	-	-	-	-	-	-
			[21:16]	RNC5	Rule Number for Channel 5	0	0	unch	unch	unch	-	-	-
			[23:22]		reserved	-	-	-	-	-	-	-	-
			[29:24]	RNC4	Rule Number for Channel 4	0	0	unch	unch	unch	-	-	-
			[31:30]		reserved	-	-	-	-	-	-	-	-

unch : unchanged

Note: After HW Reset, the **CFDGSTS.GRAMINIT** is set to indicate that the RS-CAN-FD module is initialising the RAM.

This bit is cleared automatically when the RAM initialisation is completed and will not set again until HW reset is activated.

5.5 IP operation modes

Following a list of the different IP operation modes is given.

IP Operation mode	Description	Configuration
CAN-FD mode	<p><i>The IP behaves according to ISO 11989-1 (2015)</i></p> <p><i>Classical and CAN-FD frames can be transmitted and received</i></p>	<code>rs_classic_only = 0</code> <code>rs_canfd_tol_off = 0</code> <code>CFDCnFDCFG.CLOE = 0</code> <code>CFDCnFDCFG.FDOE = 0</code>
CAN-FD only mode	<p><i>The IP behaves according to ISO 11989-1 (2015) optional CAN-FD only mode</i></p> <p><i>Only CAN-FD frames can be transmitted and received. Classical frames will cause the detection of an error</i></p>	<code>rs_classic_only = 0</code> <code>rs_canfd_tol_off = 0</code> <code>CFDCnFDCFG.CLOE = 0</code> <code>CFDCnFDCFG.FDOE = 1</code>
Non ISO mode	<i>The IP behaves according to Bosch CAN FD Specification V1.0 only mode</i>	<code>rs_canfd_non_iso_sel = 0</code> <code>rs_canfd_non_iso_en = 1</code> <code>CFDGCRCCFG.NIE = x</code> or <code>rs_canfd_non_iso_sel = 1</code> <code>rs_canfd_non_iso_en = x</code> <code>CFDGCRCCFG.NIE = 1</code> <code>CFDCnFDCFG.CLOE = 1</code> <code>CFDCnFDCFG.CFDTE = 0</code> <code>CFDCnFDCFG.FDOE = 0</code> or <code>rs_classic_only = 1</code> <code>rs_canfd_tol_off = 1</code> <code>CFDCnFDCFG.FDOE = 0</code> or <code>rs_classic_only = 1</code> <code>rs_canfd_tol_off = 0</code> <code>CFDCnFDCFG.CFDTE = 0</code> <code>CFDCnFDCFG.FDOE = 0</code> <code>CFDCnFDCFG.CLOE = 1</code> <code>CFDCnFDCFG.CFDTE = 1</code> <code>CFDCnFDCFG.FDOE = 0</code> or <code>rs_classic_only = 1</code> <code>rs_canfd_tol_off = 0</code> <code>CFDCnFDCFG.CFDTE = 1</code> <code>CFDCnFDCFG.FDOE = 0</code>
Classical CAN only mode	<i>The IP can send only Classical Frame</i> <i>FD frames will cause the detection of an error</i>	<code>CFDCnFDCFG.CLOE = 1</code> <code>CFDCnFDCFG.CFDTE = 0</code> <code>CFDCnFDCFG.FDOE = 0</code> or <code>rs_classic_only = 1</code> <code>rs_canfd_tol_off = 1</code> <code>CFDCnFDCFG.FDOE = 0</code> or <code>rs_classic_only = 1</code> <code>rs_canfd_tol_off = 0</code> <code>CFDCnFDCFG.CFDTE = 0</code> <code>CFDCnFDCFG.FDOE = 0</code> <code>CFDCnFDCFG.CLOE = 1</code> <code>CFDCnFDCFG.CFDTE = 1</code> <code>CFDCnFDCFG.FDOE = 0</code> or <code>rs_classic_only = 1</code> <code>rs_canfd_tol_off = 0</code> <code>CFDCnFDCFG.CFDTE = 1</code> <code>CFDCnFDCFG.FDOE = 0</code>
Classical CAN only mode, FD tolerant mode	<i>The IP can send only Classical Frame</i> <i>FD frames will not cause the detection of an error and will be ignored</i>	<code>CFDCnFDCFG.CLOE = 1</code> <code>CFDCnFDCFG.CFDTE = 0</code> <code>CFDCnFDCFG.FDOE = 0</code> or <code>rs_classic_only = 1</code> <code>rs_canfd_tol_off = 0</code> <code>CFDCnFDCFG.CFDTE = 1</code> <code>CFDCnFDCFG.FDOE = 0</code>

Table 5.5 IP operation modes

5.6 Influence of the register by Parameterization

The register which is subject to the influence of a parameter is shown below.

No	Register Name	Symbol	index	CH 8 6 4 2	TXMB 64 32 16	IDOW ON OFF	AFL 192 128 64	POOL 256 128 64 48 32	comment
				8 6 4 2	64 32 16	ON OFF	192 128 64	256 128 64 48 32	
1	Channel n Nominal Bitrate Configuration Register	CFDCnNCFG	n=0,1						
			n=2,3		x				
			n=4,5	x	x				
			n=6,7	x	x	x			
2	Channel n Control Registers	CFDCnCTR	n=0,1						
			n=2,3		x				
			n=4,5	x	x				
			n=6,7	x	x	x			
3	Channel n Status Registers	CFDCnSTS	n=0,1						
			n=2,3		x				
			n=4,5	x	x				
			n=6,7	x	x	x			
4	Channel n Error Flag Registers	CFDCnERFL	n=0,1						
			n=2,3		x				
			n=4,5	x	x				
			n=6,7	x	x	x			
5	Global IP Version Register	CFDGIPV		□ □ □ □ □	□ □ □ □ □	□ □ □ □ □	□ □ □ □ □	□ □ □ □ □	Value changes by parameter values
6	Global Configuration Register	CFDGCFG							
7	Global Control Register	CFDGCTR							
8	Global Status Register	CFDGSTS							
9	Global Error Flag Register	CFDGERFL		△ △ △					
10	Global Timestamp Counter Register	CFDGTSC							
11	Global Acceptance Filter List Entry Control Register	CFDGAFLCTR							
12	Global Acceptance Filter List Configuration Register w	CFDGAFLCFGw	w=0					□ □	setting range changes.
			w=1		x			□ □	setting range changes.
			w=2	x	x			□ □	setting range changes.
			w=3	x	x	x		□ □	setting range changes.
13	RX Message Buffer Number Register	CFDRMNB							
14	RX Message Buffer New Data Register t	CFDRMNDt	t=0						
			t=1		x				
			t=2	x	x				
			t=3	x	x	x			
15	RX FIFO Configuration / Control Registers a	CFDRFCCa	a=from 0 to 7						
16	RX FIFO Status Registers a	CFDRFSTSa	a=from 0 to 7						
17	RX FIFO Pointer Control Registers a	CFDRFPCTRa	a=from 0 to 7						
18	Common FIFO Configuration / Control Registers d	CFDCFCCd	d=from 0 to 5						
			d=from 6		x				















	Registers 3 [n]		n=2,3			x				△				
			n=4,5			x	x			△				
			n=6,7	x	x	x				△				
49	TX Queue Status Registers 3 [n]	CFDTXQSTS3[n]	n=0,1							△				
			n=2,3			x				△				
			n=4,5		x	x				△				
			n=6,7	x	x	x				△				
50	TX Queue Pointer Control Registers 3 [n]	CFDTXQPCTR3[n]	n=0,1											
			n=2,3			x								
			n=4,5		x	x								
			n=6,7	x	x	x								
51	TX Queue Empty Status Register	CFDTXQUESTS			△	△	△							
52	TX Queue Full Interrupt Status Register	CFDTXQFISTS			△	△	△							
53	TX Queue Message Lost Status Register	CFDTXQMSTS			△	△	△							
54	TX Queue Message Overwrite Status Register	CFDTXQOWSTS			△	△	△			x				
55	TX Queue Interrupt Status Register	CFDTXQISTS			△	△	△							
56	TX Queue One Frame TX Interrupt Status Register	CFDTXQOFTISTS			△	△	△							
57	TX Queue One Frame RX Interrupt Status Register	CFDTXQOFRISTS			△	△	△							
58	TX Queue Full Status Register	CFDTXQFSTS			△	△	△							
59	TX History List Configuration / Control Register n	CFDTHLCCn	n=0,1											
			n=2,3			x								
			n=4,5		x	x								
			n=6,7	x	x	x								
60	TX History List Status Register n	CFDTHLSTS n	n=0,1											
			n=2,3			x								
			n=4,5		x	x								
			n=6,7	x	x	x								
61	TX History List Pointer Control Registers n	CFDTHLPCTR n	n=0,1											
			n=2,3			x								
			n=4,5		x	x								
			n=6,7	x	x	x								
62	Global TX Interrupt Status Register v	CFDGTINTSTS v	v=0				△							
			v=1	△	x	x								
63	Global Test Configuration Register	CFDGTSTCFG			△	△	△			□	□	□	□	□
64	Global Test Control Register	CFDGTSTCTR												
65	Global FD Configuration register	CFDGFD CFG												
66	Global FD CRC Configuration register	CFDGCRCCFG												
67	Global Lock Key Register	CFDGLOCKK												
68	Global OTB FIFO Configuration / Status Register	CFDGLOTB												
69	Global AFL Ignore Entry Register	CFDGAFLIGNENT												
70	Global AFL Ignore Control Register	CFDGAFLIGNCTR												
71	DMA Transfer Control Register	CFDCDTCT			△	△	△							





			n=4,5	x	x								
			n=6,7	x	x	x							
92	Channel n Bus load Control Register	CFDCnBLCT	n=0,1										
			n=2,3			x							
			n=4,5		x	x							
			n=6,7	x	x	x							
93	Channel n Bus load Status Register	CFDCnBLSTS	n=0,1										
			n=2,3			x							
			n=4,5		x	x							
			n=6,7	x	x	x							
94	Global Acceptance Filter List ID Registers r = [1...10]h	CFDGAFLIDr	r=from 1 to 16										
95	Global Acceptance Filter List Mask Registers r = [1...10]h	CFDGAFLMr	r=from 1 to 16										
96	Global Acceptance Filter List Pointer 0 Registers r = [1...10]h	CFDGAFLP0r	r=from 1 to 16										
97	Global Acceptance Filter List Pointer 1 Registers r = [1...10]h	CFDGAFLP1r	r=from 1 to 16										
98	Global Pretended Network Filter List ID Registers s = [1...4]	CFDGPFLIDs	s=from 1 to 4										
99	Global Pretended Network Filter List Mask Registers s = [1...4]	CFDGPFLMs	s=from 1 to 4										
100	Global Pretended Network Filter List Pointer 0 Registers s = [1...4]	CFDGPFLP0s	s=from 1 to 4										
101	Global Pretended Network Filter List Pointer 1 Registers s = [1...4]	CFDGPFLP1s	s=from 1 to 4										
102	Global Pretended Network Filter List Filter Payload Type Registers s = [1...4]	CFDGPFLPTs	s=from 1 to 4										
103	Global Pretended Network Filter List Payload Data 0 Registers s = [1...4]	CFDGPFLPD0s	s=from 1 to 4										
104	Global Pretended Network Filter List Payload Mask 0 Registers s = [1...4]	CFDGPFLPM0s	s=from 1 to 4										
105	Global Pretended Network Filter List Payload Data 1 Registers s = [1...4]	CFDGPFLPD1s	s=from 1 to 4										
106	Global Pretended Network Filter List Payload Mask 1 Registers s = [1...4]	CFDGPFLPM1s	s=from 1 to 4										
107	Channel n TX History List Access Registers 0	CFDTHLACC0[n]	n=0,1										
			n=2,3			xx							
			n=4,5		xx	xx							
			n=6,7	xx	xx	xx							
108	Channel n TX History List Access Registers 1	CFDTHLACC1[n]	n=0,1										
			n=2,3			xx							
			n=4,5		xx	xx							
			n=6,7	xx	xx	xx							
109	RAM Test Page Access Registers k	CFDRPGACCK	k=from 0 to 63										
110	RX Message Buffer ID Registers u	CFDRMID[u]	u=from 0 to 31										
			u=from 32 to 63			xx							
			u=from 64 to 95		xx	xx							
			u=from 96 to 127	xx	xx	xx							























✗ : delete register (can not write / always read 0) When FFI mode is set, read value is unknown

✗✗ : delete register (should not write / read value is unknown)

△ : delete bit (refer to "Influence of the bit" sheet)

□ : etc (refer to comment)

5.7 Influence of the bit by Parameterization

The bit which is subject to the influence of a parameter is shown below.

RS-CAN-FD SFR feature List						CH				TXMB			IDOW		AFL			POOL					
	Module	Register name	Bit	Symbol	Description	8	6	4	2	64	32	16	O N	O F	192	128	64	256	128	64	48	32	
1. Channel Registers																							
CFDCnNCFG	m_chsfr	Channel n Configuration Register																					
			[9:0]	NBRP	Nominal Baudrate Prescaler																		
			[16:10]	NSJW	Nominal Synchronization Jump Width																		
			[24:17]	NTSEG1	Nominal Time Segment 1																		
			[31:25]	NTSEG2	Nominal Time Segment 2																		
CFDCnDCFG	m_chsfr	Channel n Configuration Register																					
			[7:0]	DBRP	Data Baudrate Prescaler																		
			[12:8]	DTSEG1	Data Time Segment 1																		
			[15:13]	-	reserved																		
			[19:16]	DTSEG2	Data Time Segment 2																		
			[23:20]	-	reserved																		
			[27:24]	DSJW	Data Synchronization Jump Width																		
			[31:28]	-	reserved																		
CFDCnCTR	m_chsfr	Channel n Control Register																					
			[1:0]	CHMDC	Channel Mode Control																		
			2	CSLPR	Channel Sleep Request																		
			3	RTBO	Return from Bus-Off																		
			[7:4]	-	reserved																		
			8	BEIE	Bus Error Interrupt Enable																		
			9	EWIE	Error Warning Interrupt Enable																		









		0	TPRI	Transmissio n Priority																		
		1	DCE	DLC Check Enable																		
		2	DRE	DLC Replaceme nt Enable																		
		3	MME	Mirror Mode Enable																		
		4	DCS	PLL By- Pass																		
		5	CMPOC	CAN-FD message Payload overflow configuratio n																		
		[7:6]	-	reserved																		
		[11:8]	TSP	Timestamp Prescaler																		
		12	TSSS	Timestamp Source Select																		
		[15:1] [3]	TSBTCS	Timestamp Bit Time Channel Select																		
		[31:1] [6]	ITRCP	Interval Timer Reference Clock Prescaler																		
CFDGCTR	m_comsfr	Global Control Register																				
		[1:0]	GMDC	Global Mode Control																		
		2	GSLPR	Global Sleep Request																		
		[7:3]	-	reserved																		
		8	DEIE	DLC Error Interrupt Enable																		
		9	MEIE	Message Lost Error Interrupt Enable																		
		10	THLEIE	TX History List Entry Lost Interrupt Enable																		
		11	CMPOFIE	CAN-FD message payload overflow Flag Interrupt enable																		
		12	QOWEIE	TXQ Message overwrite Error Interrupt Enable												xx						
		13	-	reserved																		
		14	QMEIE	TXQ Message lost Error Interrupt Enable																		

			15	MOWEIE	Message overwrite Error Interrupt Enable																				
			16	TSRST	TS Reset																				
			17	TSWR	Timestamp Write																				
			[31:1 8]	-	reserved																				
CFDGFDCFG	m_comsfr	Global FD configura tion register																							
			0	RPED	Protocol exception state disable																				
			[7:1]	-	reserved																				
			[9:8]	TSCCFG	Time stamp capture configuratio n																				
			[31:1 0]	-	reserved																				
CFDGCRCCFG	m_comsfr	Global CRC configura tion register																							
			0	NIE	Non ISO enable																				
			[31:1]	-	reserved																				
4. Global Status Registers																									
CFDGSTS	m_comsfr	Global Status Register																							
			0	GRSTSTS	Global RESET Status																				
			1	GHLTSTS	Global HALT Status																				
			2	GSLPSTS	Global SLEEP Status																				
			3	GRAMINIT	Global RAM Initialisation Status																				
			[31:4]	-	reserved																				
CFDGERFL	m_comsfr	Global Error Flag Register																							
			0	DEF	DLC Error Flag																				
			1	MES	Message Lost Error Status																				
			2	THLES	TX History List Entry Lost Error Status																				
			3	CMPOF	CAN-FD message payload overflow Flag																				

		4	QOWES	TXQ Message overwrite Error Status							x									
		5	OTBMLTSTS	OTB FIFO Msg Lost Status																
		6	QMES	TXQ Message Lost Error Status																
		7	MOWES	Message overwrite Error Status																
		8	RXSFAIL0	RX SCAN Fail Channel 0																
		9	RXSFAIL1	RX SCAN Fail Channel 1																
		10	RXSFAIL2	RX SCAN Fail Channel 2				x												
		11	RXSFAIL3	RX SCAN Fail Channel 3					x											
		12	RXSFAIL4	RX SCAN Fail Channel 4			x	x												
		13	RXSFAIL5	RX SCAN Fail Channel 5				x	x											
		14	RXSFAIL6	RX SCAN Fail Channel 6		x	x	x												
		15	RXSFAIL7	RX SCAN Fail Channel 7		x	x	x												
		16	EEF0	ECC Error Flag for Channel 0																
		17	EEF1	ECC Error Flag for Channel 1																
		18	EEF2	ECC Error Flag for Channel 2				x												
		19	EEF3	ECC Error Flag for Channel 3					x											
		20	EEF4	ECC Error Flag for Channel 4			x	x												
		21	EEF5	ECC Error Flag for Channel 5			x	x												
		22	EEF6	ECC Error Flag for Channel 6		x	x	x												
		23	EEF7	ECC Error Flag for Channel 7		x	x	x												
		[31:2 4]	-	reserved																
CFDGTSC	m_timestamp	Global Timestamp Counter Register																		
		[15:0]	TS	Timestamp Value																
		[31:1 6]	-	reserved																

5. Global Acceptance Filter List Configuration Registers																				
CFDGAFLEC TR	m_acsfr	Global Acceptance Filter List Entry Control Register																		
			[6:0]	AFLPN	Acceptance Filter List Page Number															
			7	-	reserved															
			8	AFLDAE	Acceptance Filter List Access Enable Data															
			[31:9]	-	reserved															
CFDGALCF G0																				
			[8:0]	RNC1	Rule Number for Channel 1															
			[15:9]	-	reserved															
			[24:16]	RNC0	Rule Number for Channel 0															
			[31:25]	-	reserved															
CFDGALCF G1																				
			[8:0]	RNC3	Rule Number for Channel 3				x											
			[15:9]	-	reserved															
			[24:16]	RNC2	Rule Number for Channel 2			x												
			[31:25]	-	reserved															
CFDGALCF G2																				
			[8:0]	RNC5	Rule Number for Channel 5		x	x												
			[15:9]	-	reserved															
			[24:16]	RNC4	Rule Number for Channel 4		x	x												
			[31:25]	-	reserved															

CFDGAFLCF G3	m_acsfr	Global Acceptance Filter List Configuration 3 Register																	
		[8:0]	RNC7	Rule Number for Channel 7	x	x	x												
		[15:9]	-	reserved															
		[24:16]	RNC6	Rule Number for Channel 6	x	x	x												
		[31:25]	-	reserved															
6. RX Mailbox Registers																			
CFDRMNB	m_acsfr	RX Mailbox Number Register																	
		[7:0]	NRXMB	Number of RX MB															
		[10:8]	RMPLS	Reception Message Buffer Payload Data Size															
		[31:11]	-	reserved															
CFDRMNDt	m_acsfr	RX-Mailbox NewData Registers																	
		[31:0]	RMNS	RX Mailboxes Newdata Flag															
7. RX FIFO Registers																			
CFDRFCCa	m_rxifounit	RX FIFO Configuration / Control Registers [7:0]																	
		0	RFE	RX FIFO Enable															
		1	RFIE	RX FIFO Interrupt Enable															
		[3:2]	-	reserved															
		[6:4]	RFPLS	Rx FIFO Payload Data Size configuration															
		7	-	reserved															
		[10:8]	RFDC	RX FIFO Depth Configuration															
		11	-	reserved															
		12	RFIM	RX FIFO Interrupt Mode															
		[15:13]	RFIGCV	RX FIFO Interrupt Generation Counter Value															

			16	RFFIE	RX FIFO Full interrupt Enable																	
			[23:1 7]	-	reserved																	
			24	MEIE	Message lost Error Interrupt Enable for FFI Mode																	
			[27:2 5]	-	reserved																	
			28	DMAE	DMA Transfer Enable for RXFIFO a for FFI Mode																	
			[31:2 9]	-	reserved																	
CFDRFSTSa	m_rxifounit	RX FIFO Status Registers [7:0]																				
			0	RFEMP	RX FIFO Empty																	
			1	RFFLL	RX FIFO Full																	
			2	RFMLT	RX FIFO Msg Lost Flag																	
			3	RFIF	RX FIFO Interrupt Flag																	
			[7:4]	-	reserved																	
			[15:8]	RFMC	RX FIFO Message Count																	
			16	RFFIF	RX FIFO Full Interrupt Flag																	
			[27:1 7]	-	reserved																	
			28	RFDMASTS	DMA Transfer Status for RX FIFO a for FFI Mode																	
			[31:2 9]	-	reserved																	
CFDRFPCTRa	m_rxifounit	RX FIFO Pointer Control Register[7:0]																				
			[7:0]	RPC	RX FIFO Pointer Control																	
			[31:8]	-	reserved																	
8. Common FIFO Registers																						
CFDCFC	m_comfifo	Common FIFO Configuration / Control Registers [23:0]																				
			0	CFE	Common																	







CFDFFFSTS	m_acsfr	FIFO FDC level Full Status Register																								
			[7:0]	RFxFFLL	RX FIFO FDC level full Status																					
			[13:8]	CFxFFLL	COMMON FIFO FDC level full Status																					
			[19:1]	CFxFFLL	COMMON FIFO FDC level full Status			x																		
			[25:2]	CFxFFLL	COMMON FIFO FDC level full Status			x	x																	
			[31:2]	CFxFFLL	COMMON FIFO FDC level full Status		x	x	x																	
CFDFMSTS	m_acsfr	FIFO Msg Lost Status Register																								
			[7:0]	RFxMLT	RX FIFO Msg Lost Status																					
			[13:8]	CFxMLT	Common FIFO Msg Lost Status																					
			[19:1]	CFxMLT	Common FIFO Msg Lost Status			x																		
			[25:2]	CFxMLT	Common FIFO Msg Lost Status			x	x																	
			[31:2]	CFxMLT	Common FIFO Msg Lost Status		x	x	x																	
CFDCFOWSTS	m_acsfr	Common FIFO Message OverWrite Status																								
			[5:0]	CFxMOW	Common FIFO [x] Message overwrite status																					
			[11:6]	CFxMOW	Common FIFO [x] Message overwrite status			x																		
			[17:1]	CFxMOW	Common FIFO [x] Message overwrite status			x	x																	
			[23:1]	CFxMOW	Common FIFO [x] Message overwrite status		x	x	x																	
			[31:2]	-	reserved																					
CFDRFISTS	m_acsfr	RX FIFO Interrupt Flag Status																								



		Status																							
		[5:0]	CFxOFRXIF	Common FIFO [x] One Frame RX Interrupt Flag Status																					
		[11:6]	CFxOFRXIF	Common FIFO [x] One Frame RX Interrupt Flag Status				x																	
		[17:12]	CFxOFRXIF	Common FIFO [x] One Frame RX Interrupt Flag Status			x	x																	
		[23:18]	CFxOFRXIF	Common FIFO [x] One Frame RX Interrupt Flag Status		x	x	x																	
		[31:24]	-	reserved																					
CFDCFOFTITS	m_acsfr	Common FIFO One Frame TX Interrupt Flag Status																							
		[5:0]	CFxOFTXIF	Common FIFO [x] One Frame TX Interrupt Flag Status																					
		[11:6]	CFxOFTXIF	Common FIFO [x] One Frame TX Interrupt Flag Status				x																	
		[17:12]	CFxOFTXIF	Common FIFO [x] One Frame TX Interrupt Flag Status			x	x																	
		[23:18]	CFxOFTXIF	Common FIFO [x] One Frame TX Interrupt Flag Status		x	x	x																	
		[31:24]	-	reserved																					
10. Special CAN-FD register																									
CFDCDTCT	m_dmaif	DMA Transfer Control Register																							
		[7:0]	RFDMAEe	DMA Transfer Enable for RXFIFO e																					
		8	CFDMAE0	DMA Transfer Enable for Common FIFO 0																					
		9	CFDMAE1	DMA Transfer Enable for Common FIFO 1																					
		10	CFDMAE2	DMA Transfer Enable for Common FIFO 2			x																		

		11	CFDMAE3	DMA Transfer Enable for Common FIFO 3			x														
		12	CFDMAE4	DMA Transfer Enable for Common FIFO 4			x	x													
		13	CFDMAE5	DMA Transfer Enable for Common FIFO 5			x	x													
		14	CFDMAE6	DMA Transfer Enable for Common FIFO 6		x	x	x													
		15	CFDMAE7	DMA Transfer Enable for Common FIFO 7		x	x	x													
		[31:16]	-	reserved																	
CFDCDTSTS	m_dmaif	DMA Transfer Status Register																			
		[7:0]	RFDMASTSe	DMA Transfer Status for RX FIFO e																	
		8	CFDMASTS0	DMA Transfer Status only for Common FIFO 0 of channel 0																	
		9	CFDMASTS1	DMA Transfer Status only for Common FIFO 0 of channel 1																	
		10	CFDMASTS2	DMA Transfer Status only for Common FIFO 0 of channel 2				x													
		11	CFDMASTS3	DMA Transfer Status only for Common FIFO 0 of channel 3			x														
		12	CFDMASTS4	DMA Transfer Status only for Common FIFO 0 of channel 4			x	x													
		13	CFDMASTS5	DMA Transfer Status only for Common FIFO 0 of channel 5			x	x													
		14	CFDMASTS6	DMA Transfer Status only for Common		x	x	x													

					FIFO 0 of channel 6																		
			15	CFDMASTS7	DMA Transfer Status only for Common FIFO 0 of channel 7		x	x	x														
			[31:1 6]	-	reserved																		
CFDCDTTCT	m_dmaif	DMA TX Transfer Control Register																					
			0	TQ0DMAE0	DMA TX Transfer Enable for TXQ0 0																		
			1	TQ0DMAE1	DMA TX Transfer Enable for TXQ0 1																		
			2	TQ0DMAE2	DMA TX Transfer Enable for TXQ0 2				x														
			3	TQ0DMAE3	DMA TX Transfer Enable for TXQ0 3				x														
			4	TQ0DMAE4	DMA TX Transfer Enable for TXQ0 4			x	x														
			5	TQ0DMAE5	DMA TX Transfer Enable for TXQ0 5			x	x														
			6	TQ0DMAE6	DMA TX Transfer Enable for TXQ0 6		x	x	x														
			7	TQ0DMAE7	DMA TX Transfer Enable for TXQ0 7		x	x	x														
			8	TQ3DMAE0	DMA TX Transfer Enable for TXQ3 0																		
			9	TQ3DMAE1	DMA TX Transfer Enable for TXQ3 1																		
			10	TQ3DMAE2	DMA TX Transfer Enable for TXQ3 2				x														
			11	TQ3DMAE3	DMA TX Transfer Enable for TXQ3 3				x														
			12	TQ3DMAE4	DMA TX Transfer Enable for TXQ3 4			x	x														
			13	TQ3DMAE5	DMA TX Transfer Enable for TXQ3 5			x	x														
			14	TQ3DMAE6	DMA TX Transfer Enable for TXQ3 6		x	x	x														
			15	TQ3DMAE7	DMA TX Transfer		x	x	x														

					Enable for TXQ3 7																		
			16	CFDMAE0	DMA Transfer Enable for Common FIFO 0																		
			17	CFDMAE1	DMA Transfer Enable for Common FIFO 1																		
			18	CFDMAE2	DMA Transfer Enable for Common FIFO 2				x														
			19	CFDMAE3	DMA Transfer Enable for Common FIFO 3				x														
			20	CFDMAE4	DMA Transfer Enable for Common FIFO 4			x	x														
			21	CFDMAE5	DMA Transfer Enable for Common FIFO 5			x	x														
			22	CFDMAE6	DMA Transfer Enable for Common FIFO 6		x	x	x														
			23	CFDMAE7	DMA Transfer Enable for Common FIFO 7		x	x	x														
		[31:24]	-	reserved																			
CFDCDTTSTS	m_dmaif	DMA TX Transfer Status Register																					
		0	TQ0DMASTS0	DMA Transfer Status for TXQ0 0																			
		1	TQ0DMASTS1	DMA Transfer Status for TXQ0 1																			
		2	TQ0DMASTS2	DMA Transfer Status for TXQ0 2				x															
		3	TQ0DMASTS3	DMA Transfer Status for TXQ0 3				x															
		4	TQ0DMASTS4	DMA Transfer Status for TXQ0 4				x	x														
		5	TQ0DMASTS5	DMA Transfer Status for TXQ0 5				x	x														
		6	TQ0DMASTS6	DMA Transfer Status for TXQ0 6		x	x	x															
		7	TQ0DMASTS7	DMA Transfer		x	x	x															



					channel 6																		
			23	CFDMASTS7	DMA Transfer Status only for Common FIFO 0 of channel 7		x	x	x														
			[31:24]	-	reserved																		
11. TX Mailbox Registers																							
CFDTMCi	m_mbctrl	TX Mailbox Control Registers i																					
			0	TMTR	TX Mailbox Transmission Request																		
			1	TMTAR	TX Mailbox Transmission Abortion Request																		
			2	TMOM	TX Mailbox One-shot mode																		
			[7:3]	-	reserved																		
CFDTMSTSj	m_mbctrl	TX Mailbox Status Registers j																					
			0	TMTSTS	TX Mailbox Transmission Status																		
			[2:1]	TMTRF	TX Mailbox Transmission Result Flag																		
			3	TMTRM	TX Mailbox Transmission Request																		
			4	TMTARM	TX Mailbox Transmission Abortion Request																		
			[7:5]	-	reserved																		
CFDTMTRSTSf	m_mbctrl	TX Mailboxes Transmission Request Status Registers																					
			[7:0]	CFDTMTRSTS	TX Mailbox Transmission Request Status																		
			[15:8]	CFDTMTRSTS	TX Mailbox Transmission Request Status											x							
			[23:16]	CFDTMTRSTS	TX Mailbox Transmission Request Status									x	x								
			[31:24]	CFDTMTRSTS	TX Mailbox Transmission Request Status						x	x											





					Interrupt Enable for FFI Mode																			
			[27:26]	-	reserved																			
			28	DMAE	DMA TX Transfer Enable for TXQ for FFI Mode																			
			[31:29]	-	reserved																			
CFDTXQCC1n	m_mbctrl	TX Queue1 Configuration / Control Register n																						
			0	TXQE	TX Queue Enable																			
			1	TXQGWE	TX Queue Gateway Mode Enable																			
			2	TXQOWE	TX Queue Overwrite Mode Enable											x								
			[4:3]	-	reserved																			
			5	TXQTXIE	TX Queue TX Interrupt Enable																			
			6	-	reserved																			
			7	TXQIM	TX Queue Interrupt Mode																			
			[12:8]	TXQDC	TX Queue Depth Configuration																			
			[15:13]	-	reserved																			
			16	TXQFIE	TXQ Full interrupt Enable																			
			17	TXQOFRXIE	TXQ One Frame Reception Interrupt Enable																			
			18	TXQOFTXIE	TXQ One Frame Transmission Interrupt Enable																			
			[23:19]	-	reserved																			
			24	QMEIE	TXQ Message lost Error Interrupt Enable for FFI Mode																			
			25	QOWEIE	TXQ Message overwrite Error Interrupt Enable for FFI Mode										x									
			[31:26]	-	reserved																			



		5	TXQTXIE	TX Queue TX Interrupt Enable																		
		6	-	reserved																		
		7	TXQIM	TX Queue Interrupt Mode																		
		[12:8]	TXQDC	TX Queue Depth Configuration																		
		[17:1]	-	reserved																		
		18	TXQOFTXIE	TXQ One Frame Transmission Interrupt Enable																		
		[31:19]	-	reserved																		
		[24:19]	-	reserved																		
		25	QOWEIE	TXQ Message overwrite Error Interrupt Enable for FFI Mode												x						
		[27:26]	-	reserved																		
		28	DMAE	DMA TX Transfer Enable for TXQ for FFI Mode																		
		[31:29]	-	reserved																		
CFDTXQSTS0n	m_mbctrl	TX Queue0 Status Register n																				
		0	TXQEMP	TX Queue Empty																		
		1	TXQFLL	TX Queue Full																		
		2	TXQTXIF	TX Queue TX Interrupt Flag																		
		[7:3]	-	reserved																		
		[13:8]	TXQMC	CHn TX Queue Message counter																		
		[15:14]	-	reserved																		
		16	TXQFIF	TXQ Full Interrupt Flag																		
		17	TXQOFRXIF	TXQ One Frame Reception Interrupt Flag																		
		18	TXQOFTXIF	TXQ One Frame Transmission Interrupt Flag																		
		19	TXQMLT	TXQ Message Lost																		

		20	TXQMOW	TXQ message overwrite						x					
		[27:2] - [1]		reserved											
		28	TXQDMASTS	DMA TX Transfer Status for TXQ0											
		[31:2] - [9]		reserved											
CFDTXQSTS 1n	m_mbctrl	TX Queue1 Status Register n													
		0	TXQEMP	TX Queue Empty											
		1	TXQFLL	TX Queue Full											
		2	TXQTXIF	TX Queue TX Interrupt Flag											
		[7:3] -		reserved											
		[13:8] -	TXQMC	CHn TX Queue Message counter											
		[15:1] - [4]		reserved											
		16	TXQFIF	TXQ Full Interrupt Flag											
		17	TXQOFRXIF	TXQ One Frame Reception Interrupt Flag											
		18	TXQOFTXIF	TXQ One Frame Transmission Interrupt Flag											
		19	TXQMLT	TXQ Message Lost											
		20	TXQMOW	TXQ message overwrite						x					
		[31:2] - [1]		reserved											
CFDTXQSTS 2n	m_mbctrl	TX Queue2 Status Register n													
		0	TXQEMP	TX Queue Empty											
		1	TXQFLL	TX Queue Full											
		2	TXQTXIF	TX Queue TX Interrupt Flag											
		[7:3] -		reserved											
		[13:8] -	TXQMC	CHn TX Queue Message counter											
		[15:1] - [4]		reserved											
		16	TXQFIF	TXQ Full Interrupt											



			[7:0]	TXQPC	Channel n TX Queue Pointer Control																			
			[31:8]]	-	reserved																			
CFDTXQPCT R2n	m_mbctrl	TX Queue2 Pointer Control Register n																						
			[7:0]	TXQPC	Channel n TX Queue Pointer Control																			
			[31:8]]	-	reserved																			
CFDTXQPCT R3n	m_mbctrl	TX Queue3 Pointer Control Register n																						
			[7:0]	TXQPC	Channel n TX Queue Pointer Control																			
			[31:8]]	-	reserved																			
CFDTXQEST S	m_mbctrl	TX Queue Empty Status Registers																						
			[7:0]	TXQEMP	TXQ empty Status																			
			[15:8]]	TXQEMP	TXQ empty Status				x															
			[23:1] 6]	TXQEMP	TXQ empty Status		x	x																
			[31:2] 4]	TXQEMP	TXQ empty Status	x	x	x																
CFDTXQFIST S	m_mbctrl	TX Queue Full Interrupt Status Registers																						
			[2:0]	TXQFULL[2:0]	TXQ Full Interrupt Status																			
			3	-	reserved																			
			[6:4]	TXQFULL[6:4]	TXQ Full Interrupt Status																			
			7	-	reserved																			
			[10:8]]	TXQFULL[10:8]	TXQ Full Interrupt Status		x																	
			11	-	reserved																			
			[14:1] 2]	TXQFULL[14:1]	TXQ Full Interrupt Status		x																	
			15	-	reserved																			
			[18:1] 6]	TXQFULL[18:1]	TXQ Full Interrupt Status	x	x																	
			19	-	reserved																			



			[22:20]	TXQFULL[22:20]	TXQ Full Interrupt Status		x	x											
			23	-	reserved														
			[26:24]	TXQFULL[26:24]	TXQ Full Interrupt Status		x	x	x										
			27	-	reserved														
			[30:28]	TXQFULL[30:28]	TXQ Full Interrupt Status		x	x	x										
			31	-	reserved														
<hr/>																			
CFDTXQMSTS	m_mbctrl	TX Queue Message lost Status Registers																	
			[2:0]	TXQML[2:0]	TXQ message lost Status														
			3	-	reserved														
			[6:4]	TXQML[6:4]	TXQ message lost Status														
			7	-	reserved														
			[10:8]	TXQML[10:8]	TXQ message lost Status				x										
			11	-	reserved														
			[14:12]	TXQML[14:12]	TXQ message lost Status			x											
			15	-	reserved														
			[18:16]	TXQML[18:16]	TXQ message lost Status		x	x											
			19	-	reserved														
			[22:20]	TXQML[22:20]	TXQ message lost Status		x	x											
			23	-	reserved														
			[26:24]	TXQML[26:24]	TXQ message lost Status		x	x	x										
			27	-	reserved														
			[30:28]	TXQML[30:28]	TXQ Full Interrupt Status		x	x	x										
			31	-	reserved														
<hr/>																			
CFDTXQOWSTS	m_mbctrl	TX Queue Message Overwrite Status Registers														x			
			[7:0]	TXQOW	TXQ message overwrite Status														
			[15:8]	TXQOW	TXQ message overwrite Status			x								x			
			[23:16]	TXQOW	TXQ message overwrite Status		x	x								x			
			[31:2]	TXQOW	TXQ message		x	x	x							x			

			4]		overwrite Status																				
CFDTXQISTS	m_mbctrl	TX Queue Interrupt Status Registers																							
			[7:0]	TXQISF	TXQ Interrupt Status Flag																				
			[15:8]	TXQISF	TXQ Interrupt Status Flag			x																	
			[23:16]	TXQISF	TXQ Interrupt Status Flag			x	x	x															
			[31:4]	TXQISF	TXQ Interrupt Status Flag		x	x	x																
CFDTXQOFTI STS	m_mbctrl	TX Queue One Frame TX Interrupt Status Registers																							
			[7:0]	TXQOFTISF	TXQ One Frame Tx Interrupt Status Flag																				
			[15:8]	TXQOFTISF	TXQ One Frame Tx Interrupt Status Flag			x																	
			[23:16]	TXQOFTISF	TXQ One Frame Tx Interrupt Status Flag			x	x	x															
			[31:4]	TXQOFTISF	TXQ One Frame Tx Interrupt Status Flag		x	x	x																
CFDTXQOFRIST S	m_mbctrl	TX Queue One Frame RX Interrupt Status Registers																							
			[2:0]	TXQOFRISF[2:0]	TXQ One Frame RX Interrupt Status Flag																				
			3	-	reserved																				
			[6:4]	TXQOFRISF[6:4]	TXQ One Frame RX Interrupt Status Flag																				
			7	-	reserved																				
			[10:8]	TXQOFRISF[10:8]	TXQ One Frame RX Interrupt Status Flag			x																	
			11	-	reserved																				
			[14:12]	TXQOFRISF[14:12]	TXQ One Frame RX Interrupt Status Flag			x																	
			15	-	reserved																				
			[18:16]	TXQOFRISF[18:16]	TXQ One Frame RX Interrupt Status Flag			x	x																
			19	-	reserved																				

			[22:20]	TXQOFRISF[22:20]	TXQ One Frame RX Interrupt Status Flag			x	x														
			23	-	reserved																		
			[26:24]	TXQOFRISF[26:24]	TXQ One Frame RX Interrupt Status Flag		x	x	x														
			27	-	reserved																		
			[30:28]	TXQOFRISF[30:28]	TXQ One Frame RX Interrupt Status Flag		x	x	x														
			31	-	reserved																		
CFDTXQFSTS	m_mbctrl	TX Queue Full Status Registers																					
			[7:0]	TXQFSF	TXQ Full Status																		
			[15:8]	TXQFSF	TXQ Full Status				x														
			[23:16]	TXQFSF	TXQ Full Status			x	x														
			[31:24]	TXQFSF	TXQ Full Status	x	x	x															
13. TX History List Registers																							
CFDTHLCn	m_thlififou nit	Channel n TX History List Configuration / Control Register																					
			0	THLE	TX History List Enable																		
			[7:1]	-	reserved																		
			8	THLIE	TX History List Interrupt Enable																		
			9	THLIM	TX History List Interrupt Mode																		
			10	THLDTE	TX History List Dedicated TX Enable																		
			11	THLDGE	TX History List Dedicated GW Enable																		
			[23:12]	-	reserved																		
			24	THLEIE	TX History List Entry Lost Interrupt Enable																		
			[31:25]	-	reserved																		
CFDHLSTS	m_thlififou nit	Channel n TX History List Status Register																					
			0	THLEMP	TX History																		

					List Empty										
		1	THLFLL	TX History List Full											
		2	THLELT	TX History List Entry Lost Flag											
		3	THLIF	TX History List Interrupt Flag											
		[7:4]	-	reserved											
		[13:8]	THLMC	TX History List Message Count											
		[31:14]	-	reserved											
CFDTHLPCTR	m_thififou nit	CHn TX History List pointer control register													
		[7:0]	THLPC	TX History List Pointer Control											
		[31:8]	-	reserved											
14. TX Interrupt Status Register															
CFDGTINTSTS0	m_interrupt	Global Interrupt Status Register 0													
		0	TSIFO	TX Successful Transmission Interrupt Flag Channel 0											
		1	TAIF0	TX Abortion Interrupt Flag Channel 0											
		2	TQIF0	TX Queue Interrupt Flag Channel 0											
		3	CFTIF0	COM FIFO TX/GW Mode Interrupt Flag Channel 0											
		4	THIF0	TX History List Interrupt Channel 0											
		5	TQOFIF0	TX Queue One Frame Transmission Interrupt Flag Channel 0											
		6	CFOTIF0	COM FIFO One Frame Transmission Interrupt Channel 0											
		7	-	reserved											
		8	TSIF1	TX Successful Transmission Interrupt Flag											

				Channel 1															
		9	TAIF1	TX Abortion Interrupt Flag Channel 1															
		10	TQIF1	TX Queue Interrupt Flag Channel 1															
		11	CFTIF1	COM FIFO TX/GW Mode Interrupt Flag Channel 1															
		12	THIF1	TX History List Interrupt Channel 1															
		13	TQOFIG1	TX Queue One Frame Transmission Interrupt Flag Channel 1															
		14	CFOTIF1	COM FIFO One Frame Transmission Interrupt Channel 1															
		15	-	reserved															
		16	TSIF2	TX Successful Transmission Interrupt Flag Channel 2			x												
		17	TAIF2	TX Abortion Interrupt Flag Channel 2			x												
		18	TQIF2	TX Queue Interrupt Flag Channel 2			x												
		19	CFTIF2	COM FIFO TX/GW Mode Interrupt Flag Channel 2			x												
		20	THIF2	TX History List Interrupt Channel 2			x												
		21	TQOFIG2	TX Queue One Frame Transmission Interrupt Flag Channel 2			x												
		22	CFOTIF2	COM FIFO One Frame Transmission Interrupt Channel 2			x												
		23	-	reserved															
		24	TSIF3	TX Successful Transmission Interrupt Flag Channel 3			x												
		25	TAIF3	TX Abortion Interrupt Flag Channel 3			x												
		26	TQIF3	TX Queue Interrupt Flag			x												

					Channel 3																				
		27	CFTIF3	COM FIFO TX/GW Mode Interrupt Flag Channel 3			x																		
		28	THIF3	TX History List Interrupt Channel 3			x																		
		29	TQOFIF3	TX Queue One Frame Transmissio n Interrupt Flag Channel 3			x																		
		30	CFOTIF3	COM FIFO One Frame Transmissio n Interrupt Channel 3			x																		
		31	-	reserved																					
CFDGTINTST S1	m_interrupt	Global Interrupt Status Register 1																							
		0	TSIF4	TX Successful Transmission Interrupt Flag Channel 4			x	x																	
		1	TAIF4	TX Abortion Interrupt Flag Channel 4			x	x																	
		2	TQIF4	TX Queue Interrupt Flag Channel 4			x	x																	
		3	CFTIF4	COM FIFO TX/GW Mode Interrupt Flag Channel 4			x	x																	
		4	THIF4	TX History List Interrupt Channel 4			x	x																	
		5	TQOFIF4	TX Queue One Frame Transmissio n Interrupt Flag Channel 4			x	x																	
		6	CFOTIF4	COM FIFO One Frame Transmissio n Interrupt Channel 4			x	x																	
		7	-	reserved																					
		8	TSIF5	TX Successful Transmission Interrupt Flag Channel 5			x	x																	
		9	TAIF5	TX Abortion Interrupt Flag Channel 5			x	x																	
		10	TQIF5	TX Queue Interrupt Flag Channel 5			x	x																	



					Channel 7																		
		29	TQOFIF7	TX Queue One Frame Transmission Interrupt Flag Channel 7		x	x	x															
		30	CFOTIF7	COM FIFO One Frame Transmission Interrupt Channel 7		x	x	x															
		31	-	reserved																			
15. Global Test / Evaluation / Diag. Control Register																							
CFDGTSFCFG	m_test	Global Test Configuration Register																					
		0	C0ICBCE	Channel 0 Internal CAN Bus Communication Test Mode Enable																			
		1	C1ICBCE	Channel 1 Internal CAN Bus Communication Test Mode Enable																			
		2	C2ICBCE	Channel 2 Internal CAN Bus Communication Test Mode Enable				x															
		3	C3ICBCE	Channel 3 Internal CAN Bus Communication Test Mode Enable				x															
		4	C4ICBCE	Channel 4 Internal CAN Bus Communication Test Mode Enable			x	x															
		5	C5ICBCE	Channel 5 Internal CAN Bus Communication Test Mode Enable			x	x															
		6	C6ICBCE	Channel 6 Internal CAN Bus Communication Test Mode Enable		x	x	x															
		7	C7ICBCE	Channel 7 Internal CAN Bus Communication Test Mode Enable		x	x	x															
		[15:8]	-	reserved																			
		[25:16]	RTMPS	RAM Test Mode Page Select																			



		[]													
		8	BLCLD	BUS load counter load											
		[31:9] []	-	reserved											
17. Flexible CAN mode Register															
CFDGFCMC	m_comsfr	Global Flexible CAN mode Configuration Register													
		0	FLXC0	Flexible CAN mode between Channel 0 and Channel 1											
		1	FLXC1	Flexible CAN mode between Channel 2 and Channel 3			x								
		2	FLXC2	Flexible CAN mode between Channel 4 and Channel 5		x	x	x							
		3	FLXC3	Flexible CAN mode between Channel 6 and Channel 7	x	x	x								
		[31:4] []	-	reserved											
18. Flexible transmission buffer assignment															
CFDGFTBAC	m_bus_if	Global Flexible transmission buffer assignment Configuration Register													
		[3:0]	FLXMB0	Flexible transmission buffer assignment between Channel 0 and Channel 1											
		[7:4]	-	reserved											
		[11:8] []	FLXMB1	Flexible transmission buffer assignment between Channel 2 and Channel 3			x								
		[15:1] [2]	-	reserved											
		[19:1] [6]	FLXMB2	Flexible transmission buffer assignment between Channel 4 and Channel 5		x	x								
		[23:2]	-	reserved											

			[0]																					
			[27:24]	FLXMB3	Flexible transmission buffer assignment between Channel 6 and Channel 7		x	x	x															
			[31:28]	-	reserved																			
19. BUS Interface Register																								
CFDGBISC	m_bus_if	Global Bus Interface Select Configuration Register																						
0	0	0	0	IFSW	Select BUS Interface APB or R-ACE																			
0	0	0	[7:1]	-	reserved																			
0	0	0	[15:8]	KEY	Key code																			
0	0	0	[31:16]	-	reserved																			
20. Reset Control Register																								
CFDGRSTC	m_bus_if	Global SW reset Register																						
		0	SRST	SW reset																				
		[7:1]	-	reserved																				
		[15:8]	KEY	Key code																				
		[31:16]	-	reserved																				
21. RX Interrupt Status Register (COMMON FIFO RX-GW mode & TXQ GW mode)																								
CFDGRINTST	m_interrupt	Global RX Interrupt Status Register n																						
		[2:0]	QFIF	TXQ Full Interrupt Flag Channel n																				
		[3]	-	reserved																				
		[5:4]	BQFIF	Borrowed TXQ Full Interrupt Flag Channel n																				
		[7:6]	-	reserved																				
		[10:8]	QOFRIF	TXQ One Frame RX Interrupt Flag Channel n																				
		[11]	-	reserved																				
		[13:12]	BQOFRIF	Borrowed TXQ One Frame RX Interrupt Flag Channel n																				
		[15:14]	-	reserved																				
		[18:16]	CFRIF	Common FIFO RX Interrupt																				







					Interrupt Flag														
			14	TXQFIF	TXQ Full Interrupt Flag														
			15	-	reserved														
			16	RFMLT	RXFIFO Message Lost														
			17	CFMLT	COMFIFO Message Lost														
			18	CFMOW	COMFIFO Message Overwrite														
			19	TXQMLT	TXQ Message Lost														
			20	TXQMOW	TXQ Message Overwrite							x							
			21	THLELT	TX History List Entry Lost														
			[31:2]	-	reserved														
24. Pretended Network Filter (Entry & Control)																			
CFDGPFLCTR	m_pnacstr	Pretended Network Filter List Entry control Register																	
			[5:0]	PFLPN	Pretended Network Filter List Page Number														
			[7:6]	-	reserved														
			8	PFLDAE	Pretended Network Filter List Data Access Enable														
			[31:9]	-	reserved														
CFDGPFLCFG0	m_pnacstr	Pretended Network Filter List Entry Configuration Register 0																	
			[5:0]	RNC3	Rule Number for Channel 3			x											
			[7:6]	-	reserved														
			[13:8]	RNC2	Rule Number for Channel 2			x											
			[15:1]	-	reserved														
			[21:1]	RNC1	Rule Number for Channel 1														
			[23:2]	0	reserved														
			[29:2]	RNC0	Rule Number for Channel 0														



		s																		
			[7:0]	GPFLFDP	Global Pretended Network Filter List FIFO Direction Pointer															
			[13:8]]	GPFLFDP	Global Pretended Network Filter List FIFO Direction Pointer															
			[19:1] 4]	GPFLFDP	Global Pretended Network Filter List FIFO Direction Pointer				xx											
			[25:2] 0]	GPFLFDP	Global Pretended Network Filter List FIFO Direction Pointer			xx	xx											
			[31:2] 6]	GPFLFDP	Global Pretended Network Filter List FIFO Direction Pointer		xx	xx	xx											

x : delete bit (can not write / always read 0)

xx : delete bit (should not write / read value is unknown)

5.8 software constraint

In specific parameter values, User should protect the following restrictions.

The range of the value which a user can set up is as follows.

	Resister	Bit	range of values	parameter
1	CFDCFCCd	CFTML	from 0 to 7	TXMB=16
			from 0 to 15	TXMB=32
2	CFDTXQCC0n	TXQDC	from 2 to 7	TXMB=16
	CFDTXQCC1n			
	CFDTXQCC2n		from 2 to 15	TXMB=32
	CFDTXQCC3n			
3	CFDGFTBAC	FLXMB0 FLXMB1 FLXMB2 FLXMB3	from 0 to 2	TXMB=16
			from 0 to 4	TXMB=32
4	CFDGF CFG	TSBTCS	from 0 to 1	CH=2
			from 0 to 3	CH=4
			from 0 to 5	CH=6
5	CFDGAFLP0r	GAFLRMDP	from 0 to 31	CH=2
			from 0 to 63	CH=4
			from 0 to 95	CH=6
6	CFDRMNB	NRXMB	from 0 to 32	CH=2
			from 0 to 64	CH=4
			from 0 to 96	CH=6
7	CFDGAFLIGNENT	ICN	from 0 to 1	CH=2
			from 0 to 3	CH=4
			from 0 to 5	CH=6
8	CFDGAFLCTR	AFLPN	from 0 to 7	CH=2, AFL=64
			from 0 to 15	CH=2, AFL=128
			from 0 to 23	CH=2, AFL=192
			from 0 to 15	CH=4, AFL=64
			from 0 to 31	CH=4, AFL=128
			from 0 to 47	CH=4, AFL=192
			from 0 to 23	CH=6, AFL=64
			from 0 to 47	CH=6, AFL=128
			from 0 to 71	CH=6, AFL=192
			from 0 to 31	CH=8, AFL=64
9	CFDGPFLCTR	PFLPN	from 0 to 63	CH=8, AFL=128
			from 0 to 95	CH=8, AFL=192
			from 0 to 9	CH=2
			from 0 to 19	CH=4
			from 0 to 29	CH=6
			from 0 to 39	CH=8

10	CFDGTSTCFG	RTMPS	from 0 to 39	CH=2, AFL=64, POOL=32, TXMB=16
			from 0 to 49	CH=2, AFL=64, POOL=32, TXMB=32
			from 0 to 68	CH=2, AFL=64, POOL=32, TXMB=64
			from 0 to 49	CH=2, AFL=64, POOL=48, TXMB=16
			from 0 to 58	CH=2, AFL=64, POOL=48, TXMB=32
			from 0 to 77	CH=2, AFL=64, POOL=48, TXMB=64
			from 0 to 58	CH=2, AFL=64, POOL=64, TXMB=16
			from 0 to 68	CH=2, AFL=64, POOL=64, TXMB=32
			from 0 to 87	CH=2, AFL=64, POOL=64, TXMB=64
			from 0 to 96	CH=2, AFL=64, POOL=128, TXMB=16
			from 0 to 106	CH=2, AFL=64, POOL=128, TXMB=32
			from 0 to 125	CH=2, AFL=64, POOL=128, TXMB=64
			from 0 to 172	CH=2, AFL=64, POOL=256, TXMB=16
			from 0 to 182	CH=2, AFL=64, POOL=256, TXMB=32
			from 0 to 201	CH=2, AFL=64, POOL=256, TXMB=64
			from 0 to 47	CH=2, AFL=128, POOL=32, TXMB=16
			from 0 to 57	CH=2, AFL=128, POOL=32, TXMB=32
			from 0 to 76	CH=2, AFL=128, POOL=32, TXMB=64
			from 0 to 57	CH=2, AFL=128, POOL=48, TXMB=16
			from 0 to 66	CH=2, AFL=128, POOL=48, TXMB=32
			from 0 to 85	CH=2, AFL=128, POOL=48, TXMB=64
			from 0 to 66	CH=2, AFL=128, POOL=64, TXMB=16
			from 0 to 76	CH=2, AFL=128, POOL=64, TXMB=32
			from 0 to 95	CH=2, AFL=128, POOL=64, TXMB=64
			from 0 to 104	CH=2, AFL=128, POOL=128, TXMB=16
			from 0 to 114	CH=2, AFL=128, POOL=128, TXMB=32
			from 0 to 133	CH=2, AFL=128, POOL=128, TXMB=64
			from 0 to 180	CH=2, AFL=128, POOL=256, TXMB=16
			from 0 to 190	CH=2, AFL=128, POOL=256, TXMB=32
			from 0 to 209	CH=2, AFL=128, POOL=256, TXMB=64
			from 0 to 55	CH=2, AFL=192, POOL=32, TXMB=16
			from 0 to 65	CH=2, AFL=192, POOL=32, TXMB=32
			from 0 to 84	CH=2, AFL=192, POOL=32, TXMB=64
			from 0 to 65	CH=2, AFL=192, POOL=48, TXMB=16
			from 0 to 74	CH=2, AFL=192, POOL=48, TXMB=32
			from 0 to 93	CH=2, AFL=192, POOL=48, TXMB=64
			from 0 to 74	CH=2, AFL=192, POOL=64, TXMB=16
			from 0 to 84	CH=2, AFL=192, POOL=64, TXMB=32
			from 0 to 103	CH=2, AFL=192, POOL=64, TXMB=64
			from 0 to 112	CH=2, AFL=192, POOL=128, TXMB=16
			from 0 to 122	CH=2, AFL=192, POOL=128, TXMB=32
			from 0 to 141	CH=2, AFL=192, POOL=128, TXMB=64

		from 0 to 188	CH=2, AFL=192, POOL=256, TXMB=16
		from 0 to 198	CH=2, AFL=192, POOL=256, TXMB=32
		from 0 to 217	CH=2, AFL=192, POOL=256, TXMB=64
		from 0 to 79	CH=4, AFL=64, POOL=32, TXMB=16
		from 0 to 98	CH=4, AFL=64, POOL=32, TXMB=32
		from 0 to 136	CH=4, AFL=64, POOL=32, TXMB=64
		from 0 to 98	CH=4, AFL=64, POOL=48, TXMB=16
		from 0 to 117	CH=4, AFL=64, POOL=48, TXMB=32
		from 0 to 155	CH=4, AFL=64, POOL=48, TXMB=64
		from 0 to 117	CH=4, AFL=64, POOL=64, TXMB=16
		from 0 to 136	CH=4, AFL=64, POOL=64, TXMB=32
		from 0 to 174	CH=4, AFL=64, POOL=64, TXMB=64
		from 0 to 193	CH=4, AFL=64, POOL=128, TXMB=16
		from 0 to 212	CH=4, AFL=64, POOL=128, TXMB=32
		from 0 to 250	CH=4, AFL=64, POOL=128, TXMB=64
		from 0 to 345	CH=4, AFL=64, POOL=256, TXMB=16
		from 0 to 364	CH=4, AFL=64, POOL=256, TXMB=32
		from 0 to 402	CH=4, AFL=64, POOL=256, TXMB=64
		from 0 to 95	CH=4, AFL=128, POOL=32, TXMB=16
		from 0 to 114	CH=4, AFL=128, POOL=32, TXMB=32
		from 0 to 152	CH=4, AFL=128, POOL=32, TXMB=64
		from 0 to 114	CH=4, AFL=128, POOL=48, TXMB=16
		from 0 to 133	CH=4, AFL=128, POOL=48, TXMB=32
		from 0 to 171	CH=4, AFL=128, POOL=48, TXMB=64
		from 0 to 133	CH=4, AFL=128, POOL=64, TXMB=16
		from 0 to 152	CH=4, AFL=128, POOL=64, TXMB=32
		from 0 to 190	CH=4, AFL=128, POOL=64, TXMB=64
		from 0 to 209	CH=4, AFL=128, POOL=128, TXMB=16
		from 0 to 228	CH=4, AFL=128, POOL=128, TXMB=32
		from 0 to 266	CH=4, AFL=128, POOL=128, TXMB=64
		from 0 to 361	CH=4, AFL=128, POOL=256, TXMB=16
		from 0 to 380	CH=4, AFL=128, POOL=256, TXMB=32
		from 0 to 418	CH=4, AFL=128, POOL=256, TXMB=64
		from 0 to 111	CH=4, AFL=192, POOL=32, TXMB=16
		from 0 to 130	CH=4, AFL=192, POOL=32, TXMB=32
		from 0 to 168	CH=4, AFL=192, POOL=32, TXMB=64
		from 0 to 130	CH=4, AFL=192, POOL=48, TXMB=16
		from 0 to 149	CH=4, AFL=192, POOL=48, TXMB=32
		from 0 to 187	CH=4, AFL=192, POOL=48, TXMB=64
		from 0 to 149	CH=4, AFL=192, POOL=64, TXMB=16
		from 0 to 168	CH=4, AFL=192, POOL=64, TXMB=32
		from 0 to 206	CH=4, AFL=192, POOL=64, TXMB=64

		from 0 to 225	CH=4, AFL=192, POOL=128, TXMB=16
		from 0 to 244	CH=4, AFL=192, POOL=128, TXMB=32
		from 0 to 282	CH=4, AFL=192, POOL=128, TXMB=64
		from 0 to 377	CH=4, AFL=192, POOL=256, TXMB=16
		from 0 to 396	CH=4, AFL=192, POOL=256, TXMB=32
		from 0 to 434	CH=4, AFL=192, POOL=256, TXMB=64
		from 0 to 119	CH=6, AFL=64, POOL=32, TXMB=16
		from 0 to 147	CH=6, AFL=64, POOL=32, TXMB=32
		from 0 to 204	CH=6, AFL=64, POOL=32, TXMB=64
		from 0 to 147	CH=6, AFL=64, POOL=48, TXMB=16
		from 0 to 176	CH=6, AFL=64, POOL=48, TXMB=32
		from 0 to 233	CH=6, AFL=64, POOL=48, TXMB=64
		from 0 to 176	CH=6, AFL=64, POOL=64, TXMB=16
		from 0 to 204	CH=6, AFL=64, POOL=64, TXMB=32
		from 0 to 261	CH=6, AFL=64, POOL=64, TXMB=64
		from 0 to 290	CH=6, AFL=64, POOL=128, TXMB=16
		from 0 to 318	CH=6, AFL=64, POOL=128, TXMB=32
		from 0 to 375	CH=6, AFL=64, POOL=128, TXMB=64
		from 0 to 518	CH=6, AFL=64, POOL=256, TXMB=16
		from 0 to 546	CH=6, AFL=64, POOL=256, TXMB=32
		from 0 to 603	CH=6, AFL=64, POOL=256, TXMB=64
		from 0 to 143	CH=6, AFL=128, POOL=32, TXMB=16
		from 0 to 171	CH=6, AFL=128, POOL=32, TXMB=32
		from 0 to 228	CH=6, AFL=128, POOL=32, TXMB=64
		from 0 to 171	CH=6, AFL=128, POOL=48, TXMB=16
		from 0 to 200	CH=6, AFL=128, POOL=48, TXMB=32
		from 0 to 257	CH=6, AFL=128, POOL=48, TXMB=64
		from 0 to 200	CH=6, AFL=128, POOL=64, TXMB=16
		from 0 to 228	CH=6, AFL=128, POOL=64, TXMB=32
		from 0 to 285	CH=6, AFL=128, POOL=64, TXMB=64
		from 0 to 314	CH=6, AFL=128, POOL=128, TXMB=16
		from 0 to 342	CH=6, AFL=128, POOL=128, TXMB=32
		from 0 to 399	CH=6, AFL=128, POOL=128, TXMB=64
		from 0 to 542	CH=6, AFL=128, POOL=256, TXMB=16
		from 0 to 570	CH=6, AFL=128, POOL=256, TXMB=32
		from 0 to 627	CH=6, AFL=128, POOL=256, TXMB=64
		from 0 to 167	CH=6, AFL=192, POOL=32, TXMB=16
		from 0 to 195	CH=6, AFL=192, POOL=32, TXMB=32
		from 0 to 252	CH=6, AFL=192, POOL=32, TXMB=64
		from 0 to 195	CH=6, AFL=192, POOL=48, TXMB=16
		from 0 to 224	CH=6, AFL=192, POOL=48, TXMB=32
		from 0 to 281	CH=6, AFL=192, POOL=48, TXMB=64

		from 0 to 224	CH=6, AFL=192, POOL=64, TXMB=16
		from 0 to 252	CH=6, AFL=192, POOL=64, TXMB=32
		from 0 to 309	CH=6, AFL=192, POOL=64, TXMB=64
		from 0 to 338	CH=6, AFL=192, POOL=128, TXMB=16
		from 0 to 366	CH=6, AFL=192, POOL=128, TXMB=32
		from 0 to 423	CH=6, AFL=192, POOL=128, TXMB=64
		from 0 to 566	CH=6, AFL=192, POOL=256, TXMB=16
		from 0 to 594	CH=6, AFL=192, POOL=256, TXMB=32
		from 0 to 651	CH=6, AFL=192, POOL=256, TXMB=64
		from 0 to 158	CH=8, AFL=64, POOL=32, TXMB=16
		from 0 to 196	CH=8, AFL=64, POOL=32, TXMB=32
		from 0 to 272	CH=8, AFL=64, POOL=32, TXMB=64
		from 0 to 196	CH=8, AFL=64, POOL=48, TXMB=16
		from 0 to 234	CH=8, AFL=64, POOL=48, TXMB=32
		from 0 to 310	CH=8, AFL=64, POOL=48, TXMB=64
		from 0 to 234	CH=8, AFL=64, POOL=64, TXMB=16
		from 0 to 272	CH=8, AFL=64, POOL=64, TXMB=32
		from 0 to 348	CH=8, AFL=64, POOL=64, TXMB=64
		from 0 to 386	CH=8, AFL=64, POOL=128, TXMB=16
		from 0 to 424	CH=8, AFL=64, POOL=128, TXMB=32
		from 0 to 500	CH=8, AFL=64, POOL=128, TXMB=64
		from 0 to 690	CH=8, AFL=64, POOL=256, TXMB=16
		from 0 to 728	CH=8, AFL=64, POOL=256, TXMB=32
		from 0 to 804	CH=8, AFL=64, POOL=256, TXMB=64
		from 0 to 190	CH=8, AFL=128, POOL=32, TXMB=16
		from 0 to 228	CH=8, AFL=128, POOL=32, TXMB=32
		from 0 to 304	CH=8, AFL=128, POOL=32, TXMB=64
		from 0 to 228	CH=8, AFL=128, POOL=48, TXMB=16
		from 0 to 266	CH=8, AFL=128, POOL=48, TXMB=32
		from 0 to 342	CH=8, AFL=128, POOL=48, TXMB=64
		from 0 to 266	CH=8, AFL=128, POOL=64, TXMB=16
		from 0 to 304	CH=8, AFL=128, POOL=64, TXMB=32
		from 0 to 380	CH=8, AFL=128, POOL=64, TXMB=64
		from 0 to 418	CH=8, AFL=128, POOL=128, TXMB=16
		from 0 to 456	CH=8, AFL=128, POOL=128, TXMB=32
		from 0 to 532	CH=8, AFL=128, POOL=128, TXMB=64
		from 0 to 722	CH=8, AFL=128, POOL=256, TXMB=16
		from 0 to 760	CH=8, AFL=128, POOL=256, TXMB=32
		from 0 to 836	CH=8, AFL=128, POOL=256, TXMB=64
		from 0 to 222	CH=8, AFL=192, POOL=32, TXMB=16
		from 0 to 260	CH=8, AFL=192, POOL=32, TXMB=32
		from 0 to 336	CH=8, AFL=192, POOL=32, TXMB=64

		from 0 to 260	CH=8, AFL=192, POOL=48, TXMB=16
		from 0 to 298	CH=8, AFL=192, POOL=48, TXMB=32
		from 0 to 374	CH=8, AFL=192, POOL=48, TXMB=64
		from 0 to 298	CH=8, AFL=192, POOL=64, TXMB=16
		from 0 to 336	CH=8, AFL=192, POOL=64, TXMB=32
		from 0 to 412	CH=8, AFL=192, POOL=64, TXMB=64
		from 0 to 450	CH=8, AFL=192, POOL=128, TXMB=16
		from 0 to 488	CH=8, AFL=192, POOL=128, TXMB=32
		from 0 to 564	CH=8, AFL=192, POOL=128, TXMB=64
		from 0 to 754	CH=8, AFL=192, POOL=256, TXMB=16
		from 0 to 792	CH=8, AFL=192, POOL=256, TXMB=32
		from 0 to 868	CH=8, AFL=192, POOL=256, TXMB=64

Initialization conditions change with TXMB parameter values.

For Example:

When TXMB parameter = 64 and **CFDGFTBAC.FLXMB0=8** and **CFDTXQCC1[1].TXQDC=31**, **CFDTXQSTS1[1]** is initialized automatically when Channel 0 enters CH_RESET mode.

When TXMB parameter = 64 and **CFDGFTBAC.FLXMB0=4** and **CFDTXQCC1[1].TXQDC=15**, **CFDTXQSTS1[1]** is initialized automatically when Channel 1 enters CH_RESET mode.

When TXMB parameter = 32 and **CFDGFTBAC.FLXMB0=4** and **CFDTXQCC1[1].TXQDC=15**, **CFDTXQSTS1[1]** is initialized automatically when Channel 0 enters CH_RESET mode.

When TXMB parameter = 64 and **CFDGFTBAC.FLXMB0=2** and **CFDTXQCC1[1].TXQDC=7**, **CFDTXQSTS1[1]** is initialized automatically when Channel 1 enters CH_RESET mode.

When TXMB parameter = 32 and **CFDGFTBAC.FLXMB0=2** and **CFDTXQCC1[1].TXQDC=7**, **CFDTXQSTS1[1]** is initialized automatically when Channel 1 enters CH_RESET mode.

When TXMB parameter = 16 and **CFDGFTBAC.FLXMB0=2** and **CFDTXQCC1[1].TXQDC=7**, **CFDTXQSTS1[1]** is initialized automatically when Channel 0 enters CH_RESET mode.

The user should input the same value as channel parameter values.

Or user should input a value smaller than channel parameter.

No	Input pin	range of values	parameter
1	num_ram_chan	from 0 to 2	CH=2
2		from 0 to 4	CH=4
3		from 0 to 6	CH=6
4		from 0 to 8	CH=8

If input value is 0, this is the same meaning as having inputted the channel parameter.

The number of the maximum rules with an AFL rule parameter is as follows.

AFL parameter	Maximum number of AFL rules
AFL=64	128 rules / ch
AFL=128	255 rules / ch
AFL=192	384 rules / ch

5.9 Influence of the operation by Parameterization

The part of the change of operation with a parameter is explained.

The user manual preparation person needs to create a user manual with reference to this section.

7 Acceptance Filtering Function using Global Acceptance Filter List (AFL)

7.1 Overview

The number and sum total value of an AFL entry are modified by the parameter.

7.2 Allocation of AFL entries to each CAN channel

The number and sum total value of an AFL entry are modified by the parameter.

Modification of Figure 7.1 is required.

7.4 Entering entries in the AFL

Number of channels, the number of AFL rules, and page number need to be modified.

Modification of Figure 7.3 is required.

8 FIFO Buffers & Normal MB Configuration

Modification of Figure 8.1 is required.

8.1.1 Normal RX Message Buffer configuration

The number of the sum totals of RXMB needs to be modified.

8.2.1.2 FIFO TX-Message Buffer Link configuration

The TXMB number of link destinations needs to be modified.

8.2.1.3 FIFO Depth Configuration

The range of a RAM allocation needs to be modified.

8.2.1.4 FIFO Payload Size Configuration

The range of a RAM allocation needs to be modified.

9 Interrupts and DMA

9.1 Interrupts

Modification of Figure is required.

9.2 DMA Transfer

Modification of Figure 9.3 is required.

The number of COMFIFO and TXQ for DMA needs to be modified.

10 Reception and Transmission

10.2 Transmission

The number of TXMB needs to be modified.

The width of TXQ needs to be modified.

Modification of Figure 10.5 is required.

10.2.3.3 Interval Timer for FIFO Transmission

Worst case time needs to be modified.

10.2.4 TX Queue

The width of TXQ needs to be modified.

10.2.5 TX History List

Worst case time needs to be modified.

Modification of Table 10.3 is required.

11 ECC Check

Modification of Figure 11.1, 11.2 and 11.3 is required.

Since the parameter maximum specification explains, change is required.

12 Test Mode

12.2.1 RAM Test Mode

Modification of Figure 12.2 is required.

Since the parameter maximum specification explains, change is required.

13 RAM area configuration

Since the parameter maximum specification explains, change is required.

Modification of Figure 13.1 is required.

13.1 Examples

Modification of Figure 13.2 is required.

13.3 num_ram_chan dependencies

Since the parameter maximum specification explains, change is required.

15 Flexible CAN mode

Since the parameter maximum specification explains, change is required.

16 Flexible transmission buffer assignment

Since the parameter maximum specification explains, change is required.

Modification of Figure is required.

Modification of Figure 16.1, 16.2 and 16.3 is required.

17 FFI mode

17.4 Restrictions function in FFI mode

Since 8 channels explain, change is required.

Modification of Figure 17.2 is required.

Modification of Table 17.1 to Table 17.12 is required.

18 PNF(Pretended Network Filter)

18.2.3 Setup of the number of rules of PNF

Modification of Figure 18.3 is required.

18.2.4 Software flow of a payload filter

Since 8 channels explain, change is required.

5.10 Influence of the bit by Classical CAN only mode

The bit which is subject to the influence of a Classical CAN only mode is shown below.

The user manual preparation person needs to create a user manual with reference to this section.

RS-CAN-FD SFR feature List								
	Module	Register name	Bit	Symbol	Description	unused bit		comment
						Classical CAN only mode	Classical CAN only mode FD tolerant mode	
1. Channel Registers								
CFDCnNCF G	<i>m_chsfr</i>	Channel n Configuration Register						
			[9:0]	NBRP	Nominal Baudrate Prescaler			
			[16: 10]	NSJW	Nominal Synchronization Jump Width			
			[24: 17]	NTSEG1	Nominal Time Segment 1			
			[31: 25]	NTSEG2	Nominal Time Segment 2			
CFDCnDCF G	<i>m_chsfr</i>	Channel n Configuration Register				x	x	
			[7:0]	DBRP	Data Baudrate Prescaler	x	x	
			[12: 8]	DTSEG1	Data Time Segment 1	x	x	
			[15: 13]	-	reserved	x	x	
			[19: 16]	DTSEG2	Data Time Segment 2	x	x	
			[23: 20]	-	reserved	x	x	
			[27: 24]	DSJW	Data Synchroniz	x	x	

					<i>ation Jump Width</i>			
			[31: 28]	-	<i>reserved</i>	x	x	
<i>CFDCnCTR</i>	<i>m_chsfr</i>	<i>Channel n Control Register</i>						
			[1:0]	<i>CHMDC</i>	<i>Channel Mode Control</i>			
			2	<i>CSLPR</i>	<i>Channel Sleep Request</i>			
			3	<i>RTBO</i>	<i>Return from Bus-Off</i>			
			[7:4]	-	<i>reserved</i>			
			8	<i>BEIE</i>	<i>Bus Error Interrupt Enable</i>			
			9	<i>EWIE</i>	<i>Error Warning Interrupt Enable</i>			
			10	<i>EPIE</i>	<i>Error Passive Interrupt Enable</i>			
			11	<i>BOEIE</i>	<i>Bus-Off Entry Interrupt Enable</i>			
			12	<i>BORIE</i>	<i>Bus-Off Recovery Interrupt Enable</i>			
			13	<i>OLIE</i>	<i>Overload Interrupt Enable</i>			
			14	<i>BLIE</i>	<i>Bus Lock Interrupt Enable</i>			
			15	<i>ALIE</i>	<i>Arbitration Lost Interrupt Enable</i>			
			16	<i>TAIE</i>	<i>Transmission Abortion Interrupt Enable</i>			
			17	<i>EOCOIE</i>	<i>Error Occurrence Counter Overflow Interrupt Enable</i>			

			18	<i>SOCOIE</i>	<i>Successful Occurrence Counter Overflow Interrupt Enable</i>			
			19	<i>TDCVFIE</i>	<i>Transceiver Delay compensation Bit FIFO Msg Lost Interrupt enable</i>	x	x	
			20	-	<i>reserved</i>			
		[22:21]	<i>BOM</i>	<i>Bus-Off Mode</i>				
		23	<i>ERRD</i>	<i>Error Display</i>				
		24	<i>CTME</i>	<i>Channel Test Mode Enable</i>				
		[26:25]	<i>CTMS</i>	<i>Channel Test Mode Select</i>				
		27	<i>TRWE</i>	<i>TEC/REC Write Enable</i>				
		28	<i>TRH</i>	<i>TEC/REC Hold</i>				
		29	<i>TRR</i>	<i>TEC/REC Reset</i>				
		30	<i>CRCT</i>	<i>CRC Test mode</i>				<i>This bit has description of CANFD. It is necessary to delete this description.</i>
		31	<i>ROM</i>	<i>Restricted Operation Mode</i>	x	x		
<i>CFDCnSTS</i>	<i>m_chsfr</i>	<i>Channel n Status Register</i>						
		0	<i>CRSTSTS</i>	<i>Channel RESET State</i>				
		1	<i>CHLTSTS</i>	<i>Channel HALT State</i>				

			2	CSLPSTS	Channel SLEEP State			
			3	EPSTS	Error Passive Status			
			4	BOSTS	Bus-Off Status			
			5	TRMSTS	Transmit Status			
			6	RECSTS	Receive Status			
			7	COMSTS	Communication Status			
			8	ESIF	Error State Indication Flag	x	x	
		[15: 9]	-		reserved			
		[23: 16]	REC		Reception Error Count			
		[31: 24]	TEC		Transmission Error Count			
<hr/>								
CFDCnERFL	<i>m_chsfr</i>	Channel n Error Flag Register						
			0	BEF	Bus Error Flag			
			1	EWF	Error Warning Flag			
			2	EPF	Error Passive Flag			
			3	BOEF	Bus-Off Entry Flag			
			4	BORF	Bus-Off Recovery Flag			
			5	OVLF	Overload Flag			
			6	BLF	Bus Lock Flag			
			7	ALF	Arbitration Lost Flag			
			8	SERR	Stuff Error			
			9	FERR	Form Error			
			10	AERR	Ack Error			
			11	CERR	CRC Error			
			12	B1ERR	Bit 1 Error			
			13	B0ERR	Bit 0 Error			

			14	<i>ADERR</i>	<i>Ack Del Error</i>			
			15	-	<i>reserved</i>			
			[30:16]	<i>CRCREG</i>	<i>CRC register value</i>			
			31	-	<i>reserved</i>			
<i>CFDCnFDCF G</i>	<i>m_chsfr</i>	<i>Channel n CAN-FD Configuration Register</i>	[2:0]	<i>EOCCFG</i>	<i>Error Occurrence Counter Configuration</i>			
			[7:3]	-	<i>reserved</i>			
			8	<i>TDCOC</i>	<i>Transceiver Delay Compensation Offset Configuration</i>	x	x	
			9	<i>TDCE</i>	<i>Transceiver Delay Compensation Enable</i>	x	x	
			10	<i>ESIC</i>	<i>Error State Indication Configuration</i>	x	x	
			[11]	-	<i>reserved</i>			
			[13:12]	<i>RPNMD</i>	<i>Return Pretended Network Filter Mode</i>			
			[15:14]	-	<i>reserved</i>			
			[23:16]	<i>TDCO</i>	<i>Transceiver Delay Compensation Offset</i>	x	x	
			24	<i>GWEN</i>	<i>CAN2.0 <> CAN FD GW Enable</i>	x	x	
			25	<i>GWDF</i>	<i>GW FDF Configuration</i>	x	x	
			26	<i>GWBR</i>	<i>GW BRS Configuration</i>	x	x	
			27	-	<i>reserved</i>			
			28	<i>FDOE</i>	<i>FD only enable</i>	x	x	
			29	<i>REFE</i>	<i>RX edge filter enable</i>	x		

			30	CLOE	Classical CAN only enable	x	x	Terminal Fix
			31	CFDTE	CAN-FD Tolerance enable	x		Terminal Fix
CFDCnFDCT R	m_chsfr	Channel n CAN-FD Control Register	0	EOCCLR	Error Occurrence Counter Clear			
			1	SOCCLR	Successful Occurrence Counter Clear			
			[15: 2]	-	reserved			
			[17: 16]	PNMDC	Pretended Network Filter Mode Control			
			[23: 18]	-	reserved			
			[31: 24]	KEY	Key code			
CFDCnFDSTS	m_chsfr	Channel n CAN-FD Status Register	[7:0]	TDCR	Transceiver Delay Compensation Result	x	x	
			8	EOCO	Error occurrence counter overflow			
			9	SOCO	Successful occurrence counter overflow			
			[11: 10]	-	reserved			
			[13: 12]	PNSTS	Pretended Network Filter State			
			14	-	reserved			
			15	TDCVF	Transceiver Delay compensation violation flag	x	x	
			[23: 16]	EOC	Error occurrence counter register			
			[31: 24]	SOC	Successful occurrence			

					<i>counter register</i>			
<i>CFDCnFDRC</i>	<i>m_chsfr</i>	<i>Channel n CAN-FD CRC Register</i>	<i>[20:0]</i>	<i>CRCREG</i>	<i>CRC Register value</i>	x	x	
			<i>[23:21]</i>	-	<i>reserved</i>	x	x	
			<i>[27:24]</i>	<i>SCNT</i>	<i>Stuff bit count</i>	x	x	
			<i>[31:28]</i>	-	<i>reserved</i>	x	x	
2. Global IP Registers								
<i>CFDGIPV</i>	<i>m_comsfr</i>	<i>Global IP Version Register</i>						
			<i>[7:0]</i>	<i>IPV</i>	<i>IP Version Release number</i>			
			<i>[9:8]</i>	<i>IPT</i>	<i>IP Type</i>			<i>This bit has description of CANFD . It is necessary to delete this description.</i>
			<i>[15:10]</i>	-	<i>reserved</i>			
			16	<i>PSI</i>	<i>Same ID overwrite function of TXQ</i>			
			<i>[20:17]</i>	<i>PSI</i>	<i>Number of Pool Buffer</i>			
			<i>[23:21]</i>	<i>PSI</i>	<i>Number of AFL entry</i>			
			<i>[26:24]</i>	<i>PSI</i>	<i>Number of TXMB</i>			
			<i>[29:27]</i>	<i>PSI</i>	<i>Number of channel</i>			
			<i>[31:30]</i>	-	<i>reserved</i>			
3. Global Configuration / Control Registers								
<i>CFDGCFG</i>	<i>m_comsfr</i>	<i>Global Configuration Register</i>	1					
			0	<i>TPRI</i>	<i>Transmission Priority</i>			

			1	DCE	DLC Check Enable			
			2	DRE	DLC Replacement Enable			
			3	MME	Mirror Mode Enable			
			4	DCS	PLL By-Pass			
			5	CMPOC	CAN-FD message Payload overflow configuration	x	x	
			[7:6]	-	reserved			
			[11:8]	TSP	Timestamp Prescaler			
			12	TSSS	Timestamp Source Select			This bit has description of CANFD. It is necessary to delete this description.
			[15:13]	TSBTCS	Timestamp Bit Time Channel Select			
			[31:16]	ITRCP	Interval Timer Reference Clock Prescaler			
CFDGCTR	m_comsfr	Global Control Register						
			[1:0]	GMDC	Global Mode Control			
			2	GSLPR	Global Sleep Request			
			[7:3]	-	reserved			
			8	DEIE	DLC Error Interrupt Enable			
			9	MEIE	Message Lost Error			

					<i>Interrupt Enable</i>			
			10	<i>THLEIE</i>	<i>TX History List Entry Lost Interrupt Enable</i>			
			11	<i>CMPOFIE</i>	<i>CAN-FD message payload overflow Flag Interrupt enable</i>	x	x	
			12	<i>QOWEIE</i>	<i>TXQ Message overwrite Error Interrupt Enable</i>			
			13	-	<i>reserved</i>			
			14	<i>QMEIE</i>	<i>TXQ Message lost Error Interrupt Enable</i>			
			15	<i>MOWEIE</i>	<i>Message overwrite Error Interrupt Enable</i>			
			16	<i>TSRST</i>	<i>TS Reset</i>			
			17	<i>TSWR</i>	<i>Timestamp Write</i>			
			[31: 18]	-	<i>reserved</i>			
<i>CFDGFDCF G</i>	<i>m_comsfr</i>	<i>Global FD configuration register</i>						
			0	<i>RPED</i>	<i>Protocol exception state disable</i>	x		
			[7:1]	-	<i>reserved</i>			
			[9:8]	<i>TSCCFG</i>	<i>Time stamp capture configuration</i>			<i>This bit has description of CANFD . It is necessary to delete</i>

								<i>this descrip- tion.</i>
			[31: 10]	-	reserved			
CFDGCRCC FG	<i>m_comstr</i>	<i>Global CRC configura- tion register</i>						
			0	<i>NIE</i>	<i>Non ISO enable</i>	x	x	
			[31: 1]	-	reserved	x	x	
4. Global Status Registers								
CFDGSTS	<i>m_comsfr</i>	<i>Global Status Register</i>						
			0	<i>GRSTSTS</i>	<i>Global RESET Status</i>			
			1	<i>GHLTSTS</i>	<i>Global HALT Status</i>			
			2	<i>GSLPSTS</i>	<i>Global SLEEP Status</i>			
			3	<i>GRAMINIT</i>	<i>Global RAM Initialisatio- n Status</i>			
			[31: 4]	-	reserved			
CFDGERFL	<i>m_comsfr</i>	<i>Global Error Flag Register</i>						
			0	<i>DEF</i>	<i>DLC Error Flag</i>			
			1	<i>MES</i>	<i>Message Lost Error Status</i>			
			2	<i>THLES</i>	<i>TX History List Entry Lost Error Status</i>			
			3	<i>CMPOF</i>	<i>CAN-FD message payload overflow Flag</i>	x	x	
			4	<i>QOWES</i>	<i>TXQ Message overwrite</i>			

					Error Status			
			5	OTBMLTSTS	OTB FIFO Msg Lost Status			
			6	QMES	TXQ Message Lost Error Status			
			7	MOWES	Message overwrite Error Status			
			8	RXSFAIL0	RX SCAN Fail Channel 0			
			9	RXSFAIL1	RX SCAN Fail Channel 1			
			10	RXSFAIL2	RX SCAN Fail Channel 2			
			11	RXSFAIL3	RX SCAN Fail Channel 3			
			12	RXSFAIL4	RX SCAN Fail Channel 4			
			13	RXSFAIL5	RX SCAN Fail Channel 5			
			14	RXSFAIL6	RX SCAN Fail Channel 6			
			15	RXSFAIL7	RX SCAN Fail Channel 7			
			16	EEF0	ECC Error Flag for Channel 0			
			17	EEF1	ECC Error Flag for Channel 1			
			18	EEF2	ECC Error Flag for Channel 2			
			19	EEF3	ECC Error Flag for Channel 3			
			20	EEF4	ECC Error Flag for Channel 4			
			21	EEF5	ECC Error Flag for Channel 5			

			22	<i>EEF6</i>	<i>ECC Error Flag for Channel 6</i>			
			23	<i>EEF7</i>	<i>ECC Error Flag for Channel 7</i>			
			[31: 24]	-	<i>reserved</i>			
<hr/>								
<i>CFDGTSC</i>	<i>m_timestamp</i>	<i>Global Timestamp Counter Register</i>						
			[15: 0]	<i>TS</i>	<i>Timestamp Value</i>			
			[31: 16]	-	<i>reserved</i>			
5. Global Acceptance Filter List Configuration Registers								
<i>CFDGAFLEC TR</i>	<i>m_acsfr</i>	<i>Global Acceptance Filter List Entry Control Register</i>						
			[6:0]	<i>AFLPN</i>	<i>Acceptance Filter List Page Number</i>			
			7	-	<i>reserved</i>			
			8	<i>AFLDAE</i>	<i>Acceptance Filter List Access Enable Data</i>			
			[31: 9]	-	<i>reserved</i>			
<hr/>								
<i>CFDGAFLCF G0</i>	<i>m_acsfr</i>	<i>Global Acceptance Filter List Configuration 0 Register</i>						
			[8:0]	<i>RNC1</i>	<i>Rule Number for Channel 1</i>			
			[15: 9]	-	<i>reserved</i>			
			[24: 16]	<i>RNC0</i>	<i>Rule Number for Channel 0</i>			
			[31: 25]	-	<i>reserved</i>			
<hr/>								

<i>CFDGAFLCF G1</i>	<i>m_acsfr</i>	<i>Global Acceptance Filter List Configuration 1 Register</i>						
			[8:0]	RNC3	<i>Rule Number for Channel 3</i>			
			[15: 9]	-	<i>reserved</i>			
			[24: 16]	RNC2	<i>Rule Number for Channel 2</i>			
			[31: 25]	-	<i>reserved</i>			
<hr/>								
<i>CFDGAFLCF G2</i>	<i>m_acsfr</i>	<i>Global Acceptance Filter List Configuration 2 Register</i>						
			[8:0]	RNC5	<i>Rule Number for Channel 5</i>			
			[15: 9]	-	<i>reserved</i>			
			[24: 16]	RNC4	<i>Rule Number for Channel 4</i>			
			[31: 25]	-	<i>reserved</i>			
<hr/>								
<i>CFDGAFLCF G3</i>	<i>m_acsfr</i>	<i>Global Acceptance Filter List Configuration 3 Register</i>						
			[8:0]	RNC7	<i>Rule Number for Channel 7</i>			
			[15: 9]	-	<i>reserved</i>			
			[24: 16]	RNC6	<i>Rule Number for Channel 6</i>			
			[31: 25]	-	<i>reserved</i>			
6. RX Mailbox Registers								
<i>CFDRMNB</i>	<i>m_acsfr</i>	<i>RX Mailbox</i>						

		<i>Number Register</i>						
			[7:0]	<i>NRXMB</i>	<i>Number of RX MB</i>			
			[10:8]	<i>RMPLS</i>	<i>Reception Message Buffer Payload Data Size</i>	x	x	
			[31:11]	-	<i>reserved</i>			
<i>CFDRMNDt</i>	<i>m_acsfr</i>	<i>RX-Mailbox NewData Registers</i>						
			[31:0]	<i>RMNS</i>	<i>RX Mailboxes Newdata Flag</i>			<i>This bit has description of CANFD . It is necessary to delete this description.</i>
7. RX FIFO Registers								
<i>CFDRFCCa</i>	<i>m_rxifou nit</i>	<i>RX FIFO Configuration / Control Registers [7:0]</i>						
			0	<i>RFE</i>	<i>RX FIFO Enable</i>			
			1	<i>RFIE</i>	<i>RX FIFO Interrupt Enable</i>			
			[3:2]	-	<i>reserved</i>			
			[6:4]	<i>RFPLS</i>	<i>Rx FIFO Payload Data Size configuration</i>	x	x	
			7	-	<i>reserved</i>			
			[10:8]	<i>RFDC</i>	<i>RX FIFO Depth Configuration</i>			
			11	-	<i>reserved</i>			
			12	<i>RFIM</i>	<i>RX FIFO Interrupt Mode</i>			

			[15: 13]	<i>RFIGCV</i>	<i>RX FIFO Interrupt Generation Counter Value</i>			
			16	<i>RFFIE</i>	<i>RX FIFO Full interrupt Enable</i>			
			[23: 17]	-	<i>reserved</i>			
			24	<i>MEIE</i>	<i>Message lost Error Interrupt Enable for FFI Mode</i>			
			[27: 25]	-	<i>reserved</i>			
			28	<i>DMAE</i>	<i>DMA Transfer Enable for RXFIFO a for FFI Mode</i>			
			[31: 29]	-	<i>reserved</i>			
<hr/>								
<i>CFDRFSTS_a</i>	<i>m_rxifou nit</i>	<i>RX FIFO Status Registers [7:0]</i>						
			0	<i>RFEMP</i>	<i>RX FIFO Empty</i>			
			1	<i>RFULL</i>	<i>RX FIFO Full</i>			
			2	<i>RFMLT</i>	<i>RX FIFO Msg Lost Flag</i>			
			3	<i>RFIF</i>	<i>RX FIFO Interrupt Flag</i>			
			[7:4]	-	<i>reserved</i>			
			[15: 8]	<i>RFMC</i>	<i>RX FIFO Message Count</i>			
			16	<i>RFFIF</i>	<i>RX FIFO Full Interrupt Flag</i>			
			[27: 17]	-	<i>reserved</i>			
			28	<i>RFDMASTS</i>	<i>DMA Transfer Status for RX FIFO a for FFI Mode</i>			

			[31: 29]	-	reserved			
<i>CFDRFPCTR a</i>	<i>m_rxifou nit</i>	<i>RX FIFO Pointer Control Register[7:0]</i>						
			[7:0]	<i>RPC</i>	<i>RX FIFO Pointer Control</i>			
			[31: 8]	-	reserved			
8. Common FIFO Registers								
<i>CFDCFCCd</i>	<i>m_comfif ounit</i>	<i>Common FIFO Configur ation / Control Registers [23:0]</i>						
			0	<i>CFE</i>	<i>Common FIFO Enable</i>			
			1	<i>CFRXIE</i>	<i>Common FIFO Interrupt Enable for RX Mode</i>			
			2	<i>CFTXIE</i>	<i>Common FIFO Interrupt Enable for TX Mode</i>			
			3	-	reserved			
			[6:4]	<i>CFPLS</i>	<i>Common FIFO Payload Data Size configura tion</i>	x	x	
			7	-	reserved			
			[9:8]	<i>CFM</i>	<i>Common FIFO Mode</i>			
			10	<i>CFITSS</i>	<i>Common FIFO Interval Timer Source Select</i>			<i>This bit has descrip tion of CANFD . It is necess ary to delete this descrip tion.</i>

			11	<i>CFITR</i>	<i>Common FIFO Interval Timer Reference Clock Resolution</i>			
			12	<i>CFIM</i>	<i>Common FIFO Interrupt Mode</i>			
			[15: 13]	<i>CFIGCV</i>	<i>Common FIFO Interrupt Generation Counter Value</i>			
			[20: 16]	<i>CFTML</i>	<i>Common FIFO TX-Mailbox Link</i>			
			[23: 21]	<i>CFDC</i>	<i>Common FIFO Depth Configuration</i>			
			[31: 24]	<i>CFITT</i>	<i>Common FIFO Interval Transmission Time</i>			
<i>CFDCFCCEd</i>	<i>m_comfifounit</i>	<i>Common FIFO Configuration / Control Registers 2 [23:0]</i>						
			0	<i>CFFIE</i>	<i>COMMON FIFO Full interrupt Enable</i>			
			1	<i>CFOFRXIE</i>	<i>COMMON FIFO One Frame Reception Interrupt Enable</i>			
			2	<i>CFOFTXIE</i>	<i>COMMON FIFO One Frame Transmission Interrupt Enable</i>			
			[7:3]	-	<i>reserved</i>			

			8	<i>CFMOWM</i>	<i>Common FIFO message overwrite mode</i>			
			[15: 9]	-	<i>reserved</i>			
			16	<i>CFBME</i>	<i>COMMON FIFO Buffering Mode Enable</i>			
			[23: 17]	-	<i>reserved</i>			
			24	<i>MEIE</i>	<i>Message lost Error Interrupt Enable for FFI Mode</i>			
			25	<i>MOWEIE</i>	<i>GW FIFO Message overwrite Error Interrupt Enable for FFI Mode</i>			
			[27: 26]	-	<i>reserved</i>			
			28	<i>DMAE</i>	<i>DMA Transfer Enable for Common FIFO a for FFI Mode</i>			
			[31: 29]	-	<i>reserved</i>			
<hr/>								
<i>CFDCFSTSd</i>	<i>m_comif ounit</i>	<i>Common FIFO status registers [23:0]</i>						
			0	<i>CFEMP</i>	<i>Common FIFO Empty</i>			
			1	<i>CFFLL</i>	<i>Common FIFO Full</i>			
			2	<i>CFMLT</i>	<i>Common FIFO Msg Lost</i>			
			3	<i>CFRXIF</i>	<i>Common reception FIFO Interrupt Flag</i>			
			4	<i>CFTXIF</i>	<i>Common transmit</i>			

					<i>FIFO Interrupt Flag</i>			
			[7:5]	-	<i>reserved</i>			
			[15:8]	<i>CFMC</i>	<i>Common FIFO Message Count</i>			
			16	<i>CFFIF</i>	<i>Common FIFO Full Interrupt Flag</i>			
			17	<i>CFOFRXIF</i>	<i>Common FIFO One Frame Reception</i>			
			18	<i>CFOFTXIF</i>	<i>Common FIFO One Frame Transmission</i>			
			[23:19]	-	<i>reserved</i>			
			24	<i>CFMOW</i>	<i>Common FIFO message overwrite</i>			
			[27:25]	-	<i>reserved</i>			
			28	<i>CFDMASTS</i>	<i>DMA Transfer Status for Common FIFO a for FFI Mode</i>			
			[31:29]	-	<i>reserved</i>			
<i>CFDCFPCTRd</i>	<i>m_comfitounit</i>	<i>Common FIFO Pointer Control Register [23:0]</i>						
			[7:0]	<i>CFPC</i>	<i>Common FIFO Pointer Control</i>			
			[31:8]	-	<i>reserved</i>			
9. FIFO Status Support Registers								
<i>CFDFESTS</i>	<i>m_acsfr</i>	<i>FIFO Empty Status Register</i>						

			[7:0]	RFxEMP	RX FIFO Empty Status			
			[13: 8]	CFxEMP	Common FIFO Empty Status			
			[19: 14]	CFxEMP	Common FIFO Empty Status			
			[25: 20]	CFxEMP	Common FIFO Empty Status			
			[31: 26]	CFxEMP	Common FIFO Empty Status			
CFDFFSTS	<i>m_acsfr</i>	<i>FIFO Full Status Register</i>						
			[7:0]	RFxFLL	RX FIFO Full Status			
			[13: 8]	CFxFLL	Common FIFO Full Status			
			[19: 14]	CFxFLL	Common FIFO Full Status			
			[25: 20]	CFxFLL	Common FIFO Full Status			
			[31: 26]	CFxFLL	Common FIFO Full Status			
CFDFFFSTS	<i>m_acsfr</i>	<i>FIFO FDC level Full Status Register</i>						
			[7:0]	RFxFFLL	RX FIFO FDC level full Status			
			[13: 8]	CFxFFLL	COMMON FIFO FDC level full Status			
			[19: 14]	CFxFFLL	COMMON FIFO FDC level full Status			
			[25: 20]	CFxFFLL	COMMON FIFO FDC			

					<i>level full Status</i>			
			[31: 26]	CFxFFLL	<i>COMMON FIFO FDC level full Status</i>			
<i>CFDFMSTS</i>	<i>m_acsfr</i>	<i>FIFO Msg Lost Status Register</i>						
			[7:0]	RFxMLT	<i>RX FIFO Msg Lost Status</i>			
			[13: 8]	CFxMLT	<i>Common FIFO Msg Lost Status</i>			
			[19: 14]	CFxMLT	<i>Common FIFO Msg Lost Status</i>			
			[25: 20]	CFxMLT	<i>Common FIFO Msg Lost Status</i>			
			[31: 26]	CFxMLT	<i>Common FIFO Msg Lost Status</i>			
<i>CFDCFMOW STS</i>	<i>m_acsfr</i>	<i>Common FIFO Message OverWrite Status</i>						
			[5:0]	CFxMOW	<i>Common FIFO [x] Message overwrite status</i>			
			[11: 6]	CFxMOW	<i>Common FIFO [x] Message overwrite status</i>			
			[17: 12]	CFxMOW	<i>Common FIFO [x] Message overwrite status</i>			
			[23: 18]	CFxMOW	<i>Common FIFO [x] Message overwrite status</i>			
			[31: 24]	-	<i>reserved</i>			
<i>CFDRFISTS</i>	<i>m_acsfr</i>	<i>RX FIFO Interrupt</i>						

		<i>Flag Status Register</i>						
			[7:0]	<i>RFxIF</i>	<i>RX FIFO Interrupt Flag Status</i>			
			[15:8]	-	<i>reserved</i>			
			[23:16]	<i>RFxFFLL</i>	<i>RX FIFO Full Interrupt Flag Status</i>			
			[31:24]	-	<i>reserved</i>			
<hr/>								
<i>CFDCFRIST</i> S	<i>m_acsfr</i>	<i>COM FIFO RX Interrupt Flag Status Register</i>						
			[5:0]	<i>CFxRXIF</i>	<i>RX Common FIFO Interrupt Flag Status</i>			
			[11:6]	<i>CFxRXIF</i>	<i>RX Common FIFO Interrupt Flag Status</i>			
			[17:12]	<i>CFxRXIF</i>	<i>RX Common FIFO Interrupt Flag Status</i>			
			[23:18]	<i>CFxRXIF</i>	<i>RX Common FIFO Interrupt Flag Status</i>			
			[31:24]	-	<i>reserved</i>			
<hr/>								
<i>CFDCFTIST</i> S	<i>m_acsfr</i>	<i>COM FIFO TX Interrupt Flag Status Register</i>						
			[5:0]	<i>CFxTXIF</i>	<i>TX Common FIFO Interrupt Flag Status</i>			
			[11:6]	<i>CFxTXIF</i>	<i>TX Common</i>			

					<i>FIFO Interrupt Flag Status</i>			
			[17: 12]	CFxTXIF	<i>TX Common FIFO Interrupt Flag Status</i>			
			[23: 18]	CFxTXIF	<i>TX Common FIFO Interrupt Flag Status</i>			
			[31: 24]	-	<i>reserved</i>			
<i>CFDCFOFRI STS</i>	<i>m_acsfr</i>	<i>Common FIFO One Frame RX Interrupt Flag Status</i>						
			[5:0]	CFxOFRXIF	<i>Common FIFO [x] One Frame RX Interrupt Flag Status</i>			
			[11: 6]	CFxOFRXIF	<i>Common FIFO [x] One Frame RX Interrupt Flag Status</i>			
			[17: 12]	CFxOFRXIF	<i>Common FIFO [x] One Frame RX Interrupt Flag Status</i>			
			[23: 18]	CFxOFRXIF	<i>Common FIFO [x] One Frame RX Interrupt Flag Status</i>			
			[31: 24]	-	<i>reserved</i>			
<i>CFDCFOFTI STS</i>	<i>m_acsfr</i>	<i>Common FIFO One Frame TX Interrupt</i>						

		<i>Flag Status</i>						
		[5:0]	<i>CFxOFTXIF</i>	<i>Common FIFO [x] One Frame TX Interrupt Flag Status</i>				
		[11: 6]	<i>CFxOFTXIF</i>	<i>Common FIFO [x] One Frame TX Interrupt Flag Status</i>				
		[17: 12]	<i>CFxOFTXIF</i>	<i>Common FIFO [x] One Frame TX Interrupt Flag Status</i>				
		[23: 18]	<i>CFxOFTXIF</i>	<i>Common FIFO [x] One Frame TX Interrupt Flag Status</i>				
		[31: 24]	-	<i>reserved</i>				
10. Special CAN-FD register								
<i>CFDCDTCT</i>	<i>m_dmaif</i>	<i>DMA Transfer Control Register</i>						
			[7:0]	<i>RFDMAEe</i>	<i>DMA Transfer Enable for RXFIFO e</i>			
			8	<i>CFDMAE0</i>	<i>DMA Transfer Enable for Common FIFO 0</i>			
			9	<i>CFDMAE1</i>	<i>DMA Transfer Enable for Common FIFO 1</i>			
			10	<i>CFDMAE2</i>	<i>DMA Transfer Enable for Common FIFO 2</i>			
			11	<i>CFDMAE3</i>	<i>DMA Transfer Enable for</i>			

					<i>Common FIFO 3</i>			
			12	<i>CFDMAE4</i>	<i>DMA Transfer Enable for Common FIFO 4</i>			
			13	<i>CFDMAE5</i>	<i>DMA Transfer Enable for Common FIFO 5</i>			
			14	<i>CFDMAE6</i>	<i>DMA Transfer Enable for Common FIFO 6</i>			
			15	<i>CFDMAE7</i>	<i>DMA Transfer Enable for Common FIFO 7</i>			
			[31: 16]	-	<i>reserved</i>			
<i>CFDCDTSTS</i>	<i>m_dmaif</i>	<i>DMA Transfer Status Register</i>						
			[7:0]	<i>RFDMASTSe</i>	<i>DMA Transfer Status for RX FIFO e</i>			
			8	<i>CFDMASTS0</i>	<i>DMA Transfer Status only for Common FIFO 0 of channel 0</i>			
			9	<i>CFDMASTS1</i>	<i>DMA Transfer Status only for Common FIFO 0 of channel 1</i>			
			10	<i>CFDMASTS2</i>	<i>DMA Transfer Status only for Common FIFO 0 of channel 2</i>			
			11	<i>CFDMASTS3</i>	<i>DMA Transfer Status only</i>			

					<i>for Common FIFO 0 of channel 3</i>			
			12	<i>CFDMASTS4</i>	<i>DMA Transfer Status only for Common FIFO 0 of channel 4</i>			
			13	<i>CFDMASTS5</i>	<i>DMA Transfer Status only for Common FIFO 0 of channel 5</i>			
			14	<i>CFDMASTS6</i>	<i>DMA Transfer Status only for Common FIFO 0 of channel 6</i>			
			15	<i>CFDMASTS7</i>	<i>DMA Transfer Status only for Common FIFO 0 of channel 7</i>			
			[31: 16]	-	<i>reserved</i>			
<i>CFDCDTTCT</i>	<i>m_dmaif</i>	<i>DMA TX Transfer Control Register</i>						
			0	<i>TQ0DMAE0</i>	<i>DMA TX Transfer Enable for TXQ0 0</i>			
			1	<i>TQ0DMAE1</i>	<i>DMA TX Transfer Enable for TXQ0 1</i>			
			2	<i>TQ0DMAE2</i>	<i>DMA TX Transfer Enable for TXQ0 2</i>			
			3	<i>TQ0DMAE3</i>	<i>DMA TX Transfer Enable for TXQ0 3</i>			
			4	<i>TQ0DMAE4</i>	<i>DMA TX Transfer</i>			

					<i>Enable for TXQ0 4</i>			
			5	TQ0DMAE5	<i>DMA TX Transfer Enable for TXQ0 5</i>			
			6	TQ0DMAE6	<i>DMA TX Transfer Enable for TXQ0 6</i>			
			7	TQ0DMAE7	<i>DMA TX Transfer Enable for TXQ0 7</i>			
			8	TQ3DMAE0	<i>DMA TX Transfer Enable for TXQ3 0</i>			
			9	TQ3DMAE1	<i>DMA TX Transfer Enable for TXQ3 1</i>			
			10	TQ3DMAE2	<i>DMA TX Transfer Enable for TXQ3 2</i>			
			11	TQ3DMAE3	<i>DMA TX Transfer Enable for TXQ3 3</i>			
			12	TQ3DMAE4	<i>DMA TX Transfer Enable for TXQ3 4</i>			
			13	TQ3DMAE5	<i>DMA TX Transfer Enable for TXQ3 5</i>			
			14	TQ3DMAE6	<i>DMA TX Transfer Enable for TXQ3 6</i>			
			15	TQ3DMAE7	<i>DMA TX Transfer Enable for TXQ3 7</i>			
			16	CFDMAE0	<i>DMA Transfer Enable for Common FIFO 0</i>			
			17	CFDMAE1	<i>DMA Transfer Enable for Common FIFO 1</i>			

			18	<i>CFDMAE2</i>	DMA Transfer Enable for Common FIFO 2			
			19	<i>CFDMAE3</i>	DMA Transfer Enable for Common FIFO 3			
			20	<i>CFDMAE4</i>	DMA Transfer Enable for Common FIFO 4			
			21	<i>CFDMAE5</i>	DMA Transfer Enable for Common FIFO 5			
			22	<i>CFDMAE6</i>	DMA Transfer Enable for Common FIFO 6			
			23	<i>CFDMAE7</i>	DMA Transfer Enable for Common FIFO 7			
			[31: 24]	-	reserved			
<hr/>								
<i>CFDCDTTSTS</i>	<i>m_dmaif</i>	DMA TX Transfer Status Register						
			0	<i>TQ0DMASTS 0</i>	DMA Transfer Status for TXQ0 0			
			1	<i>TQ0DMASTS 1</i>	DMA Transfer Status for TXQ0 1			
			2	<i>TQ0DMASTS 2</i>	DMA Transfer Status for TXQ0 2			
			3	<i>TQ0DMASTS 3</i>	DMA Transfer Status for TXQ0 3			
			4	<i>TQ0DMASTS 4</i>	DMA Transfer Status for TXQ0 4			

			5	TQ0DMASTS 5	DMA Transfer Status for TXQ0 5			
			6	TQ0DMASTS 6	DMA Transfer Status for TXQ0 6			
			7	TQ0DMASTS 7	DMA Transfer Status for TXQ0 7			
			8	TQ3DMASTS 0	DMA Transfer Status for TXQ3 0			
			9	TQ3DMASTS 1	DMA Transfer Status for TXQ3 1			
			10	TQ3DMASTS 2	DMA Transfer Status for TXQ3 2			
			11	TQ3DMASTS 3	DMA Transfer Status for TXQ3 3			
			12	TQ3DMASTS 4	DMA Transfer Status for TXQ3 4			
			13	TQ3DMASTS 5	DMA Transfer Status for TXQ3 5			
			14	TQ3DMASTS 6	DMA Transfer Status for TXQ3 6			
			15	TQ3DMASTS 7	DMA Transfer Status for TXQ3 7			
			16	CFDMASTS0	DMA Transfer Status only for Common FIFO 0 of channel 0			
			17	CFDMASTS1	DMA Transfer Status only for Common			

					<i>FIFO 0 of channel 1</i>			
			18	<i>CFDMASTS2</i>	<i>DMA Transfer Status only for Common FIFO 0 of channel 2</i>			
			19	<i>CFDMASTS3</i>	<i>DMA Transfer Status only for Common FIFO 0 of channel 3</i>			
			20	<i>CFDMASTS4</i>	<i>DMA Transfer Status only for Common FIFO 0 of channel 4</i>			
			21	<i>CFDMASTS5</i>	<i>DMA Transfer Status only for Common FIFO 0 of channel 5</i>			
			22	<i>CFDMASTS6</i>	<i>DMA Transfer Status only for Common FIFO 0 of channel 6</i>			
			23	<i>CFDMASTS7</i>	<i>DMA Transfer Status only for Common FIFO 0 of channel 7</i>			
			[31: 24]	-	<i>reserved</i>			
11. TX Mailbox Registers								
<i>CFDTMCi</i>	<i>m_mbctrl</i>	<i>TX Mailbox Control Registers i</i>						
			0	<i>TMTR</i>	<i>TX Mailbox Transmission Request</i>			
			1	<i>TMTAR</i>	<i>TX Mailbox Transmissi</i>			

					<i>on Abortion Request</i>			
			2	<i>TMOM</i>	<i>TX Mailbox One-shot mode</i>			
			[7:3]	-	<i>reserved</i>			
<i>CFDTMSTSj</i>	<i>m_mbctrl</i>	<i>TX Mailbox Status Registers j</i>						
			0	<i>TMTSTS</i>	<i>TX Mailbox Transmission Status</i>			
			[2:1]	<i>TMTRF</i>	<i>TX Mailbox Transmission Result Flag</i>			
			3	<i>TMTRM</i>	<i>TX Mailbox Transmission Request</i>			
			4	<i>TMTARM</i>	<i>TX Mailbox Transmission Abortion Request</i>			
			[7:5]	-	<i>reserved</i>			
<i>CFDTMTRST Sf</i>	<i>m_mbctrl</i>	<i>TX Mailboxe s Transmis sion Request Status Registers</i>						
			[7:0]	<i>CFDTMTRST S</i>	<i>TX Mailbox Transmission Request Status</i>			
			[15: 8]	<i>CFDTMTRST S</i>	<i>TX Mailbox Transmission Request Status</i>			
			[23: 16]	<i>CFDTMTRST S</i>	<i>TX Mailbox Transmission Request Status</i>			
			[31: 24]	<i>CFDTMTRST S</i>	<i>TX Mailbox Transmission Request Status</i>			
<i>CFDTMTAR STSf</i>	<i>m_mbctrl</i>	<i>TX Mailboxe s Transmis</i>						

		<i>sion Abortion Request Status Registers</i>						
			[7:0]	<i>CFDTMTARS TS</i>	<i>TX Mailbox Transmissi on Abortion Request Status</i>			
			[15: 8]	<i>CFDTMTARS TS</i>	<i>TX Mailbox Transmissi on Abortion Request Status</i>			
			[23: 16]	<i>CFDTMTARS TS</i>	<i>TX Mailbox Transmissi on Abortion Request Status</i>			
			[31: 24]	<i>CFDTMTARS TS</i>	<i>TX Mailbox Transmissi on Abortion Request Status</i>			
<i>CFDTMTCST Sf</i>	<i>m_mbctrl</i>	<i>TX Mailboxe s Transmis sion Completi on Status Registers</i>						
			[7:0]	<i>CFDTMTCST S</i>	<i>TX Mailbox Transmissi on Completion Status</i>			
			[15: 8]	<i>CFDTMTCST S</i>	<i>TX Mailbox Transmissi on Completion Status</i>			
			[23: 16]	<i>CFDTMTCST S</i>	<i>TX Mailbox Transmissi on Completion Status</i>			
			[31: 24]	<i>CFDTMTCST S</i>	<i>TX Mailbox Transmissi on Completion Status</i>			

<i>CFDTMTAST</i> <i>Sf</i>	<i>m_mbctrl</i>	<i>TX Mailboxes Transmission Abortion Status Registers</i>						
			[7:0]	<i>CFDTMTAST</i> <i>S</i>	<i>TX Mailbox Transmission Abortion Status</i>			
			[15:8]	<i>CFDTMTAST</i> <i>S</i>	<i>TX Mailbox Transmission Abortion Status</i>			
			[23:16]	<i>CFDTMTAST</i> <i>S</i>	<i>TX Mailbox Transmission Abortion Status</i>			
			[31:24]	<i>CFDTMTAST</i> <i>S</i>	<i>TX Mailbox Transmission Abortion Status</i>			
<i>CFDTMIEC</i> <i>f</i>	<i>m_mbctrl</i>	<i>TX Mailboxes Interrupt Enable Configuration Registers</i>						
			[7:0]	<i>TMIE</i>	<i>TX Mailbox Interrupt Enable</i>			
			[15:8]	<i>TMIE</i>	<i>TX Mailbox Interrupt Enable</i>			
			[23:16]	<i>TMIE</i>	<i>TX Mailbox Interrupt Enable</i>			
			[31:24]	<i>TMIE</i>	<i>TX Mailbox Interrupt Enable</i>			
12. TX Queue Registers								
<i>CFDTXQCC0</i> <i>n</i>	<i>m_mbctrl</i>	<i>TX Queue0 Configuration / Control Register n</i>						
			0	<i>TXQE</i>	<i>TX Queue Enable</i>			
			1	<i>TXQGWE</i>	<i>TX Queue Gateway</i>			

					<i>Mode Enable</i>			
			2	<i>TXQOWE</i>	<i>TX Queue Overwrite Mode Enable</i>			
			[4:3]	-	<i>reserved</i>			
			5	<i>TXQTXIE</i>	<i>TX Queue TX Interrupt Enable</i>			
			6	-	<i>reserved</i>			
			7	<i>TXQIM</i>	<i>TX Queue Interrupt Mode</i>			
			[12: 8]	<i>TXQDC</i>	<i>TX Queue Depth Configurati on</i>			
			[15: 13]	-	<i>reserved</i>			
			16	<i>TXQFIE</i>	<i>TXQ Full interrupt Enable</i>			
			17	<i>TXQOFRXIE</i>	<i>TXQ One Frame Reception Interrupt Enable</i>			
			18	<i>TXQOFTXIE</i>	<i>TXQ One Frame Transmissi on Interrupt Enable</i>			
			[23: 19]	-	<i>reserved</i>			
			24	<i>QMEIE</i>	<i>TXQ Message lost Error Interrupt Enable for FFI Mode</i>			
			25	<i>QOWEIE</i>	<i>TXQ Message overwrite Error Interrupt Enable for FFI Mode</i>			
			[27: 26]	-	<i>reserved</i>			
			28	<i>DMAE</i>	<i>DMA TX Transfer Enable for TXQ for FFI Mode</i>			

			[31: 29]	-	reserved			
<i>CFDTXQCC1 n</i>	<i>m_mbctrl</i>	<i>TX Queue1 Configur ation / Control Register n</i>						
			0	<i>TXQE</i>	<i>TX Queue Enable</i>			
			1	<i>TXQGWE</i>	<i>TX Queue Gateway Mode Enable</i>			
			2	<i>TXQOWE</i>	<i>TX Queue Overwrite Mode Enable</i>			
			[4:3]	-	reserved			
			5	<i>TXQTXIE</i>	<i>TX Queue TX Interrupt Enable</i>			
			6	-	reserved			
			7	<i>TXQIM</i>	<i>TX Queue Interrupt Mode</i>			
			[12: 8]	<i>TXQDC</i>	<i>TX Queue Depth Configura tion</i>			
			[15: 13]	-	reserved			
			16	<i>TXQFIE</i>	<i>TXQ Full interrupt Enable</i>			
			17	<i>TXQOFRXIE</i>	<i>TXQ One Frame Reception Interrupt Enable</i>			
			18	<i>TXQOFTXIE</i>	<i>TXQ One Frame Transmissi on Interrupt Enable</i>			
			[23: 19]	-	reserved			
			24	<i>QMEIE</i>	<i>TXQ Message lost Error Interrupt Enable for FFI Mode</i>			

			25	QOWEIE	TXQ Message overwrite Error Interrupt Enable for FFI Mode			
			[31: 26]	-	reserved			
<i>CFDTXQC2</i> <i>n</i>	<i>m_mbctrl</i>	<i>TX Queue2 Configuration / Control Register n</i>						
			0	TXQE	TX Queue Enable			
			1	TXQGWE	TX Queue Gateway Mode Enable			
			2	TXQOWE	TX Queue Overwrite Mode Enable			
			[4:3]	-	reserved			
			5	TXQTXIE	TX Queue TX Interrupt Enable			
			6	-	reserved			
			7	TXQIM	TX Queue Interrupt Mode			
			[12: 8]	TXQDC	TX Queue Depth Configuration			
			[15: 13]	-	reserved			
			16	TXQFIE	TXQ Full interrupt Enable			
			17	TXQOFRXIE	TXQ One Frame Reception Interrupt Enable			
			18	TXQOFTXIE	TXQ One Frame Transmission Interrupt Enable			

			[23: 19]	-	reserved			
			24	QMEIE	TXQ Message lost Error Interrupt Enable for FFI Mode			
			25	QOWEIE	TXQ Message overwrite Error Interrupt Enable for FFI Mode			
			[31: 26]	-	reserved			
CFDTXQCC3 <i>n</i>	<i>m_mbctrl</i>	TX Queue3 Configur ation / Control Register <i>n</i>						
			0	TXQE	TX Queue Enable			
			1	-	reserved			
			2	TXQOWE	TX Queue Overwrite Mode Enable			
			[4:3]	-	reserved			
			5	TXQTXIE	TX Queue TX Interrupt Enable			
			6	-	reserved			
			7	TXQIM	TX Queue Interrupt Mode			
			[12: 8]	TXQDC	TX Queue Depth Configurati on			
			[17: 13]	-	reserved			
			18	TXQOFTXIE	TXQ One Frame Transmissi on Interrupt Enable			
			[31: 19]	-	reserved			
			[24: 19]	-	reserved			

			25	QOWEIE	TXQ Message overwrite Error Interrupt Enable for FFI Mode			
			[27: 26]	-	reserved			
			28	DMAE	DMA TX Transfer Enable for TXQ for FFI Mode			
			[31: 29]	-	reserved			
<i>CFDTXQSTS On</i>	<i>m_mbctrl</i>	<i>TX Queue0 Status Register n</i>						
			0	TXQEMP	TX Queue Empty			
			1	TXQFLL	TX Queue Full			
			2	TXQTXIF	TX Queue TX Interrupt Flag			
			[7:3]	-	reserved			
			[13: 8]	TXQMC	CHn TX Queue Message counter			
			[15: 14]	-	reserved			
			16	TXQFIF	TXQ Full Interrupt Flag			
			17	TXQOFRXIF	TXQ One Frame Reception Interrupt Flag			
			18	TXQOFTXIF	TXQ One Frame Transmissi on Interrupt Flag			
			19	TXQMLT	TXQ Message Lost			
			20	TXQMOW	TXQ message overwrite			

			[27: 21]	-	reserved			
			28	TXQDMASTS	DMA TX Transfer Status for TXQ0			
			[31: 29]	-	reserved			
CFDTXQSTS 1n	m_mbctrl	TX Queue1 Status Register n						
			0	TXQEMP	TX Queue Empty			
			1	TXQFLL	TX Queue Full			
			2	TXQTXIF	TX Queue TX Interrupt Flag			
			[7:3]	-	reserved			
			[13: 8]	TXQMC	CHn TX Queue Message counter			
			[15: 14]	-	reserved			
			16	TXQFIF	TXQ Full Interrupt Flag			
			17	TXQOFRXIF	TXQ One Frame Reception Interrupt Flag			
			18	TXQOFTXIF	TXQ One Frame Transmission Interrupt Flag			
			19	TXQMLT	TXQ Message Lost			
			20	TXQMOW	TXQ message overwrite			
			[31: 21]	-	reserved			
CFDTXQSTS 2n	m_mbctrl	TX Queue2 Status Register n						

			0	<i>TXQEMP</i>	<i>TX Queue Empty</i>			
			1	<i>TXQFLL</i>	<i>TX Queue Full</i>			
			2	<i>TXQTXIF</i>	<i>TX Queue TX Interrupt Flag</i>			
			[7:3]	-	<i>reserved</i>			
			[13:8]	<i>TXQMC</i>	<i>CHn TX Queue Message counter</i>			
			[15:14]	-	<i>reserved</i>			
			16	<i>TXQFIF</i>	<i>TXQ Full Interrupt Flag</i>			
			17	<i>TXQOFRXIF</i>	<i>TXQ One Frame Reception Interrupt Flag</i>			
			18	<i>TXQOFTXIF</i>	<i>TXQ One Frame Transmission Interrupt Flag</i>			
			19	<i>TXQMLT</i>	<i>TXQ Message Lost</i>			
			20	<i>TXQMOW</i>	<i>TXQ message overwrite</i>			
			[31:21]	-	<i>reserved</i>			
<hr/>								
<i>CFDTXQSTS</i> <i>3n</i>	<i>m_mbctrl</i>	<i>TX Queue3 Status Register n</i>						
			0	<i>TXQEMP</i>	<i>TX Queue Empty</i>			
			1	<i>TXQFLL</i>	<i>TX Queue Full</i>			
			2	<i>TXQTXIF</i>	<i>TX Queue TX Interrupt Flag</i>			
			[7:3]	-	<i>reserved</i>			
			[13:8]	<i>TXQMC</i>	<i>CHn TX Queue Message counter</i>			
			[17:14]	-	<i>reserved</i>			

			18	TXQOFTXIF	TXQ One Frame Transmission Interrupt Flag			
			19	-	reserved			
			20	TXQMOW	TXQ message overwrite			
			[27: 21]	-	reserved			
			28	TXQDMASTS	DMA TX Transfer Status for TXQ3			
			[31: 29]	-	reserved			
<hr/>								
CFDTXQPCT R0n	<i>m_mbctrl</i>	TX Queue0 Pointer Control Register n						
			[7:0]	TXQPC	Channel n TX Queue Pointer Control			
			[31: 8]	-	reserved			
<hr/>								
CFDTXQPCT R1n	<i>m_mbctrl</i>	TX Queue1 Pointer Control Register n						
			[7:0]	TXQPC	Channel n TX Queue Pointer Control			
			[31: 8]	-	reserved			
<hr/>								
CFDTXQPCT R2n	<i>m_mbctrl</i>	TX Queue2 Pointer Control Register n						
			[7:0]	TXQPC	Channel n TX Queue Pointer Control			
			[31: 8]	-	reserved			
<hr/>								

<i>CFDTXQPCT R3n</i>	<i>m_mbctrl</i>	<i>TX Queue3 Pointer Control Register n</i>						
			[7:0]	<i>TXQPC</i>	<i>Channel n TX Queue Pointer Control</i>			
			[31: 8]	-	<i>reserved</i>			
<hr/>								
<i>CFDTXQEST S</i>	<i>m_mbctrl</i>	<i>TX Queue Empty Status Registers</i>						
			[7:0]	<i>TXQEMP</i>	<i>TXQ empty Status</i>			
			[15: 8]	<i>TXQEMP</i>	<i>TXQ empty Status</i>			
			[23: 16]	<i>TXQEMP</i>	<i>TXQ empty Status</i>			
			[31: 24]	<i>TXQEMP</i>	<i>TXQ empty Status</i>			
<hr/>								
<i>CFDTXQFIS TS</i>	<i>m_mbctrl</i>	<i>TX Queue Full Interrupt Status Registers</i>						
			[2:0]	<i>TXQFULL[2:0]</i>	<i>TXQ Full Interrupt Status</i>			
			3	-	<i>reserved</i>			
			[6:4]	<i>TXQFULL[6:4]</i>	<i>TXQ Full Interrupt Status</i>			
			7	-	<i>reserved</i>			
			[10: 8]	<i>TXQFULL[10: 8]</i>	<i>TXQ Full Interrupt Status</i>			
			11	-	<i>reserved</i>			
			[14: 12]	<i>TXQFULL[14: 12]</i>	<i>TXQ Full Interrupt Status</i>			
			15	-	<i>reserved</i>			
			[18: 16]	<i>TXQFULL[18: 16]</i>	<i>TXQ Full Interrupt Status</i>			
			19	-	<i>reserved</i>			
			[22: 20]	<i>TXQFULL[22: 20]</i>	<i>TXQ Full Interrupt Status</i>			

			23	-	reserved			
			[26: 24]	TXQFULL[26: 24]	TXQ Full Interrupt Status			
			27	-	reserved			
			[30: 28]	TXQFULL[30: 28]	TXQ Full Interrupt Status			
			31	-	reserved			
CFDTXQMS TS	<i>m_mbctrl</i>	<i>TX Queue Message lost Status Registers</i>						
			[2:0]	TXQML[2:0]	TXQ message lost Status			
			3	-	reserved			
			[6:4]	TXQML[6:4]	TXQ message lost Status			
			7	-	reserved			
			[10: 8]	TXQML[10:8]	TXQ message lost Status			
			11	-	reserved			
			[14: 12]	TXQML[14:12]	TXQ message lost Status			
			15	-	reserved			
			[18: 16]	TXQML[18:16]	TXQ message lost Status			
			19	-	reserved			
			[22: 20]	TXQML[22:20]	TXQ message lost Status			
			23	-	reserved			
			[26: 24]	TXQML[26:24]	TXQ message lost Status			
			27	-	reserved			
			[30: 28]	TXQML[30:28]	TXQ Full Interrupt Status			
			31	-	reserved			
CFDTXQOW STS	<i>m_mbctrl</i>	<i>TX Queue Message Overwrite Status Registers</i>						

			[7:0]	TXQOW	TXQ message overwrite Status			
			[15:8]	TXQOW	TXQ message overwrite Status			
			[23:16]	TXQOW	TXQ message overwrite Status			
			[31:24]	TXQOW	TXQ message overwrite Status			
CFDTXQISTS	<i>m_mbctrl</i>	TX Queue Interrupt Status Registers						
			[7:0]	TXQISF	TXQ Interrupt Status Flag			
			[15:8]	TXQISF	TXQ Interrupt Status Flag			
			[23:16]	TXQISF	TXQ Interrupt Status Flag			
			[31:24]	TXQISF	TXQ Interrupt Status Flag			
CFDTXQOFTISTS	<i>m_mbctrl</i>	TX Queue One Frame TX Interrupt Status Registers						
			[7:0]	TXQOFTISF	TXQ One Frame Tx Interrupt Status Flag			
			[15:8]	TXQOFTISF	TXQ One Frame Tx Interrupt Status Flag			
			[23:16]	TXQOFTISF	TXQ One Frame Tx Interrupt Status Flag			
			[31:24]	TXQOFTISF	TXQ One Frame Tx			

					<i>Interrupt Status Flag</i>			
<i>CFDTXQOF RISTS</i>	<i>m_mbctrl</i>	<i>TX Queue One Frame RX Interrupt Status Registers</i>						
			[2:0]	<i>TXQOFRISF[2:0]</i>	<i>TXQ One Frame RX Interrupt Status Flag</i>			
			3	-	<i>reserved</i>			
			[6:4]	<i>TXQOFRISF[6:4]</i>	<i>TXQ One Frame RX Interrupt Status Flag</i>			
			7	-	<i>reserved</i>			
			[10:8]	<i>TXQOFRISF[10:8]</i>	<i>TXQ One Frame RX Interrupt Status Flag</i>			
			11	-	<i>reserved</i>			
			[14:12]	<i>TXQOFRISF[14:12]</i>	<i>TXQ One Frame RX Interrupt Status Flag</i>			
			15	-	<i>reserved</i>			
			[18:16]	<i>TXQOFRISF[18:16]</i>	<i>TXQ One Frame RX Interrupt Status Flag</i>			
			19	-	<i>reserved</i>			
			[22:20]	<i>TXQOFRISF[22:20]</i>	<i>TXQ One Frame RX Interrupt Status Flag</i>			
			23	-	<i>reserved</i>			
			[26:24]	<i>TXQOFRISF[26:24]</i>	<i>TXQ One Frame RX Interrupt Status Flag</i>			
			27	-	<i>reserved</i>			
			[30:28]	<i>TXQOFRISF[30:28]</i>	<i>TXQ One Frame RX Interrupt Status Flag</i>			
			31	-	<i>reserved</i>			
<i>CFDTXQFST S</i>	<i>m_mbctrl</i>	<i>TX Queue Full</i>						

		Status Registers						
			[7:0]	TXQFSF	TXQ Full Status			
			[15:8]	TXQFSF	TXQ Full Status			
			[23:16]	TXQFSF	TXQ Full Status			
			[31:24]	TXQFSF	TXQ Full Status			
13. TX History List Registers								
CFDTHLCCn	<i>m_thlfifou nit</i>	Channel n TX History List Configuration / Control Register						
			0	THLE	TX History List Enable			
			[7:1]	-	reserved			
			8	THLIE	TX History List Interrupt Enable			
			9	THLIM	TX History List Interrupt Mode			
			10	THLDTE	TX History List Dedicated TX Enable			
			11	THLDGE	TX History List Dedicated GW Enable			
			[23:12]	-	reserved			
			24	THLEIE	TX History List Entry Lost Interrupt Enable			
			[31:25]	-	reserved			
CFDHLSTS n	<i>m_thlfifou nit</i>	Channel n TX History List Status Register						
			0	THLEMP	TX History List Empty			

			1	<i>THLFLL</i>	<i>TX History List Full</i>			
			2	<i>THLELT</i>	<i>TX History List Entry Lost Flag</i>			
			3	<i>THLIF</i>	<i>TX History List Interrupt Flag</i>			
			[7:4]	-	<i>reserved</i>			
			[13:8]	<i>THLMC</i>	<i>TX History List Message Count</i>			
			[31:14]	-	<i>reserved</i>			
<i>CFDTHLPCTR</i>	<i>m_thlfifou</i> <i>nit</i>	<i>CHn TX History List pointer control register</i>						
			[7:0]	<i>THLPC</i>	<i>TX History List Pointer Control</i>			
			[31:8]	-	<i>reserved</i>			
14. TX Interrupt Status Register								
<i>CFDGTINTS</i> <i>TS0</i>	<i>m_interrupt</i>	<i>Global Interrupt Status Register 0</i>						
			0	<i>TSIFO</i>	<i>TX Successful Transmission Interrupt Flag Channel 0</i>			
			1	<i>TAIFO</i>	<i>TX Abortion Interrupt Flag Channel 0</i>			
			2	<i>TQIFO</i>	<i>TX Queue Interrupt Flag Channel 0</i>			
			3	<i>CFTIFO</i>	<i>COM FIFO TX/GW Mode Interrupt Flag Channel 0</i>			
			4	<i>THIFO</i>	<i>TX History List</i>			

					<i>Interrupt Channel 0</i>			
			5	<i>TQOIF0</i>	<i>TX Queue One Frame Transmission Interrupt Flag Channel 0</i>			
			6	<i>CFOTIF0</i>	<i>COM FIFO One Frame Transmission Interrupt Channel 0</i>			
			7	-	<i>reserved</i>			
			8	<i>TSIF1</i>	<i>TX Successful Transmission Interrupt Flag Channel 1</i>			
			9	<i>TAIF1</i>	<i>TX Abortion Interrupt Flag Channel 1</i>			
			10	<i>TQIF1</i>	<i>TX Queue Interrupt Flag Channel 1</i>			
			11	<i>CFTIF1</i>	<i>COM FIFO TX/GW Mode Interrupt Flag Channel 1</i>			
			12	<i>THIF1</i>	<i>TX History List Interrupt Channel 1</i>			
			13	<i>TQOIF1</i>	<i>TX Queue One Frame Transmission Interrupt Flag Channel 1</i>			
			14	<i>CFOTIF1</i>	<i>COM FIFO One Frame Transmission Interrupt Channel 1</i>			
			15	-	<i>reserved</i>			
			16	<i>TSIF2</i>	<i>TX Successful Transmission Interrupt Flag Channel 2</i>			

			17	<i>TAIF2</i>	<i>TX Abortion Interrupt Flag Channel 2</i>			
			18	<i>TQIF2</i>	<i>TX Queue Interrupt Flag Channel 2</i>			
			19	<i>CFTIF2</i>	<i>COM FIFO TX/GW Mode Interrupt Flag Channel 2</i>			
			20	<i>THIF2</i>	<i>TX History List Interrupt Channel 2</i>			
			21	<i>TQOFIF2</i>	<i>TX Queue One Frame Transmissi on Interrupt Flag Channel 2</i>			
			22	<i>CFOTIF2</i>	<i>COM FIFO One Frame Transmissi on Interrupt Channel 2</i>			
			23	-	<i>reserved</i>			
			24	<i>TSIF3</i>	<i>TX Successful Transmissi on Interrupt Flag Channel 3</i>			
			25	<i>TAIF3</i>	<i>TX Abortion Interrupt Flag Channel 3</i>			
			26	<i>TQIF3</i>	<i>TX Queue Interrupt Flag Channel 3</i>			
			27	<i>CFTIF3</i>	<i>COM FIFO TX/GW Mode Interrupt Flag Channel 3</i>			
			28	<i>THIF3</i>	<i>TX History List Interrupt Channel 3</i>			

			29	TQOFIF3	TX Queue One Frame Transmissio n Interrupt Flag Channel 3			
			30	CFOTIF3	COM FIFO One Frame Transmissio n Interrupt Channel 3			
			31	-	reserved			
CFDTINTS TS1	m_interrupt	Global Interrupt Status Register 1						
			0	TSIF4	TX Successful Transmissi on Interrupt Flag Channel 4			
			1	TAIF4	TX Abortion Interrupt Flag Channel 4			
			2	TQIF4	TX Queue Interrupt Flag Channel 4			
			3	CFTIF4	COM FIFO TX/GW Mode Interrupt Flag Channel 4			
			4	THIF4	TX History List Interrupt Channel 4			
			5	TQOFIF4	TX Queue One Frame Transmissio n Interrupt Flag Channel 4			
			6	CFOTIF4	COM FIFO One Frame Transmissio n Interrupt Channel 4			
			7	-	reserved			
			8	TSIF5	TX Successful			

					<i>Transmiss on Interrupt Flag Channel 5</i>			
			9	<i>TAIF5</i>	<i>TX Abortion Interrupt Flag Channel 5</i>			
			10	<i>TQIF5</i>	<i>TX Queue Interrupt Flag Channel 5</i>			
			11	<i>CFTIF5</i>	<i>COM FIFO TX/GW Mode Interrupt Flag Channel 5</i>			
			12	<i>THIF5</i>	<i>TX History List Interrupt Channel 5</i>			
			13	<i>TQOFIF5</i>	<i>TX Queue One Frame Transmissi on Interrupt Flag Channel 5</i>			
			14	<i>CFOTIF5</i>	<i>COM FIFO One Frame Transmissi on Interrupt Channel 5</i>			
			15	-	<i>reserved</i>			
			16	<i>TSIF6</i>	<i>TX Successful Transmissi on Interrupt Flag Channel 6</i>			
			17	<i>TAIF6</i>	<i>TX Abortion Interrupt Flag Channel 6</i>			
			18	<i>TQIF6</i>	<i>TX Queue Interrupt Flag Channel 6</i>			
			19	<i>CFTIF6</i>	<i>COM FIFO TX/GW Mode Interrupt Flag Channel 6</i>			

			20	<i>THIF6</i>	<i>TX History List Interrupt Channel 6</i>			
			21	<i>TQOFIF6</i>	<i>TX Queue One Frame Transmission Interrupt Flag Channel 6</i>			
			22	<i>CFOTIF6</i>	<i>COM FIFO One Frame Transmission Interrupt Channel 6</i>			
			23	-	<i>reserved</i>			
			24	<i>TSIF7</i>	<i>TX Successful Transmission Interrupt Flag Channel 7</i>			
			25	<i>TAIF7</i>	<i>TX Abortion Interrupt Flag Channel 7</i>			
			26	<i>TQIF7</i>	<i>TX Queue Interrupt Flag Channel 7</i>			
			27	<i>CFTIF7</i>	<i>COM FIFO TX/GW Mode Interrupt Flag Channel 7</i>			
			28	<i>THIF7</i>	<i>TX History List Interrupt Channel 7</i>			
			29	<i>TQOFIF7</i>	<i>TX Queue One Frame Transmission Interrupt Flag Channel 7</i>			
			30	<i>CFOTIF7</i>	<i>COM FIFO One Frame Transmission Interrupt Channel 7</i>			
			31	-	<i>reserved</i>			
15. Global Test / Evaluation / Diag. Control Register								
<i>CFDGTSTCF G</i>	<i>m_test</i>	<i>Global Test</i>						

		<i>Configur ation Register</i>						
			0	<i>C0ICBCE</i>	<i>Channel 0 Internal CAN Bus Commun ication Test Mode Enable</i>			
			1	<i>C1ICBCE</i>	<i>Channel 1 Internal CAN Bus Commun ication Test Mode Enable</i>			
			2	<i>C2ICBCE</i>	<i>Channel 2 Internal CAN Bus Commun ication Test Mode Enable</i>			
			3	<i>C3ICBCE</i>	<i>Channel 3 Internal CAN Bus Commun ication Test Mode Enable</i>			
			4	<i>C4ICBCE</i>	<i>Channel 4 Internal CAN Bus Commun ication Test Mode Enable</i>			
			5	<i>C5ICBCE</i>	<i>Channel 5 Internal CAN Bus Commun ication Test Mode Enable</i>			
			6	<i>C6ICBCE</i>	<i>Channel 6 Internal CAN Bus Commun ication Test Mode Enable</i>			
			7	<i>C7ICBCE</i>	<i>Channel 7 Internal CAN Bus Commun ication Test</i>			

					<i>Mode Enable</i>			
			[15: 8]	-	<i>reserved</i>			
			[25: 16]	<i>RTMPS</i>	<i>RAM Test Mode Page Select</i>			
			[30: 26]	-	<i>reserved</i>			
			31	<i>PNFS</i>	<i>Pretended Network Filter List RAM Select</i>			
<i>CFDGTSTCR</i>	<i>m_test</i>	<i>Global Test Control Register</i>						
			0	<i>ICBCTME</i>	<i>Internal CAN Bus Communication Test Mode Enable</i>			
			1	-	<i>reserved</i>			
			2	<i>RTME</i>	<i>RAM Test Mode Enable</i>			
			[31: 3]	-	<i>reserved</i>			
<i>CFDGLOCKK</i>	<i>m_test</i>	<i>Global Lock Key Register</i>						
			[15: 0]	<i>LOCK</i>	<i>Lock Key</i>			
			[31: 16]	-	<i>reserved</i>			
<i>CFDGLOTB</i>	<i>m_otbfifio unit</i>	<i>OTB FIFO Configuration / Status Registers</i>						
			0	<i>OTBFE</i>	<i>OTB FIFO Enable</i>			
			[7:1]	-	<i>reserved</i>			
			8	<i>OTBEMP</i>	<i>OTB FIFO Empty</i>			
			9	<i>OTBFLL</i>	<i>OTB FIFO Full</i>			
			10	<i>OTBMLT</i>	<i>OTB FIFO Msg Lost Flag</i>			

			[15: 11]	OTBMC	OTB FIFO Message Count			
			[31: 16]	-	reserved			
16. Bus load counter Register								
CFDCnBLSTS	<i>m_busload</i>	BUS Load counter Register						
			[2:0]	-	reserved			
			[31: 3]	BLC	BUS load counter status			
CFDCnBLCT	<i>m_busload</i>	BUS Load counter control Register						
			0	BLCE	BUS load counter enable			
			[7: 1]	-	reserved			
			8	BLCLD	BUS load counter load			
			[31: 9]	-	reserved			
17. Flexible CAN mode Register								
CFDGFCMC	<i>m_comsfr</i>	Global Flexible CAN mode Configuration Register						
			0	FLXC0	Flexible CAN mode between Channel 0 and Channel 1			
			1	FLXC1	Flexible CAN mode between Channel 2 and Channel 3			
			2	FLXC2	Flexible CAN mode between Channel 4 and Channel 5			

			3	<i>FLXC3</i>	<i>Flexible CAN mode between Channel 6 and Channel 7</i>			
			[31: 4]	-	<i>reserved</i>			
18. Flexible transmission buffer assignment								
CFDGFTBAC	<i>m_bus_if</i>	<i>Global Flexible transmission buffer assignment Configuration Register</i>						
			[3:0]	<i>FLXMB0</i>	<i>Flexible transmission buffer assignment bitween Channel 0 and Channel 1</i>			
			[7:4]	-	<i>reserved</i>			
			[11: 8]	<i>FLXMB1</i>	<i>Flexible transmission buffer assignment bitween Channel 2 and Channel 3</i>			
			[15: 12]	-	<i>reserved</i>			
			[19: 16]	<i>FLXMB2</i>	<i>Flexible transmission buffer assignment bitween Channel 4 and Channel 5</i>			
			[23: 20]	-	<i>reserved</i>			
			[27: 24]	<i>FLXMB3</i>	<i>Flexible transmission buffer assignment bitween Channel 6 and Channel 7</i>			

			[31: 28]	-	reserved			
19. BUS Interface Register								
CFDGBISC	<i>m_bus_if</i>	Global Bus Interface Select Configur ation Register						
			0	IFSW	Select BUS Interface APB or R- ACE			
			[7:1]	-	reserved			
			[15: 8]	KEY	Key code			
			[31: 16]	-	reserved			
20. Reset Control Register								
CFDGRSTC	<i>m_bus_if</i>	Global SW reset Register						
			0	SRST	SW reset			
			[7:1]	-	reserved			
			[15: 8]	KEY	Key code			
			[31: 16]	-	reserved			
21. RX Interrupt Status Register (COMMON FIFO RX-GW mode & TXQ GW mode)								
CFDGRINTS TSn	<i>m_interru pt</i>	Global RX Interrupt Status Register <i>n</i>						
			[2:0]	QFIF	TXQ Full Interrupt Flag Channel <i>n</i>			
			[3]	-	reserved			
			[5:4]	BQFIF	Borrowed TXQ Full Interrupt Flag Channel <i>n</i>			
			[7:6]	-	reserved			
			[10: 8]	QOFRIF	TXQ One Frame RX Interrupt Flag Channel <i>n</i>			
			[11]	-	reserved			

			[13: 12]	BQOFRIF	Borrowed TXQ One Frame RX Interrupt Flag Channel n			
			[15: 14]	-	reserved			
			[18: 16]	CFRIF	Common FIFO RX Interrupt Flag Channel n			
			[23: 19]	-	reserved			
			[26: 24]	CFRFIF	Common FIFO One Frame FDC level Full Interrupt Flag Channel n			
			27	-	reserved			
			[30: 28]	CFOFRIF	Common FIFO One Frame RX Interrupt Flag Channel n			
			31	-	reserved			
22. Global AFL Ignore register (Entry & Control)								
CFDGAFLIGNENT	m_acsfr	Global RX Interrupt Status Register n						
			[8:0]	IRN	Ignore Rule Number			
			[15: 9]	-	reserved			
			[30: 16]	ICN	Ignore Channel Number			
			[31: 19]	-	reserved			
CFDGAFLIGNCTR	m_acsfr	Global AFL Ignore Control Register						

			0	<i>IREN</i>	<i>Ignore Rule Enable</i>			
			[7:1]	-	<i>reserved</i>			
			[15:8]	<i>KEY</i>	<i>These bits control the right or wrong of rewriting of a IREN bit.</i>			
			[31:16]	-	<i>reserved</i>			
23. Global Virtual Machine register								
<i>CFDGFFIMC</i>	<i>m_bus_if</i>	<i>Global Virtual Machine Mode configuration Register</i>						
			0	<i>FFIEN</i>	<i>FFI Mode Enable</i>			
			[7:1]	-	<i>reserved</i>			
			[15:8]	<i>KEY</i>	<i>These bits control the right or wrong of rewriting of FFIEN bit.</i>			
			[31:16]	-	<i>reserved</i>			
<i>CFDGVMEIS</i>	<i>m_bus_if</i>	<i>Global Virtual Machine Error Interrupt Select Register</i>						
			0	<i>FMLT</i>	<i>Selection of the output destination of FIFO message lost interrupt</i>			
			1	<i>TXQMLT</i>	<i>Selection of the output destination of TXQ message</i>			

					<i>lost interrupt</i>			
			2	<i>THLELT</i>	<i>Selection of the output destination of THL entry lost interrupt</i>			
			3	-	<i>reserved</i>			
			4	<i>CFMOW</i>	<i>Selection of the output destination of COMFIFO message overwrite interrupt</i>			
			5	<i>TXQOW</i>	<i>Selection of the output destination of TXQ message overwrite interrupt</i>			
			[31:6]	-	<i>reserved</i>			
<i>CFDVMCFG</i> <i>n</i>	<i>m_bus_if</i>	<i>Global Virtual Machine Common FIFO TXQ configuration Register</i>						
			[3:0]	<i>TXQ0VMN</i>	<i>VM number for TXQ0</i>			
			[7:4]	<i>TXQ1VMN</i>	<i>VM number for TXQ1</i>			
			[11:8]	<i>TXQ2VMN</i>	<i>VM number for TXQ2</i>			
			[15:12]	<i>TXQ3VMN</i>	<i>VM number for TXQ3</i>			
			[19:16]	<i>CF0VMN</i>	<i>VM number for COMFIFO 0</i>			

			[23: 20]	CF1VMN	VM number for COMFIFO 1			
			[27: 24]	CF2VMN	VM number for COMFIFO 2			
			[31: 28]	-	reserved			
CFDVMRFC FG	<i>m_bus_if</i>	Global Virtual Machine RX FIFO configura tion Register						
			[3:0]	VMN0	VM number for RXFIFO0			
			[7:4]	VMN1	VM number for RXFIFO1			
			[11: 8]	VMN2	VM number for RXFIFO2			
			[15: 12]	VMN3	VM number for RXFIFO3			
			[19: 16]	VMN4	VM number for RXFIFO4			
			[23: 20]	VMN5	VM number for RXFIFO5			
			[27: 24]	VMN6	VM number for RXFIFO6			
			[31: 28]	VMN7	VM number for RXFIFO7			
CFDVMISTS <i>n</i>	<i>m_bus_if</i>	Virtual Machine Interrupt Status Register						
			0	CFTXINT	COMFIFO TX Interrupt Flag			
			1	CFOFTXINT	COMFIFO One Frame TX Interrupt Flag			

			2	<i>TXQTXIF</i>	<i>TXQ TX Interrupt Flag</i>			
			3	<i>TXQOFTXIF</i>	<i>TXQ One Frame TX Interrupt Flag</i>			
			4	<i>THLIF</i>	<i>TX History List entry Interrupt Flag</i>			
			[7:5]	-	<i>reserved</i>			
			8	<i>RFIF</i>	<i>RXFIFO Interrupt Flag</i>			
			9	<i>RFFIF</i>	<i>RXFIFO Full Interrupt Flag</i>			
			10	<i>CFRXINT</i>	<i>COMFIFO RX Interrupt Flag</i>			
			11	<i>CFOFRXINT</i>	<i>COMFIFO One Frame RX Interrupt Flag</i>			
			12	<i>CFFIF</i>	<i>COMFIFO Full Interrupt Flag</i>			
			13	<i>TXQOFRXIF</i>	<i>TXQ One Frame RX Interrupt Flag</i>			
			14	<i>TXQFIF</i>	<i>TXQ Full Interrupt Flag</i>			
			15	-	<i>reserved</i>			
			16	<i>RFMLT</i>	<i>RXFIFO Message Lost</i>			
			17	<i>CFMLT</i>	<i>COMFIFO Message Lost</i>			
			18	<i>CFMOW</i>	<i>COMFIFO Message Overwrite</i>			
			19	<i>TXQMLT</i>	<i>TXQ Message Lost</i>			
			20	<i>TXQMOW</i>	<i>TXQ Message Overwrite</i>			

			21	THLELT	TX History List Entry Lost			
			[31:22]	-	reserved			
24. Pretended Network Filter (Entry & Control)								
CFDGPFLEC TR	<i>m_pnacsfr</i>	Pretended Network Filter List Entry control Register						
			[5:0]	PFLPN	Pretended Network Filter List Page Number			
			[7:6]	-	reserved			
			8	PFLDAE	Pretended Network Filter List Data Access Enable			
			[31:9]	-	reserved			
CFDGPFLCF G0	<i>m_pnacsfr</i>	Pretended Network Filter List Entry Configuration Register 0						
			[5:0]	RNC3	Rule Number for Channel 3			
			[7:6]	-	reserved			
			[13:8]	RNC2	Rule Number for Channel 2			
			[15:14]	-	reserved			
			[21:16]	RNC1	Rule Number for Channel 1			
			[23:22]	0	reserved			
			[29:24]	RNC0	Rule Number for Channel 0			
			[31:30]	0	reserved			

CFDGPFLCF G1	<i>m_pnacsfr</i>	Pretended Network Filter List Entry Configuration Register 1						
			[5:0]	RNC7	Rule Number for Channel 7			
			[7:6]	-	reserved			
			[13: 8]	RNC6	Rule Number for Channel 6			
			[15: 14]	-	reserved			
			[21: 16]	RNC5	Rule Number for Channel 5			
			[23: 22]	0	reserved			
			[29: 24]	RNC4	Rule Number for Channel 4			
			[31: 30]	0	reserved			
24. Global Acceptance Filter List								
CFDGAFLIDr	AFLRAM	Global Acceptan ce Filter List ID Registers r						
			[28: 0]	GAFLID	Global Acceptanc e Filter List Entry ID Field			
			[29]	GAFLLB	Global Acceptanc e Filter List Entry Loopback Configurati on			
			[30]	GAFLRTR	Global Acceptanc e Filter List Entry RTR Field			
			[31]	GAFLIDE	Global Acceptanc			

					<i>e Filter List Entry IDE Field</i>			
<i>CFDGAFLMr</i>	<i>AFLRAM</i>	<i>Global Acceptance Filter List Mask Registers r</i>						
			[28:0]	<i>GAFLIDM</i>	<i>Global Acceptance Filter List ID Mask Field</i>			
			[29]	<i>GAFLIFL1</i>	<i>Global Acceptance Filter List Information Label 1</i>			
			[30]	<i>GAFLRTRM</i>	<i>Global Acceptance Filter List Entry RTR Mask</i>			
			[31]	<i>GAFLIDEM</i>	<i>Global Acceptance Filter List IDE Mask</i>			
<i>CFDGFLP0r</i>	<i>AFLRAM</i>	<i>Global Acceptance Filter List Pointer 0 Registers r</i>						
			[3:0]	<i>GAFLDLC</i>	<i>Global Acceptance Filter List DLC Field</i>			
			[4]	<i>GAFLSRD0</i>	<i>Global Acceptance Filter List Select Routing destination 0</i>			
			[5]	<i>GAFLSRD1</i>	<i>Global Acceptance Filter List Select Routing destination 1</i>			
			[6]	<i>GAFLSRD2</i>	<i>Global Acceptanc</i>			

					e Filter List Select Routing destination 2			
			[7]	GAFLIFL0	Global Acceptanc e Filter List Information Label 0			
			[14: 8]	GAFLRMDP	Global Acceptanc e Filter List RX Message Buffer Direction Pointer			
			[15]	GAFLRMV	Global Acceptanc e Filter List RX Message Buffer Valid			
			[31: 16]	GAFLPTR	Global Acceptanc e Filter List Pointer Field			
CFDGAFLP1 r	AFLRAM	Global Acceptan ce Filter List Pointer 1 Registers r						
			[7:0]	GAFLFDP	Global Acceptanc e Filter List FIFO Direction Pointer			
			[13: 8]	GAFLFDP	Global Acceptanc e Filter List FIFO Direction Pointer			
			[19: 14]	GAFLFDP	Global Acceptanc e Filter List FIFO Direction Pointer			

			[25: 20]	<i>GAFLFDP</i>	<i>Global Acceptance Filter List FIFO Direction Pointer</i>			
			[31: 26]	<i>GAFLFDP</i>	<i>Global Acceptance Filter List FIFO Direction Pointer</i>			
25. Global Pretended Network Filter List								
<i>CFDGPFLIDs</i>	<i>PNFRAM</i>	<i>Global Pretended Network Filter List ID Registers</i>						
			[28: 0]	<i>GPFLID</i>	<i>Global Pretended Network Filter List Entry ID Field</i>			
			[29]	<i>GPFLLB</i>	<i>Global Pretended Network Filter List Entry Loopback Configuration</i>			
			[30]	<i>GPFLRTR</i>	<i>Global Pretended Network Filter List Entry RTR Field</i>			
			[31]	<i>GPFLIDE</i>	<i>Global Pretended Network Filter List Entry IDE Field</i>			
<i>CFDGPFLMs</i>	<i>PNFRAM</i>	<i>Global Pretended Network Filter List Mask</i>						

		<i>Registers r</i>						
			[28:0]	GPFLIDM	<i>Global Pretended Network Filter List ID Mask Field</i>			
			[29]	GPFLIFL1	<i>Global Pretended Network Filter List Information Label 1</i>			
			[30]	GPFLRTRM	<i>Global Pretended Network Filter List Entry RTR Mask</i>			
			[31]	GPFLIDEM	<i>Global Pretended Network Filter List IDE Mask</i>			
CFDGPFLP0s	PNFRAM	<i>Global Pretended Network Filter List Pointer 0 Registers r</i>						
			[3:0]	GPFLDLC	<i>Global Pretended Network Filter List DLC Field</i>			
			[4]	GPFLSRD0	<i>Global Pretended Network Filter List Select Routing destination 0</i>			
			[5]	GPFLSRD1	<i>Global Pretended Network Filter List Select Routing destination 1</i>			

			[6]	GPFLSRD2	Global Pretended Network Filter List Select Routing destination 2			
			[7]	GPFLIFL0	Global Pretended Network Filter List Information Label 0			
			[14: 8]	GPFLRMDP	Global Pretended Network Filter List RX Message Buffer Direction Pointer			
			[15]	GPFLRMV	Global Pretended Network Filter List RX Message Buffer Valid			
			[31: 16]	GPFLPTR	Global Pretended Network Filter List Pointer Field			
CFDGPFLP1 s	PNFRAM	Global Pretende d Network Filter List Pointer 1 Registers r						
			[7:0]	GPFLFDP	Global Pretended Network Filter List FIFO Direction Pointer			
			[13: 8]	GPFLFDP	Global Pretended Network Filter List FIFO			

					<i>Direction Pointer</i>			
			[19: 14]	GPFLFDP	<i>Global Pretended Network Filter List FIFO Direction Pointer</i>			
			[25: 20]	GPFLFDP	<i>Global Pretended Network Filter List FIFO Direction Pointer</i>			
			[31: 26]	GPFLFDP	<i>Global Pretended Network Filter List FIFO Direction Pointer</i>			
<i>CFDGPFLPTs</i>	<i>PNFRAM</i>	<i>Global Pretended Network Filter List Filter Type Registers</i>						
			[3:0]	GPFLOFFSET1	<i>Global Pretended Network filter offset value of the filter1</i>			
			[12: 4]	-	<i>reserved</i>			
			[13]	GPFLOUT1	<i>Global Pretended Network filter conditions of upper / lower filter of the filter1</i>			
			[14]	GPFLRANG1	<i>Global Pretended Network filter comparison conditions</i>			

					<i>of the filter1</i>			
			[15]	-	<i>reserved</i>			
			[19: 16]	GPFLOFFSE T0	<i>Global Pretended Network filter offset value of the filter0</i>			
			[28: 20]	-	<i>reserved</i>			
			[29]	GPFLOUT0	<i>Global Pretended Network filter conditions of upper / lower filter of the filter0</i>			
			[30]	GPFLRANG0	<i>Global Pretended Network filter comparison conditions of the filter0</i>			
			[31]	GPFLANDOR	<i>Global Pretended Network filter conditions of the filters 0 and 1</i>			
CFDGPFLPD 0s	PNFRAM	<i>Global Pretended Network Filter List Payload Data 0 Registers</i>						
			[31: 0]	FDATA	<i>Pretended Network Filter List Filter data</i>			
CFDGPFLPD 1s	PNFRAM	<i>Global Pretended Network Filter List</i>						

		Payload Data 1 Registers						
			[31: 0]	FDATA	Pretended Network Filter List Filter data			
CFDGPFLP M0s	PNFRAM	Global Pretended Network Filter List Payload Mask 0 Registers						
			[31: 0]	FMASK	Pretended Network Filter List Filter data mask field			
CFDGPFLP M1s	PNFRAM	Global Pretended Network Filter List Payload Mask 1 Registers						
			[31: 0]	FMASK	Pretended Network Filter List Filter data mask field			
26. RX Message Buffer Component b								
CFDRMID	MRAM	RX Message Buffer ID Registers						
			[28: 0]	RMID	RX Message Buffer ID Field			
			[29]	-	reserved			
			[30]	RMRTR	RX Message Buffer RTR Bit			This bit has description of CANFD .

								<i>It is necessary to delete this description.</i>
			[31]	<i>RMIDE</i>	<i>RX Message Buffer IDE Bit</i>			
<i>CFDRMPTR</i>	<i>MRAM</i>	<i>RX Message Buffer Pointer Registers</i>						
			[15: 0]	<i>RMTS</i>	<i>RX Message Buffer Timestamp Field</i>			
			[27: 16]	-	<i>reserved</i>			
			[31: 28]	<i>RMDLC</i>	<i>RX Message Buffer DLC Field</i>			<i>This bit has description of RMPLS. It is necessary to delete this description.</i>
<i>CFDRMFDS TS</i>	<i>MRAM</i>	<i>RX Message Buffer CAN-FD Status Register</i>						
			[0]	<i>RMESI</i>	<i>Error State Indicator bit</i>	x	x	
			[1]	<i>RMBRS</i>	<i>Bit Rate Switch bit</i>	x	x	
			[2]	<i>RMFDF</i>	<i>CAN FD Format bit</i>	x	x	
			[7:3]	-	<i>reserved</i>			
			[9:8]	<i>RMIFL</i>	<i>RX Message Buffer Information label Field</i>			

			[15: 10]	-	reserved			
			[31: 16]	RMPTR	RX Message Buffer Pointer Field			
CFDRMDF p	MRAM	RX Message Buffer Data Field p Registers						$p=[0..1]$
			[7:0]	RMDB($(p^q) + (q-4)$)	RX Message Buffer Data Byte $((p^q) + (q-4))$			
			[15: 8]	RMDB($(p^q) + (q-3)$)	RX Message Buffer Data Byte $((p^q) + (q-3))$			
			[23: 16]	RMDB($(p^q) + (q-2)$)	RX Message Buffer Data Byte $((p^q) + (q-2))$			
			[31: 24]	RMDB($(p^q) + (q-1)$)	RX Message Buffer Data Byte $((p^q) + (q-1))$			
27. RX FIFO Access Message Buffer Component b								
CFDRFID	MRAM	RX FIFO Access ID Registers						
			[28: 0]	RFID	RX FIFO Buffer ID Field			
			[29]	-	reserved			
			[30]	RFTR	RX FIFO Buffer RTR Bit			This bit has descrip- tion of CANFD It is necess

								ary to delete this descrip- tion.
			[31]	RFIDE	RX FIFO Buffer IDE Bit			
CFDRFPTR	MRAM	RX FIFO Access Pointer Registers						
			[15: 0]	RFTS	RX FIFO Timestamp Value			
			[27: 16]	-	reserved			
			[31: 28]	RFDLC	RX FIFO Buffer DLC Field			
CFDRFFDSTS	MRAM	RX FIFO Access CAN-FD Status Register						
			[0]	RFESI	Error State Indicator bit	x	x	
			[1]	RFBRS	Bit Rate Switch bit	x	x	
			[2]	RFFDF	CAN FD Format bit	x	x	
			[7:3]	-	reserved			
			[9:8]	RFIFL	RX FIFO Buffer Information label Field			
			[15: 10]	-	reserved			
			[31: 16]	RFPTR	RX FIFO Buffer Pointer Field			
CFDRFDFp	MRAM	RX FIFO Access Data Field p Registers						p=[0..1]
			[7:0]	RFDB((p*q)+(q-4))	RX FIFO Buffer Data Byte ((p*q)+(q-4))			

			[15: 8]	RFDB($(p^*q)+(q-3)$)	RX FIFO Buffer Data Byte ($((p^*q)+(q-3))$)			
			[23: 16]	RFDB($(p^*q)+(q-2)$)	RX FIFO Buffer Data Byte ($((p^*q)+(q-2))$)			
			[31: 24]	RFDB($(p^*q)+(q-1)$)	RX FIFO Buffer Data Byte ($((p^*q)+(q-1))$)			
28. Common FIFO Access Message Buffer Component b								
CFDCFID	MRAM	Common FIFO Access ID Registers						
			[28: 0]	CFID	Common FIFO Buffer ID Field			
			[29]	THLEN	THL Entry enable			
			[30]	CFRTR	Common FIFO Buffer RTR Bit			This bit has description of CANFD. It is necessary to delete this description.
			[31]	CFIDE	Common FIFO Buffer IDE Bit			
CFDCPTR	MRAM	Common FIFO Access Pointer Registers						
			[15: 0]	CFTS	Common FIFO Timestamp Value			

			[27: 16]	-	reserved			
			[31: 28]	CFDLC	Common FIFO Buffer DLC Field			
CFDCFFDCS TS	MRAM	Common FIFO Access CAN-FD Control Status Register						
			[0]	CFESI	Error State Indicator bit	x	x	
			[1]	CFBRS	Bit Rate Switch bit	x	x	
			[2]	CFFDF	CAN FD Format bit	x	x	
			[7:3]	-	reserved			
			[9:8]	CFIFL	Common FIFO Buffer Information label Field			
			[15: 10]	-	reserved			
			[31: 16]	CFPTR	Common FIFO Buffer Pointer Field			
CFDCFDFp	MRAM	Common FIFO Access Data Field p Registers						p=[0..1]
			[7:0]	CFDB((p*q)+(q-4))	Common FIFO Buffer Data Byte ((p*q)+(q- 4))			This bit has descrip- tion of CFPLS. It is neces- sary to delete this descrip- tion.
			[15: 8]	CFDB((p*q)+(q-3))	Common FIFO Buffer Data Byte			This bit has descrip- tion of

					$((p^q)+(q-3))$			CFPLS. It is necessary to delete this description.
			[23:16]	CFDB($(p^q)+(q-2)$)	Common FIFO Buffer Data Byte $((p^q)+(q-2))$			This bit has description of CFPLS. It is necessary to delete this description.
			[31:24]	CFDB($(p^q)+(q-1)$)	Common FIFO Buffer Data Byte $((p^q)+(q-1))$			This bit has description of CFPLS. It is necessary to delete this description.
29. TX Message Buffer Component b								
CFDTMID	MRAM	TX Message Buffer ID Registers						
			[28:0]	T MID	TX Message Buffer ID Field			
			[29]	THLEN	Tx History List Entry			
			[30]	TMRTR	TX Message Buffer RTR Bit			This bit has description of CANFD. It is necessary to delete this description.

			[31]	<i>TMIDE</i>	<i>TX Message Buffer IDE Bit</i>			
<i>CFDTMPTR</i>	<i>MRAM</i>	<i>TX Message Buffer Pointer Registers</i>						
			[27:0]	-	<i>reserved</i>			
			[31:28]	<i>TMDLC</i>	<i>TX Message Buffer DLC Field</i>			
<i>CFDTMFDC TR</i>	<i>MRAM</i>	<i>TX Message Buffer CAN-FD Status Register</i>						
			[0]	<i>TMESI</i>	<i>Error State Indicator bit</i>	x	x	
			[1]	<i>TMBRS</i>	<i>Bit Rate Switch bit</i>	x	x	
			[2]	<i>TMFDF</i>	<i>CAN FD Format bit</i>	x	x	
			[7:3]	-	<i>reserved</i>			
			[9:8]	<i>TMIFL</i>	<i>TX Message Buffer Information label Field</i>			
			[15:10]	-	<i>reserved</i>			
			[31:16]	<i>TMPTR</i>	<i>TX Message Buffer Pointer Field</i>			
<i>CFDTMDFp</i>	<i>MRAM</i>	<i>TX Message Buffer Data Field p Registers</i>						<i>p=[0..1] *1</i>
			[7:0]	<i>TMDB((p*q)+(q-4))</i>	<i>TX Message Buffer Data Byte ((p*q)+(q-4))</i>			

			[15: 8]	TMDB((p*q)+(q-3))	TX Message Buffer Data Byte ((p*q)+(q-3))			
			[23: 16]	TMDB((p*q)+(q-2))	TX Message Buffer Data Byte ((p*q)+(q-2))			
			[31: 24]	TMDB((p*q)+(q-1))	TX Message Buffer Data Byte ((p*q)+(q-1))			
30. TX History List								
CFDTHLACC 0	MRAM	TX History List Access Registers 0						
			[2:0]	BT	Buffer Type			
			[9:3]	BN	Buffer No.			
			[11: 10]	-	Reserved			
			[14: 12]	CH	Channel No.			
			[15]	TGW	Transmit Gateway Buffer indication			
			[31: 16]	TMTS	Transmit Timestamp			
CFDTHLACC 1	MRAM	TX History List Access Registers 1						
			[15: 0]	TID	Transmit ID			
			[17: 16]	TIFL	Transmit Information Label			
			[31: 18]	-	Reserved			
31. RAM Test Page								

<i>CFDRPGAC Ck</i>	<i>RAM</i>	<i>RAM Test Page Access Registers</i>						
			[31: 0]	<i>RDTA</i>	<i>RAM Data Test Access</i>			

X : unused bit (should not write 1 / read value is unknown)

*1 Although the comment field of **CFDTMDFp** is $p=[0..1]$, when using TXQ by DMA, the writing to $p=[15]$ is required.

Next, the portion deleted subsequent to section 6 is shown.

6 Initialisation

It is necessary to delete the explanation portion of the data bit rate.

Table6.3 is deleted.

Section 6.1.5 is deleted.

DTSEG1, DTSEG2, DSJW is deleted in Figure 6.3

7 Acceptance Filtering Function using Global Acceptance Filter List (AFL)

Description of CFDGCFG.CMPOC is deleted.

Description of CFDGERFL.CMPOF is deleted.

Description of CFDRMN.B.RMPLS, CFDRFCCa.RFPLS and CFDCFCCd.CFPLS are deleted.

8 FIFO Buffers & Normal MB Configuration

Description of CFDGCFG.CMPOC is deleted.

Description of CFDRFCCa.RFPLS and CFDCFCCd.CFPLS are deleted.

Description of "FIFO payload data size" is deleted.

Section 8.2.1.4 is deleted.

9 Interrupts and DMA

9.1 Interrupts

Description of "CAN-FD Message Payload overflow flag" is deleted.

Description of CFDGCTR.CMPOFIE and CFDGERFL.CMPOF are deleted.

Description of CFDCnCTR.TDCVFIE and CFDCnFDSTS.TDCVF are deleted.

9.2 DMA Transfer

Description of CFDRFCCa.RFPLS and CFDCFCCd.CFPLS are deleted.

10 Reception and Transmission

10.1.3 Timestamp

Description of "CAN-FD frame" is deleted.

10.2.3.2 GW FIFO Operation

Description of ESI is deleted.

Section 10.2.6 is deleted.

11 ECC Check

Description of CFDRMND.RMPLS, CFDRFCCa.RFPLS and CFDCFCCd.CFPLS are deleted.

Since payload size is fixed, reexamination of the calculation of ECC error part is needed.

12 Test Mode

Section 12.1.5 is deleted.

12.2.2.1 CRC Error Test

Description of CFDCyCRC.CRCREG is deleted.

13 RAM area configuration

Description of CFDRMND.RMPLS, CFDRFCCa.RFPLS and CFDCFCCd.CFPLS are deleted.

Payload size is calculated by 8Byte fixation.

13.1 Examples

Description of CFDRMND.RMPLS, CFDRFCCa.RFPLS and CFDCFCCd.CFPLS are deleted.

Payload size is calculated by 8Byte fixation.

6 Initialisation

Before joining CAN communications the following shall be configured:

Clock setting

Bit timing setting (nominal and data rate)

Baud Rate setting (nominal and data rate)

CAN-FD setting

Acceptance Filter setting (configuration of Global Acceptance Filter List)

Reception-, Transmission- and GW-FIFO setting

CAN Operation Mode setting

6.1 Initialization of CAN Clock, Bit Timing and Baud Rate

6.1.1 Bit Timing Conditions

The following lines describe the composition of each segment and the limitations that apply to the segment setting.

(1) Each segment setting

SS = Fixed to 1 TQ

TSEG1 = Refer to (**CFDCnNCFG**) and (**CFDCnDCFG**)

TSEG2 = Refer to (**CFDCnNCFG**) and (**CFDCnDCFG**)

SJW = Refer to (**CFDCnNCFG**) and (**CFDCnDCFG**)

SS + TSEG1 + TSEG2 = 5 to 49 TQs for Data Bit Rate and 8 to 385 for Nominal Bit Rate

(2) Limitations on TSEG1, TSEG2 and SJW

TSEG1(N) > TSEG2(N) \geq SJW(N)

TSEG1(D) \geq TSEG2(D) \geq SJW(D)

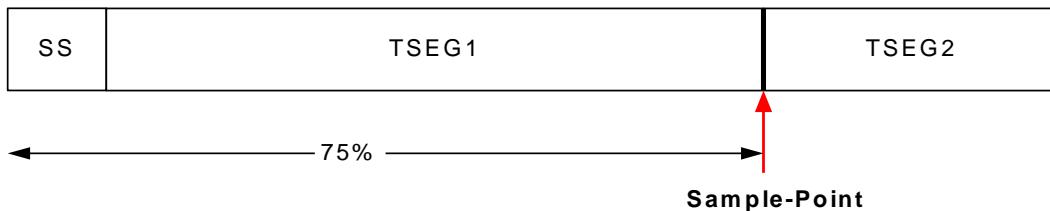
When only classical frames will be used, the user should configure the bit fields TSEG1 and TSEG2 of **CFDCnDCFG** to valid values.

Table 6.1 shows an example of how to set the bit timing to achieve required Sample Point settings.

1 bit	Set value (TQ)				Sample point ^{*1} (%)
	SS	TSEG1	TSEG2	SJW	
5TQ	1	2	2	1	60.00
8TQ	1	4	3	1	62.50
	1	5	2	1	75.00
10TQ	1	6	3	1	70.00
	1	7	2	1	80.00
12TQ	1	8	3	1	75.00
	1	9	2	1	83.33
15TQ	1	10	4	1	73.33
	1	11	3	1	80.00
16TQ	1	10	5	1	68.75
	1	11	4	1	75.00
20TQ	1	12	7	1	65.00
	1	13	6	1	70.00
24TQ	1	15	8	1	66.66
	1	16	7	1	70.83
50TQ	1	39	10	4	80.00

Table 6.1 – Bit timing examples

*1: Sample point (in case of 75%)



6.1.2 CAN Bit Timing

In the CAN protocol, each bit in a communication frame is composed of three segments, which can be configured individually for each channel via the related **CFDCnNCFG** and **CFDCnDCFG** registers.

Figure 6.1 shows the segment composition of a bit and the sample point in it.

Of these segments, the Time Segment 1 (hereafter TSEG1) and Time Segment 2 (hereafter TSEG2) are used to specify the position of the sample point, so that the timing at which each bit on the CAN Bus is sampled can be altered by changing the values of these segments.

The minimum resolution for this timing is referred to as Time Quantum (hereafter TQ), which is determined by the clock frequency supplied to the CAN channel and the divide-by-N value of the Baud Rate Prescaler (nominal and data rate).

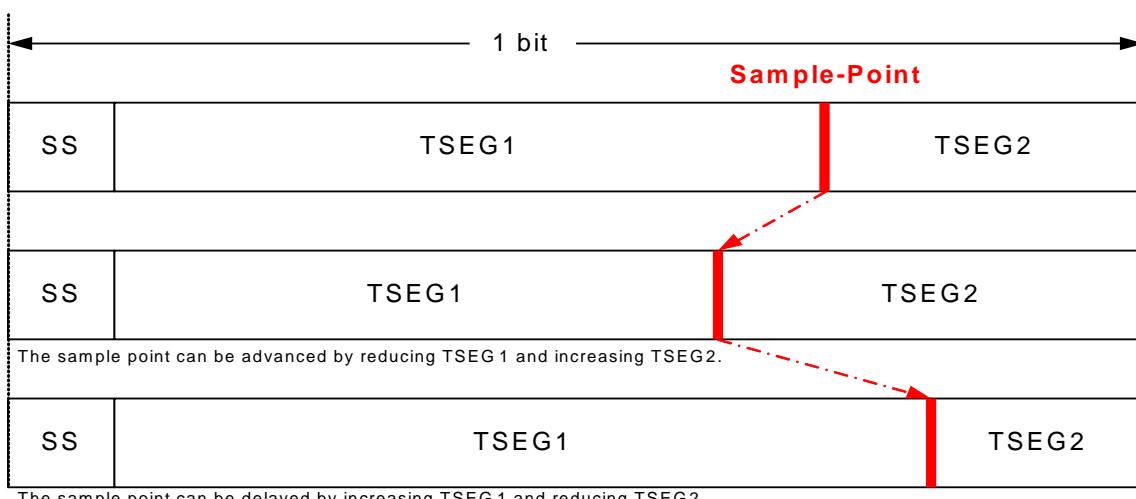


Figure 6.1 Segment composition of a bit and the sample point

1. SS: Synchronization Segment

This segment is used to synchronize bits by monitoring a recessive-to-dominant edge during the Interframe Space (comprised of Intermission, Suspend Transmission, and Bus Idle; during Bus Idle, all nodes can start transmission)

2. TSEG1: Time Segment 1

This segment absorbs physical delays on the CAN network. A physical delay on the network is two times the total sum of a bus delay, input comparator delay, and output driver delay. It can be lengthened by SJW.

3. TSEG2: Time Segment 2

This segment is used to correct a phase error by performing resynchronization. It can be shortened by SJW (While sending or receiving a message, communication frames between some nodes may get out of sync due to a drift in the oscillator frequency or a delay in the transmission path. This is referred to as a phase error).

4. SJW: Resynchronization Jump Width

This is the maximum width by which bits that have become out of sync due to a phase error may be corrected.

The above figure reports only one symbolic sample point.

6.1.3 Baud Rate

Either the CAN channel system clock (Clean clock) or the External oscillator clock can be selected globally for all CAN channels as CAN communication clock.

The transfer speed is determined by the DLL Clock, the divide-by-N value of the Baud Rate Prescaler, and the number of TQs in one bit.

$$\text{baudrate} = \frac{\text{DLL_Clock}}{(\text{number_of_time_quanta_per_bit}) \times (\text{BRP}+1)}$$

Figure 6.2 shows a block diagram of the circuit that generates the CAN channel system clock and Table 6.2 shows a Baud Rate examples.

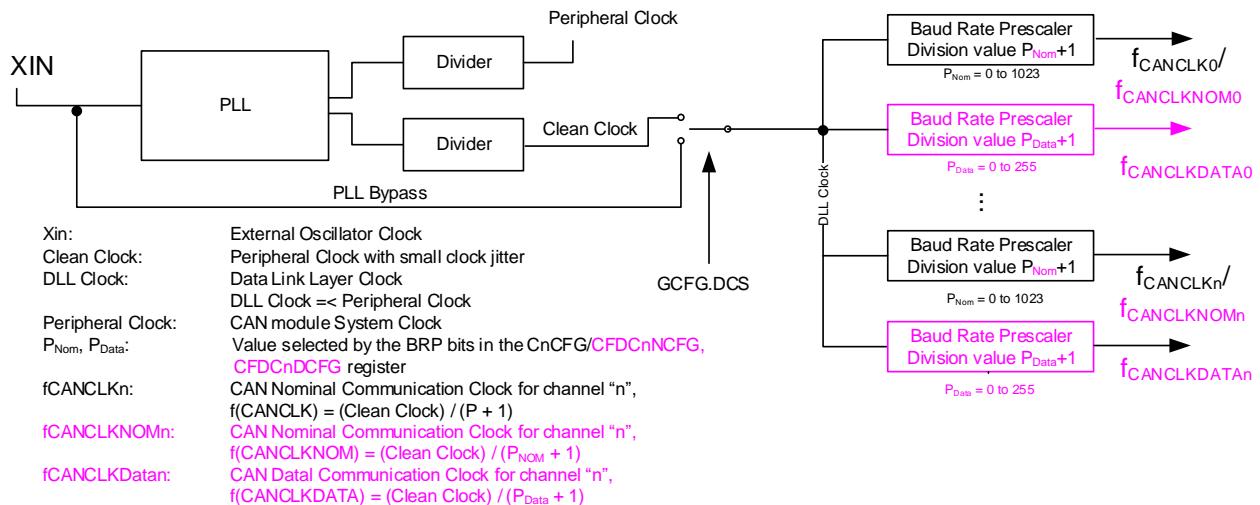


Figure 6.2 Block diagram of the circuit that generates the CAN channel communication clock

Baud rate calculation formula	(DLL Clock) (baud rate prescaler divide-by-N value ¹) x (number of TQs in one bit)									
	80MHz	40MHz	32MHz	30MHz	24MHz	20MHz	16MHz	10MHz	8MHz ³	
1Mbps	8TQ (10) 20TQ (4)	8TQ (5) 20TQ (2)	8TQ (4) 16TQ (2)	10TQ (3) 15TQ (2)	8TQ (3) 12TQ (2) 24TQ (1)	10TQ (2) 20TQ (1)	8TQ (2) 16TQ (1)	10TQ (1)	8TQ (1)	
500Kbps	8TQ (20) 20TQ (8)	8TQ (10) 20TQ (4)	8TQ (8) 16TQ (4)	10TQ (6) 15TQ (4) 20TQ (3)	8TQ (6) 12TQ (4) 24TQ (2)	10TQ (4) 20TQ (2)	8TQ (4) 16TQ (2)	10TQ (2) 20TQ (1)	8TQ (2) 16TQ (1)	
250Kbps	8TQ (40) 20TQ (16)	8TQ (20) 20TQ (8)	8TQ (16) 16TQ (8)	10TQ (12) 15TQ (8) 20TQ (6)	8TQ (12) 12TQ (8) 24TQ (4)	10TQ (8) 20TQ (4)	8TQ (8) 16TQ (4)	10TQ (4) 20TQ (2)	8TQ (4) 16TQ (2)	
125Kbps	8TQ (80) 20TQ (32)	8TQ (40) 20TQ (16)	8TQ (32) 16TQ (16)	10TQ (24) 15TQ (16) 20TQ (12)	8TQ (24) 12TQ (16) 24TQ (8)	10TQ (16) 20TQ (8)	8TQ (16) 16TQ (8)	10TQ (8) 20TQ (4)	8TQ (8) 16TQ (4)	
83.3Kbps	8TQ (120) 12TQ (80) 16TQ (60) 24TQ (40)	8TQ (60) 12TQ (40) 16TQ (30) 24TQ (20)	8TQ (48) 12TQ (32) 16TQ (24) 24TQ (16)	8TQ (45) 10TQ (36) 12TQ (30) 15TQ (24) 20TQ (18) 24TQ (15)	8TQ (36) 12TQ (24) 16TQ (18) 24TQ (12)	8TQ (30) 10TQ (24) 12TQ (20) 15TQ (16) 16TQ (15) 20TQ (12) 24TQ (10)	8TQ (24) 12TQ (16) 16TQ (12) 24TQ (8)	8TQ (15) 10TQ (12) 12TQ (10) 15TQ (8) 20TQ (6) 24TQ (5)	8TQ (12)	
33.3Kbps	8TQ (300) 12TQ (200) 16TQ (150) 20TQ (120) 24TQ (100)	8TQ (150) 12TQ (100) 16TQ (75) 20TQ (60) 24TQ (50)	8TQ (120) 10TQ (96) 12TQ (80) 15TQ (64) 16TQ (60)	10TQ (90) 12TQ (75) 15TQ (60) 20TQ (45)	8TQ (90) 10TQ (72) 12TQ (60) 15TQ (48) 16TQ (45)	8TQ (75) 10TQ (60) 12TQ (50) 15TQ (40) 20TQ (36) 24TQ (30)	8TQ (60) 10TQ (48) 12TQ (40) 15TQ (32) 16TQ (30) 20TQ (24) 24TQ (20)	10TQ (30) 12TQ (25) 15TQ (20) 20TQ (15)	8TQ (30)	

Table 6.2 – Nominal Baud Rate calculation formula and example CAN communication configurations

Baud rate calculation formula	(DLL Clock) (baud rate prescaler divide-by-N value ^{*1}) x (number of TQs in one bit)		
	80MHz	40MHz	20MHz
Nominal 1Mbps Data 8Mbps	80TQ (1)	40TQ (1)	20TQ (1)
	10TQ (1)	5TQ (1)	Not possible
Nominal 1Mbps Data 5Mbps	80TQ (1)	40TQ (1)	20TQ (1)
	16TQ (1)	8TQ (1)	Not possible
Nominal 500Kbps Data 2Mbps	160TQ (1)	80TQ (1)	40TQ (1)
	40TQ (1)	20TQ (1)	10TQ (1)

Table 6.3 – Baud Rate calculation example for nominal and data bitrate CAN communication configurations

*1: Baud Rate Prescaler divide-by-N value = P + 1 (P = 0-1023) P: value selected by the BRP bits in the Channel Configuration Registers.

*2: Shown in () are the Baud Rate Prescaler divide-by-N values

*3: Minimum Frequency to achieve max. Nominal Baud Rate of 1Mbps

For optimum clock tolerance in networks using the FD frame format, the length of the time quantum should be

the same in the nominal bit time and in the data bit time; means **CFDCnNCFG.NBRP = CFDCnDCFG.DBRP**.

Further if Transceiver Delay Compensation is used then the **CFDCnDCFG.DBRP** shall not be programmed greater than 1; 1 means divide by 2.

6.1.4 Setting of CAN Clock, Bit Timing and Baud Rate

Figure 6.3 shows the procedure for setting the CAN clock and the Baud Rate for each channel.

These settings should be performed during Channel Reset Mode (Configuration Mode) for the CAN channels.

Before going to channel communication state the Baud Rate must be configured, otherwise the mode will not switch correctly.

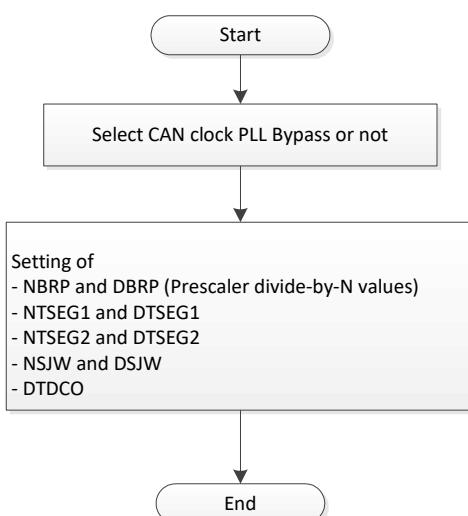


Figure 6.3 Procedure for setting the CAN bit timing and Baud Rate

6.1.5 Transmitter delay compensation

In case a high Baud Rate is used such as 5 to 8 Mbps for the Data phase, the Transmitter delay could become greater than TSEG1. In this case the transmitter would always detect a bit-error in the data-phase of the CAN-FD frame. The TDC compensates the transmitter's inability to receive its own transmitted bit at the sample point of that bit.

There is another symbolic sample point known as the Secondary Sample Point (SSP) that is used only during the Data phase of CAN-FD frames. This is derived from the Transceiver Delay Compensation Result (**CFDCnFDSTS.TDCR**) as shown in Figure 6.4.

The resolution of the configuration, measured and offset values is based on the CAN channel DLL clock.

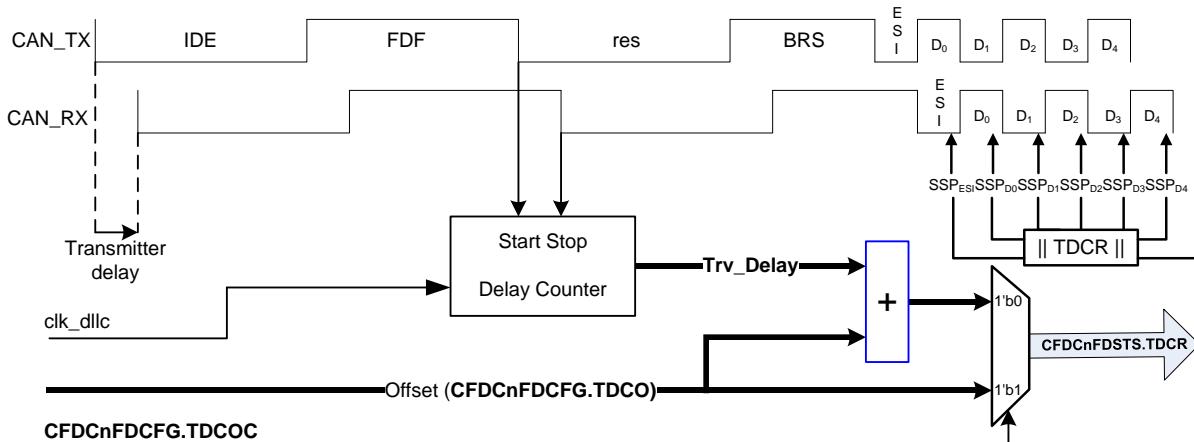


Figure 6.4 Transmitter Delay Compensation

The measured Trv_Delay is based on integer number of clk_dlc clock cycles. The delay is counted up by one for each started clock until the dominant value is seen on CAN_RX. The figure below illustrates the measured result. Trv_Delay counted to maximum 127 with a clk_dlc clock.

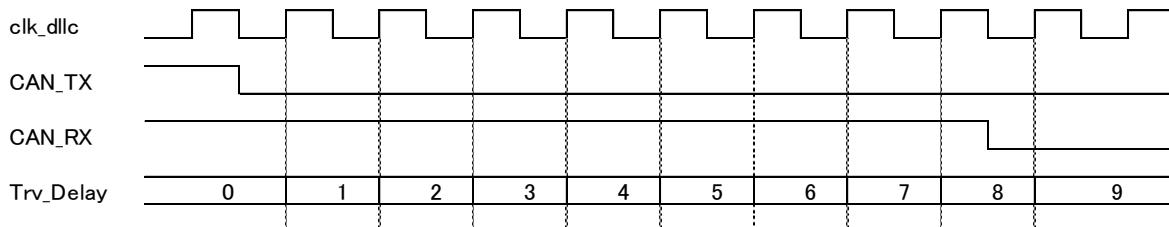


Figure 6.5 Trv_Delay measurement example

Transmitter delay is measured at the falling-edge (transition) from FDF bit to res bit. However, if a bit error is detected in the FDF bit or in the res bit, then, the transmitter delay is not measured.

The transmitter delay is also not measured if a falling edge is detected on CAN_RX in the critical time zone after Sample Point of FDF bit and before the transmitter starts to send its dominant res (TSEG2 phase of FDF bit).

If a falling-edge is detected in this critical time zone, then, it seems as if there is no delay between CAN_TX and CAN_RX. A meaningful measurement of the delay is not possible in this case and hence the measurement is suppressed.

From system point of view, this suppressed measurement will result in the transmitter node sending an error frame to recover from this unexpected behaviour.

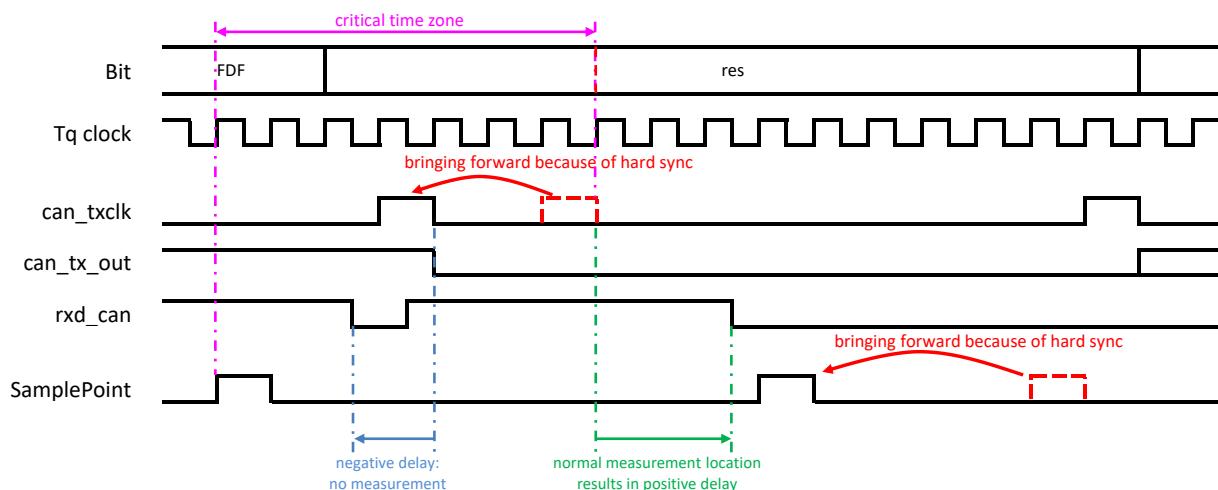


Figure 6.6 No measurement when negative delay

The SSP is calculated by taking the result from **CFDCnFDSTS.TDCR** and rounding the value down to the nearest integer number of data time quanta.

The Figure 6.7 illustrates the positioning of the secondary sample point. In case the **CFDCnFDCFG.TDCOC** is equal to 1'b0 the SSP is equal to the Trv_Delay (measured delay) + **CFDCnFDCFG.TDCO**, rounded down to the nearest integer number of time quanta. Usually the TDCO value should have the size of (**SyncSegment_{data}** + **TSEG1_{data}**) to position the SSP to the theoretical location of a sample point.

If the **CFDCnFDCFG.TDCOC** is equal to 1'b1 then the SSP is just defined by the **CFDCnFDCFG.TDCO**. If the **CFDCnDCFG.DBRP** > zero then the value will also be rounded down to the nearest integer number of time quanta.

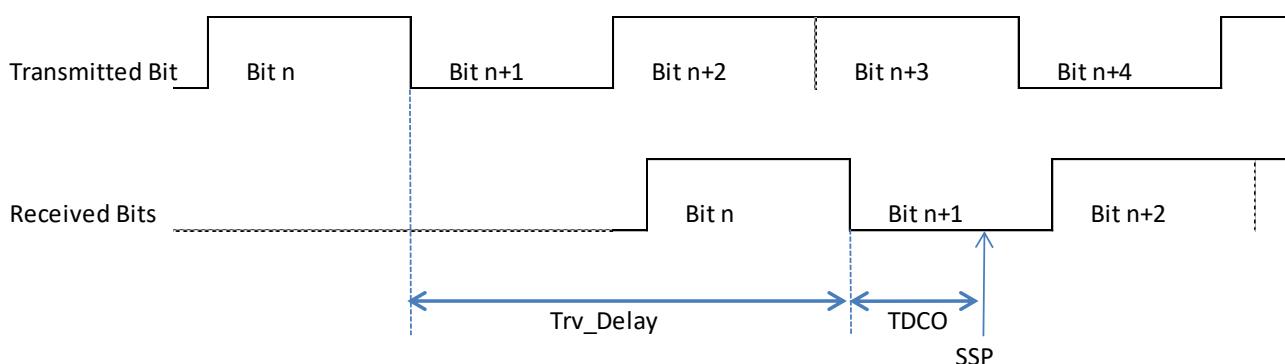


Figure 6.7 Position of the Secondary Sample Point

The maximum delay (Trv_Delay + TDCO) which can be compensated by the RS-CAN-FD module is (6 data bits – 2clk_dlc).

The ISO 11898-1 allows the users to set different values for BRP_data and BRP_nom.

If different values are used for **CFDCnNCFG.NBRP** and **CFDCnDCFG.DBRP**, then 2 CAN nodes may be out of synchronisation at the point when the bit rate changes from nominal bit rate to data bit rate after Sample point of the BRS bit. This condition is shown in Figure 6.8 below.

The length of the time quantum should be the same in the nominal bit time and in the data bit time; means **CFDCnNCFG.NBRP** = **CFDCnDCFG.DBRP**.

Different Bitrates can be achieved by selecting different configuration values for the Time Segments. Note the Nominal Bitrate can be configured from 8 to 385TQs and the Data Bit rate from 5 to 49TQs.

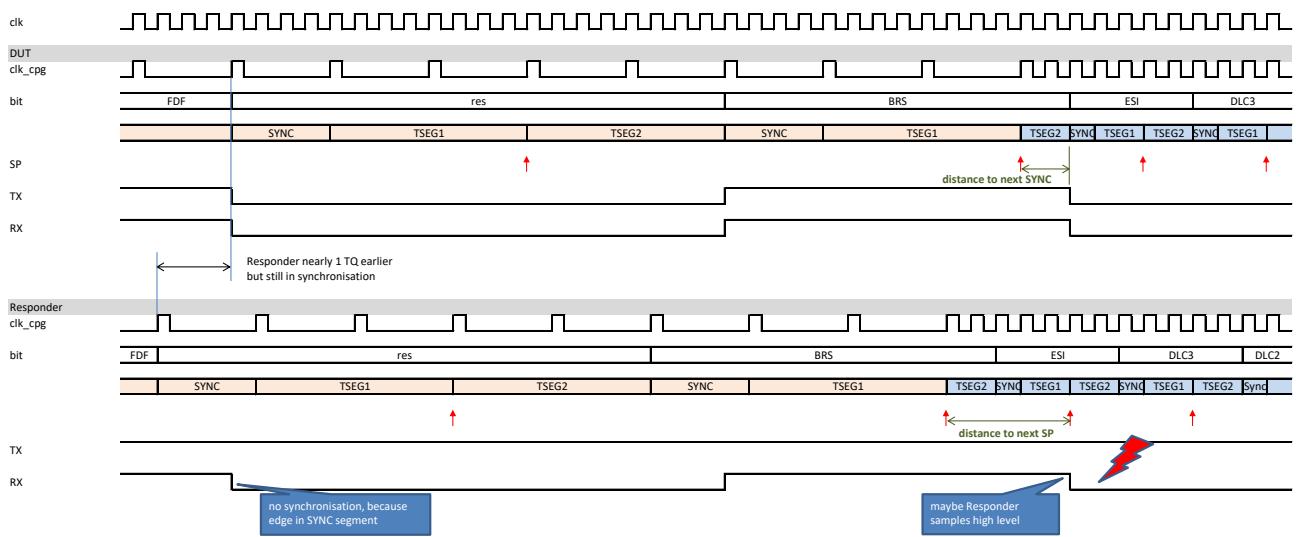


Figure 6.8 Loss of synchronization between 2 CAN nodes

The transmitter delay compensation measurement result is updated at the falling edge from FDF bit to res bit when configured accordingly (**CFDCnFDCFG.TDCE** = 1'b1, **CFDCnFDCFG.TDCOC** = 1'b0)

Figure 6.9 shows the read flow to get the measured transmitter delay compensation result.

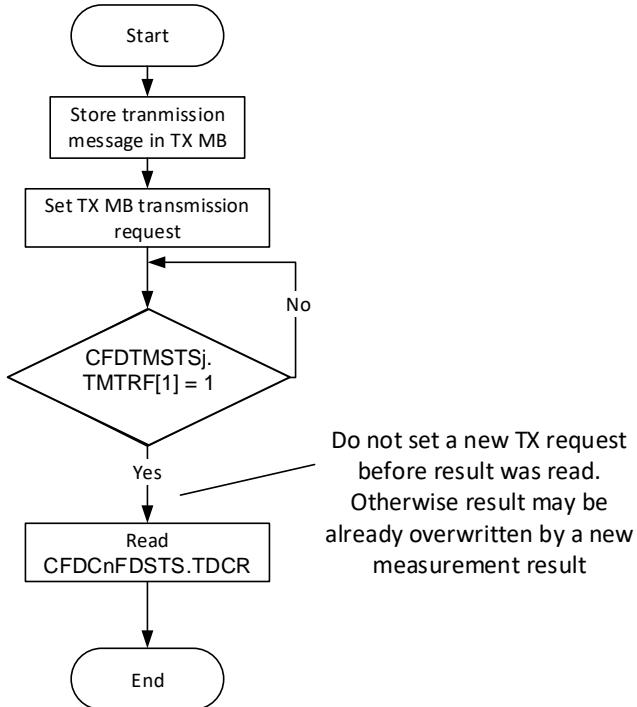


Figure 6.9 TDC result read flow

6.2 CAN Module Configuration after H/W Reset

After hardware reset (power on reset) or after setting and clearing a CFDGRSTC.SRST bit, the RS-CAN-FD module enters Global Sleep Mode automatically.

To enable configuration of the RS-CAN-FD module, the Sleep Mode has to be left by clearing the Global Sleep Request bit **CFDGCTR.GSLPR** to 1'b0.

After MCU HW reset the module starts the RAM initialisation, the **CFDGSTS.GRAMINIT** bit in the Global Status Register is set automatically indicating that the RS-CAN-FD logic is initialising the RAM.

After the RAM initialisation is completed, this bit is cleared automatically.

The RAM initialisation is necessary to avoid setting of false ECC error flag after HW reset due to random data present in the RAM.

The registers of RS-CAN-FD should not be accessed (in either read or write) until the RAM initialisation is complete and the **CFDGSTS.GRAMINIT** bit is cleared.

Before going to communication mode the Global Acceptance Filter List and message FIFO buffers must be configured. Furthermore each required CAN channel has to be configured (e.g. CAN bit timing)

For this all required CAN channels have to be released from channel Sleep Mode and have to be configured for communication in channel Reset Mode (Configuration Mode).

Figure 6.10 shows the configuration procedure. For details about each step, refer to Section 7, Section 8, Section 9 and Section 6.1.3.

The RS-CAN-FD module does not perform the RAM initialization sequence after executing SW Reset by setting CFDGRSTC.SRST.

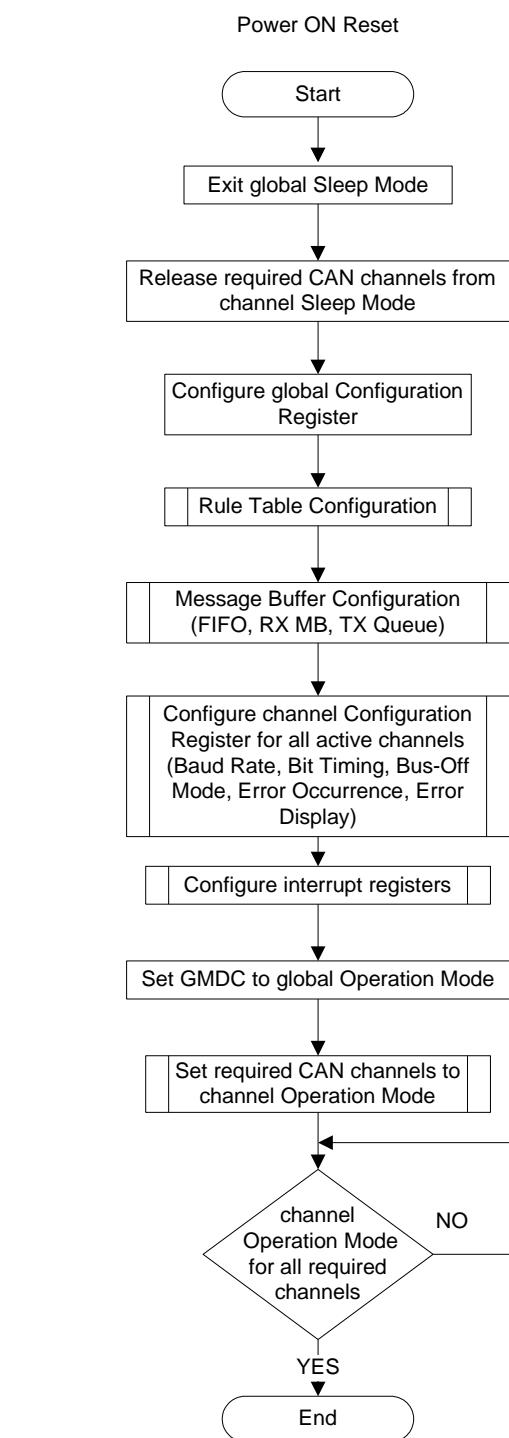


Figure 6.10 Configuration procedure after H/W Reset

7 Acceptance Filtering Function using Global Acceptance Filter List (AFL)

7.1 Overview

The RS-CAN-FD module can handle message acceptance filtering for all channels with a global Acceptance Filter List (called AFL). Each element of the AFL defines a filter rule for messages received on a specific channel.

Following actions will be done based on the AFL entries:

- acceptance filtering based on received CAN Identifier and masking
- DLC filtering based on received DLC value
- Message data payload according to the **CFDGCFG.CMPOC** bit
- storage of accepted messages in the Message Buffer objects defined in the related AFL entry
- attaching a 16 bit pointer to the stored messages defined in the related AFL entry e.g. to support AUTOSAR applications
- attaching a 2 bit information label to the stored messages defined in the related AFL entry

The 8-channel RS-CAN-FD module allows a maximum of 1536 AFL entries across all channels with a maximum of 384 AFL entries per single channel.

During acceptance filtering process, each AFL entry in a channel is checked against the received message by the acceptance filter unit. Check is starting from the lowest AFL entry number for this channel.

AFL search is stopped when a match of the received Identifier with a configured Identifier/Mask combination occurs or when the received Identifier has been compared against all AFL entries defined for the related channel. If no match occurs, then the received message is rejected. No notification is given to the application in this case.

Additionally, an automatic DLC filtering is performed for each accepted message if DLC Check is globally enabled. If the DLC value of the received message is equal to or higher than the configured DLC value in the matching AFL entry, then the DLC check is passed.

If DLC replacement (**CFDGCFG.DRE** bit) is enabled, DLC value configured in the matching AFL entry is greater than 4'h0 and DLC check passes, then the configured value of DLC in the matching AFL entry is stored in the destination RXMB or FIFO Buffer.

If the received value of DLC is greater than the configured DLC value in the matching AFL entry, then the additional Data Bytes received on the CAN Bus are not stored in the destination RXMB or FIFO Buffer.

These additional Data Bytes will be stored as 8'h0 in the destination RXMB or FIFO Buffer.

If DLC replacement is enabled and DLC value of matching AFL entry is 4'h0, then the received value of DLC will be stored in the destination RX MB or FIFO Buffer.

If DLC replacement (**CFDGCFG.DRE** bit) is disabled and DLC check passes, then the received value of DLC on the CAN Bus is stored in the destination RXMB or FIFO Buffer.

If the received value of DLC is greater than the configured DLC value in the matching AFL entry, then the additional Data Bytes received from the CAN Bus are also stored in the destination RXMB or FIFO Buffer.

If DLC value of the received message is less than the configured DLC value in the matching AFL entry, then DLC check fails. In this case, the received message is rejected and it is not stored in any RXMB or FIFO Buffer.

Additionally, DLC check failure will be flagged by the DLC Error Flag in the Global Error Flag Register. If configured, an error interrupt will also be generated.

The DLC replacement configuration has no impact if the DLC check fails.

If a message has passed both acceptance filtering and DLC filtering, it is stored in a single reception Message Buffer and/or in FIFO buffers configured for reception or gateway function.

This message storage target information is also defined in the same AFL entry. Users should not set a target at the AFL entry which is not configured.

Each accepted received message could be stored into a maximum of 8 different target destinations (single reception Message Buffer and/or FIFO buffers).

(Programming of more than 8 target destinations is not allowed. In the case more destinations are programmed then internal timing race condition can occur and received message may not be stored to the

Message RAM. Correct configuration of the numbers of Target destination is the responsibility of the application)

Further protection mechanism is made for the case when a received message contains more data payload Bytes than possible to store in the target destination (**CFDRMNB.RMPLS**, **CFDRFCCa.RFPLS** or **CFDCFCCd.CFPLS**).

If the **CFDGCFG.CMPOC** = 1'b0 then the message is completely rejected and will not be stored in the target destination. In the case of **CFDGCFG.CMPOC** = 1'b0 and RX or Common FIFO full and the received message contains more data payload Bytes than possible to store in the target destination (**CFDRMNB.RMPLS**, **CFDRFCCa.RFPLS** or **CFDCFCCd.CFPLS**), the corresponding **CFDFMSTS.RFxMLT** or **CFDFMSTS.CFxMLT** will not be set to 1'b1 respectively.

If the **CFDGCFG.CMPOC** = 1'b1 then the received Data Bytes greater than **CFDRMNB.RMPLS** will be rejected. In the case of **CFDGCFG.CMPOC** = 1'b1 and RX or Common FIFO full and the received message contains more data payload Bytes than possible to store in the target destination (**CFDRMNB.RMPLS**, **CFDRFCCa.RFPLS** or **CFDCFCCd.CFPLS**), the corresponding **CFDFMSTS.RFxMLT** or **CFDFMSTS.CFxMLT** will be set to 1'b1 respectively.

Depending on the **CFDGCFG.DRE** the original received DLC or the DLC value configured at the AFL entry will be stored.

Regardless of **CFDGCFG.CMPOC** configuration, **CFDGERFL.CMPOF** will be set to 1'b1 if payload overflow condition is detected.

The DLC filtering is performed before the payload overflow function. Hence for one reception frame only one flag could be set at the same time **CFDGERFL.DEF** or **CFDGERFL.CMPOF**.

7.2 Allocation of AFL entries to each CAN channel

The number of AFL entries per channel can be configured using the dedicated field in the related Global Acceptance Filter Configuration Registers (see Figure 7.1).

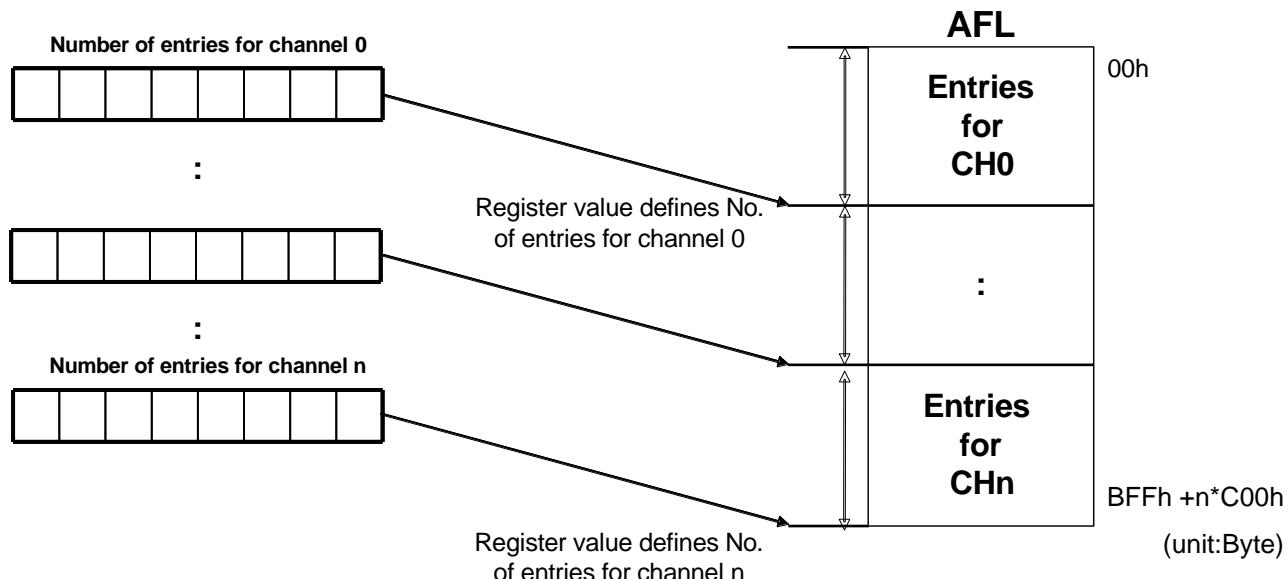


Figure 7.1 Configuration of AFL for each channel

The minimum number of entries for one channel is 0 (no entries defined for the channel), the maximum number of entries for one channel is 384. The total number of entries for all channels should not exceed the maximum limit of $(n+1)*192$.

All entries are unique for a channel and overlapping or sharing of entries is not supported. Correct configuration of the AFL is the responsibility of the application.

The RS-CAN-FD module will not flag errors related to the configuration of the AFL.

7.3 AFL entry description

Each AFL entry consists of 16 Bytes. The fields in all entries are identical.

Each entry contains the following information for acceptance filtering and DLC filtering:

Identifier (11 bits for Standard Frame Format, 29 bits for Extended Frame Format):

Acceptance filter unit will check Identifier field of received message against Identifier field of each AFL entry (full 29 bit masking of Identifier bits possible, see below).

IDE bit:

Acceptance filter unit will check IDE bit of received message against this bit and selects the relevant part of the Identifier field for acceptance filtering (masking of IDE bit possible, see below).

RTR bit:

Acceptance filter unit will only accept Data Frames (RTR = 0) or Remote Frames (RTR = 1) according to the setting of this bit (masking of RTR bit possible, see below).

Loopback Configuration bit:

This bit can enable/disable the AFL entry depending on the Loopback Configuration or Mirror Mode condition.

Mask for Identifier bits (29 bits):

Each bit in the Identifier Mask can mask the corresponding Identifier bit in the AFL entry during acceptance filtering (Figure 7.2).

Mask for IDE bit:

If this Mask bit masks the IDE bit of the AFL entry both Standard Identifier and Extended Identifier format messages can be accepted by this AFL entry. The identifier of the received message is compared against the Standard Identifier part of the AFL entry for Standard Identifier format messages and against the Extended Identifier part of the AFL entry for Extended Identifier format messages.

Mask for RTR bit:

If this Mask bit masks the RTR bit of the AFL entry both frame formats 'Data Frame' and 'Remote Frame' will be accepted by this AFL entry.

Pointer information (16 bits):

This 16 bit pointer will be attached to a received message accepted by the related AFL entry. The Pointer will be added during message storage in the Message Buffer area and can be used by application as support function. The pointer information could be used for example to support PDU Identifier allocation for the received message in AUTOSAR systems.

Information Label (2 bits):

This 2bit label will be attached to a received message accepted by the related AFL entry. The label will be added during message storage in the Message Buffer area and can be used by application as support function.

DLC value for automatic DLC filtering:

If the DLC value of the received message is equal or higher than the configured DLC value the DLC check is passed.

If the DLC value in this AFL entry is configured to 1'b0 DLC filtering is effectively disabled for this entry (all accepted messages will pass DLC filtering).

Each AFL entry contains the following information for the handling of received messages

Message Buffer number of one single reception Message Buffer as target for received message storage
Single reception Message Buffer enable bit to configure the single reception Message Buffer number as valid/invalid as target for received message storage

FIFO direction pointer: each bit of the FIFO direction pointer configures a dedicated FIFO as possible target for a received message

Note: A message received on channel "A" can be routed to Common FIFO Buffer of another channel and if this Common FIFO Buffer is configured in GW mode, then the message stored in this Common FIFO Buffer will be transmitted on that channel since Common FIFO Buffer is associated with channel.

There is no hardware protection against such storage of message. Hence the FIFO direction pointer should be configured carefully.

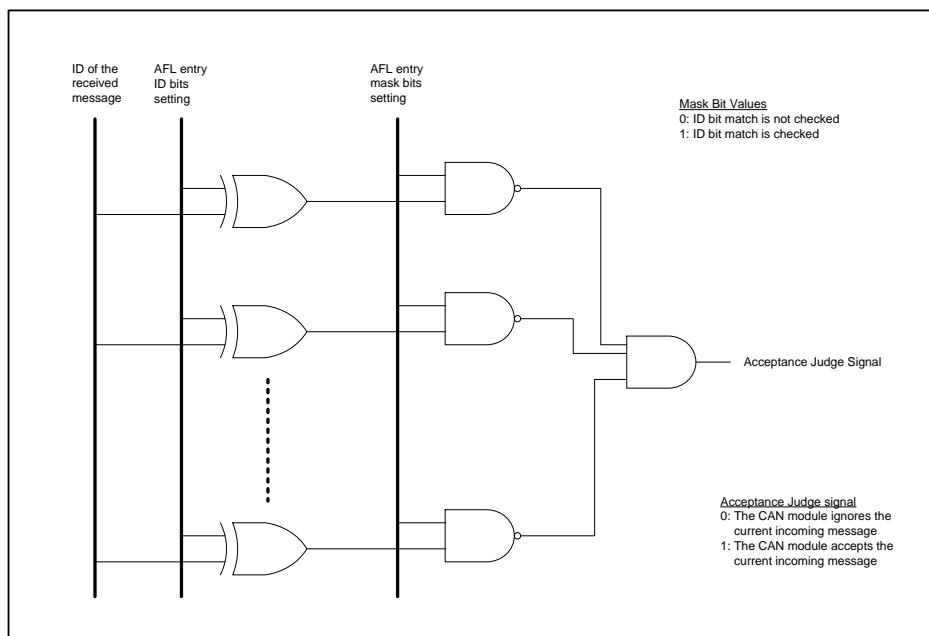


Figure 7.2 Acceptance Function

7.4 Entering entries in the AFL

Application SW can enter one full entry into the AFL via following registers:

Global AFL ID Entry Register:Part 1 of the AFL entry

Global AFL Mask Entry Register:Part 2 of the AFL entry

Global AFL Pointer 0 Entry Register:Part 3 of the AFL entry

Global AFL Pointer 1 Entry Register:Part 4 of the AFL entry

16 sets of these registers form a group of AFL entries. Each group can be accessed via a page mechanism. For the RS-CAN-FD module, 8-channel version, 96 of these pages exist to allow access to the whole AFL range. The AFL should only be configured in CH_RESET or CH_HALT. Pages are linked to the AFL entries in the following way:

Page 0	Entry 0 – 15
Page 1	Entry 16 – 31
Page 2	Entry 32 – 47
Page 3	Entry 48 – 63
Page 4	Entry 64 – 79
Page 5	Entry 80 – 95
Page 6	Entry 96 – 111
Page 7	Entry 112 – 127
Page 8	Entry 128 – 143
Page 9	Entry 144 – 159
Page 10	Entry 160 – 175
Page 11	Entry 176 – 191
Page 12	Entry 192 – 207
Page 13	Entry 208 – 223
Page 14	Entry 224 – 239
Page 15	Entry 240 – 255
Page 16	Entry 256 – 271
Page 17	Entry 272 – 287
Page 18	Entry 288 – 303
Page 19	Entry 304 – 319
Page 20	Entry 320 – 335
Page 21	Entry 336 – 351
Page 22	Entry 352 – 367
Page 23	Entry 368 – 383
Page 24	Entry 384 – 399
Page 25	Entry 400 – 415
Page 26	Entry 416 – 431
Page 27	Entry 432 – 447
Page 28	Entry 448 – 463
Page 29	Entry 464 – 479
Page 30	Entry 480 – 495
Page 31	Entry 496 – 511
Page 32	Entry 512 – 527

Page 33	Entry 528 – 543
Page 34	Entry 544 – 559
Page 35	Entry 560 – 575
Page 36	Entry 576 – 591
Page 37	Entry 592 – 607
Page 38	Entry 608 – 623
Page 39	Entry 624 – 639
Page 40	Entry 640 – 655
Page 41	Entry 656 – 671
Page 42	Entry 672 – 687
Page 43	Entry 688 – 703
Page 44	Entry 704 – 719
Page 45	Entry 720 – 735
Page 46	Entry 736 – 751
Page 47	Entry 752 – 767
Page 48	Entry 768 – 783
Page 49	Entry 784 – 799
Page 50	Entry 800 – 815
Page 51	Entry 816 – 831
Page 52	Entry 832 – 847
Page 53	Entry 848 – 863
Page 54	Entry 864 – 879
Page 55	Entry 880 – 895
Page 56	Entry 896 – 911
Page 57	Entry 912 – 927
Page 58	Entry 928 – 943
Page 59	Entry 944 – 959
Page 60	Entry 960 – 975
Page 61	Entry 976 – 991
Page 62	Entry 992 – 1007
Page 63	Entry 1008 – 1023
Page 64	Entry 1024 – 1039
Page 65	Entry 1040 – 1055
Page 66	Entry 1056 – 1071
Page 67	Entry 1072 – 1087
Page 68	Entry 1088 – 1103
Page 69	Entry 1104 – 1119
Page 70	Entry 1120 – 1135
Page 71	Entry 1136 – 1151
Page 72	Entry 1152 – 1167
Page 73	Entry 1168 – 1183
Page 74	Entry 1184 – 1199
Page 75	Entry 1200 – 1215
Page 76	Entry 1216 – 1231

Page 77	Entry 1232 – 1247
Page 78	Entry 1248 – 1263
Page 79	Entry 1264 – 1279
Page 80	Entry 1280 – 1295
Page 81	Entry 1296 – 1311
Page 82	Entry 1312 – 1327
Page 83	Entry 1328 – 1343
Page 84	Entry 1344 – 1359
Page 85	Entry 1360 – 1375
Page 86	Entry 1376 – 1391
Page 87	Entry 1392 – 1407
Page 88	Entry 1408 – 1423
Page 89	Entry 1424 – 1439
Page 90	Entry 1440 – 1455
Page 91	Entry 1456 – 1471
Page 92	Entry 1472 – 1487
Page 93	Entry 1488 – 1503
Page 94	Entry 1504 – 1519
Page 95	Entry 1520 – 1535

The selection of the AFL access page is done using the Global Acceptance Filter List Entry Control Register (**CFDGAFLECTR**) (Figure 7.3). This register has following fields:

7bits to select the AFL Page number

1 bit to enable / disable the AFL Data access to prevent unwanted write access to the AFL

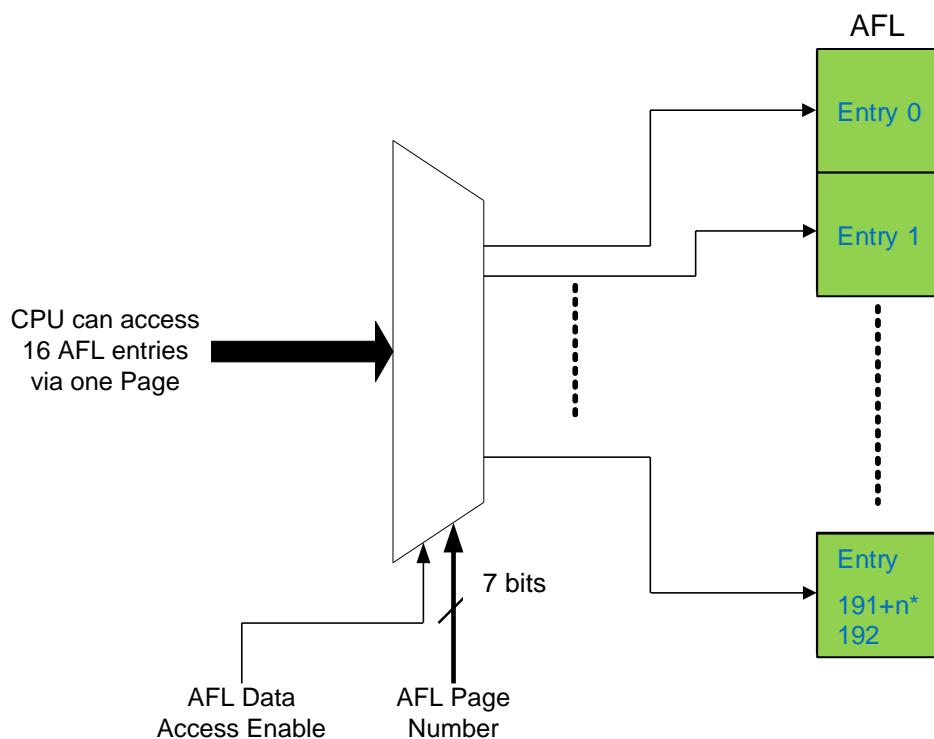


Figure 7.3 AFL page access

Application SW should not write numbers higher than 7'h5F for the AFL Page number.

The configuration flow shown in Figure 7.4 should be followed to program the AFL.

After entering all entries in Configuration Mode, locking of the AFL access should be performed to protect unwanted write access to the AFL.

Write protection is active during all global modes if the lock bit is set (GL_RESET, GL_HALT, and GL_OPERATION).

Read access to AFL is still possible during all global modes even if AFL data access is disabled (consistency check of AFL contents possible during runtime).

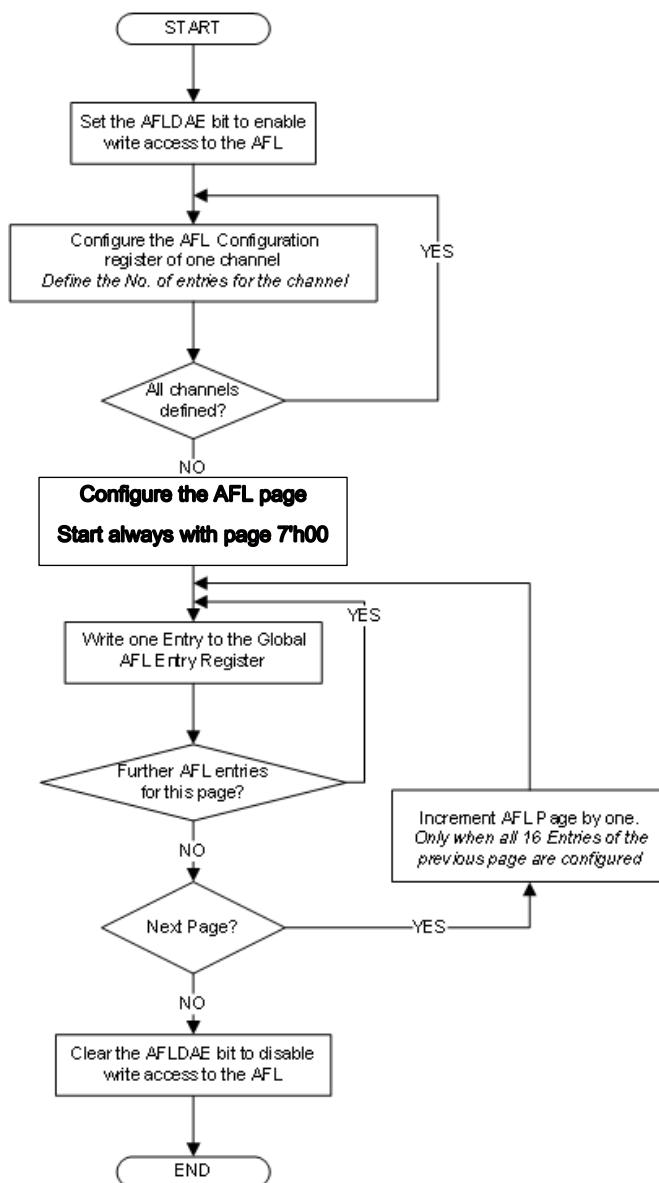


Figure 7.4 AFL configuration flow

7.5 Loopback modes

If the Loopback Configuration bit is set, the AFL entry is only valid in loopback test mode (Self test mode 0 or Self test mode 1) or in mirror mode when receiving messages that were transmitted by the respective CAN channel itself.

The AFL entry is not valid for received messages in loopback mode transmitted by other CAN nodes on the bus (the expression ‘valid or invalid’ for the related entry means that this AFL entry ‘will or will not’ be compared against the received message ID respectively).

If the Loopback Configuration bit is 1'b0 the AFL entry is only valid for received messages transmitted by other CAN nodes on the bus in normal (non-loopback mode) and mirror mode

received messages transmitted by other CAN nodes or the CAN channel itself in loopback test mode

The mirror mode can be enabled via the **CFDGCFG.MME** bit in the Global Configuration Register. If **CFDGCFG.MME** bit is set, then a successfully Transmitted message can be stored back in an RX MB or FIFO Buffer if a matching Entry is configured in the AFL for that channel.

The Loopback configuration bit in the matching AFL Entry must be set for storing this Frame.

If mirror mode and loopback test mode are configured at the same time the loopback test mode behaviour applies.

Table 7.1 shows the behaviour of the acceptance filter unit depending on the setting of the related input signals.

Mirror mode enable (MME configuration bit)	Loopback in test mode (Selftest Mode 0 or Selftest Mode 1)	Channel Mode is	Loopback configuration bit in AFL Entry	AFL entry is
0	0	Receiver	0	valid
			1	invalid
		Transmitter	0	invalid
			1	invalid
	1	Receiver	0	valid
			1	invalid
		Transmitter	0	valid
			1	valid
1	0	Receiver	0	valid
			1	invalid
		Transmitter	0	invalid
			1	valid
	1	Receiver	0	valid
			1	invalid
		Transmitter	0	valid
			1	valid

Table 7.1 Acceptance filter behaviour based on Loopback Configuration setting in AFL entry

Note: The expression ‘valid’ or ‘invalid’ for the related entry means that this AFL entry ‘will or will not’ be compared against the received message ID respectively.

7.6 IDE Masking

When the **GAFLIDEM** bit is 1'b0 in an AFL entry, then the IDE bit configured in the AFL Entry is not considered for ID matching. In this case the decision of ID[10:0] or ID[28:0] matching is based on the received IDE bit.

Consider the following Example:

The ID & Mask fields of an AFL Entry "x" is configured as follows:

CFDGAFID [x] = C0553A20h → IDE = 1, RTR = 1, LLB = 0, ID[10:0] = 220h / ID[28:0] = 0553A20h

CFDGAFLMr = 0000FFFFh → IDEM = 0, RTRM = 0, IDM[10:0] = 7FFh / IDM[28:0] = 0000FFFFh

The result of comparison of 4 different received Ids with AFL Entry X is described below:

If a frame with IDE = 1'b0 & ID = 220 is received, then this is considered as a match.

If a frame with IDE = 1'b0 & ID = 320 is received, then this is not a match.

If a frame with IDE = 1'b1 & ID = 1FFF3A20 is received, then this is considered as a match.

If a frame with IDE = 1'b1 & ID = 08803220 is received, then this is not a match.

7.7 updating AFL entry during communication

User can update the AFL entry without disabling all CAN communications.

User chooses the entry number which is due to be updated.

Set AFL entry number and channel number and ignore enable bit.

This entry number is ignored from AFL matching while user update the entry.

Figure below shows the AFL entry update flow.

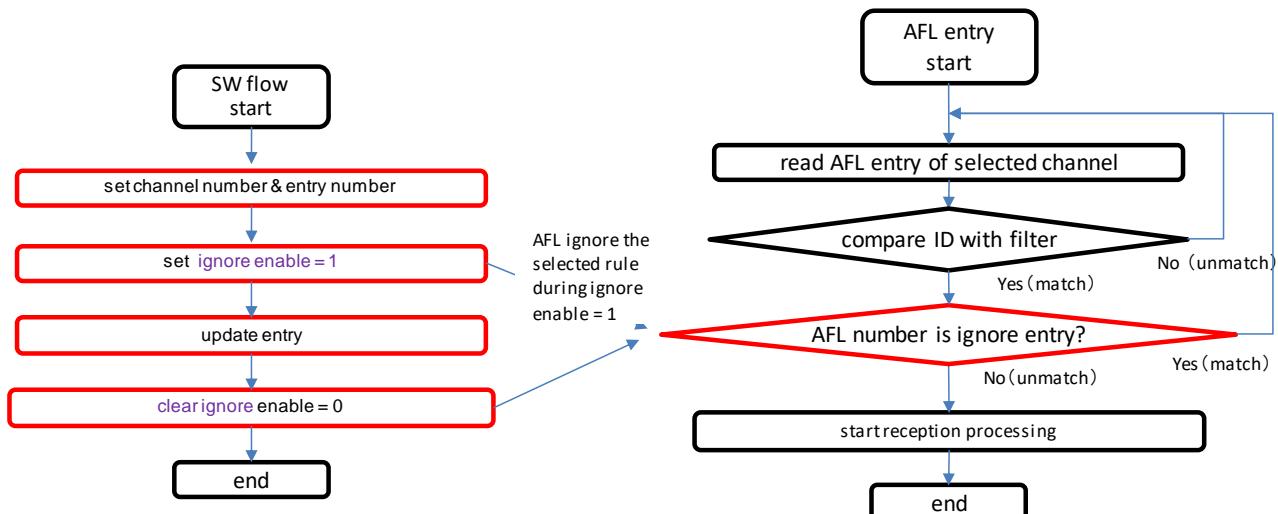


Figure 7.5 AFL entry update flow

The method of update of an AFL entry is shown below.

- (1) Set entry number and a channel number to **CFDGAFLIGNENT** register
- (2) Set the value 16'hC401 (key code & enable bit) to **CFDGAFLIGNCTR** register.
- (3) Set entry page to **CFDGAFLECTR** register. This page includes the selected entry. **CFDGAFLECTR.AFLDAE** is set to 1'b1.
- (4) Set the new rule to **CFDGAFLIDr**, **CFDGAFLMr**, **CFDGAFLP0r**, **CFDGAFLP1r** registers.
- (5) **CFDGAFLECTR.AFLDAE** is cleared to 1'b0.
- (6) Set the value 16'hC400 (key code & clear enable bit) to **CFDGAFLIGNCTR** register.

(*) This entry number becomes ignoring for RXSCAN during the periods from (2) to (5).

For example: case1

When total entry is 6 per channel, delete one of the entry.

Current entry is follows. Delete the entry3 of channel 1.

		entry number of page0	
for channel 0 (total entry=6)	entry0	0	ID=11'h050
	entry1	1	ID=11'h051
	entry2	2	ID=11'h052
	entry3	3	ID=11'h053
	entry4	4	ID=11'h054
	entry5	5	ID=11'h055
for channel 1 (total entry=6)	entry0	6	ID=11'h150
	entry1	7	ID=11'h151
	entry2	8	ID=11'h152
	entry3	9	ID=11'h153
	entry4	10	ID=11'h154
	entry5	11	ID=11'h155
for channel 2 (total entry=4)	entry0	12	ID=11'h250
	entry1	13	ID=11'h251
	entry2	14	ID=11'h252
	entry3	15	ID=11'h253

←delete rule

How to delete the entry

- (1) Set 32'h0001_0003 to **CFDGAFLIGNENT** register.
- (2) Set 32'h0000_c401 to **CFDGAFLIGNCTR** register
- (3) Set 32'h0000_0100 to **CFDGAFLECTR** register
- (4) Set same rule as previous rule by accessing to **CFDGAFLIDr**, **CFDGAFLMr**, **CFDGAFLP0r**, **CFDGAFLP1r** (r=9, this is entry3 of channel 1)
- (5) Set 32'h0000_0000 to **CFDGAFLECTR** register
- (6) Set 32'h0000_c400 to **CFDGAFLIGNCTR** register

Finish to delete the entry3 of channel 1. Current entry is follows

		entry number of page0	
for channel 0 (total entry=6)	entry0	0	ID=11'h050
	entry1	1	ID=11'h051
	entry2	2	ID=11'h052
	entry3	3	ID=11'h053
	entry4	4	ID=11'h054
	entry5	5	ID=11'h055
for channel 1 (total entry=5)	entry0	6	ID=11'h150
	entry1	7	ID=11'h151
	entry2	8	ID=11'h152
	entry3	9	ID=11'h152
	entry4	10	ID=11'h154
	entry5	11	ID=11'h155
for channel 2 (total entry=4)	entry0	12	ID=11'h250
	entry1	13	ID=11'h251
	entry2	14	ID=11'h252
	entry3	15	ID=11'h253

←set rule same as previous rule

For example: case2 When total entry is 6 per channel, add one of the entry.
 Current entry is follows. New entry is added to entry3 of the channel 1.

entry number of page0		
for channel 0 (total entry=6)	entry0	0 ID=11'h050
	entry1	1 ID=11'h051
	entry2	2 ID=11'h052
	entry3	3 ID=11'h053
	entry4	4 ID=11'h054
	entry5	5 ID=11'h055
for channel 1 (total entry=5) entry2=entry3	entry0	6 ID=11'h150
	entry1	7 ID=11'h151
	entry2	8 ID=11'h152
	entry3	9 ID=11'h152
	entry4	10 ID=11'h154
	entry5	11 ID=11'h155
for channel 2 (total entry=4)	entry0	12 ID=11'h250
	entry1	13 ID=11'h251
	entry2	14 ID=11'h252
	entry3	15 ID=11'h253

←add new rule in this position

How to add entry

- (1) Set 32'h0001_0003 to **CFDGAFLIGNENT** register.
- (2) Set 32'h0000_c401 to **CFDGAFLIGNCTR** register
- (3) Set 32'h0000_0100 to **CFDGAFLECTR** register
- (4) Set new rule by accessing to **CFDGAFLIDr**, **CFDGAFLMr**, **CFDGAFLP0r**, **CFDGAFLP1r**
 $(r=9, \text{this is entry3 of channel 1})$
- (5) Set 32'h0000_0000 to **CFDGAFLECTR** register
- (6) Set 32'h0000_c400 to **CFDGAFLIGNCTR** register

Finish to add entry. Current entry is follows

entry number of page0		
for channel 0 (total entry=6)	entry0	0 ID=11'h050
	entry1	1 ID=11'h051
	entry2	2 ID=11'h052
	entry3	3 ID=11'h053
	entry4	4 ID=11'h054
	entry5	5 ID=11'h055
for channel 1 (total entry=6)	entry0	6 ID=11'h150
	entry1	7 ID=11'h151
	entry2	8 ID=11'h152
	entry3	9 ID=11'h156
	entry4	10 ID=11'h154
	entry5	11 ID=11'h155
for channel 2 (total entry=4)	entry0	12 ID=11'h250
	entry1	13 ID=11'h251
	entry2	14 ID=11'h252
	entry3	15 ID=11'h253

←add new rule



AFL filter can be used to the range set **CFDGAFLCFGw**. An addition and deletion of an entry are possible in it. Therefore, it is necessary to set the maximum number to be used to **CFDGAFLCFGw**.

8 FIFO Buffers & Normal MB Configuration

This Section describes the process for configuration of the number of RX Message Buffers, the FIFO Buffers and the Flat TX Message Buffers in the RS-CAN-FD module. The Message Buffers are mapped as shown in Figure 8.1 below.

The RX Message Buffers can be accessed via RX Message Buffer Registers.

The RX FIFO buffers and the common FIFO buffers configured in RX mode or TX mode or GW mode can only be accessed via the FIFO Access Registers.

If the common FIFO is configured in TX mode, then users can only write data into the FIFO via FIFO Access registers.

If the common FIFO is configured in GW mode or RX mode then users can only read data from the FIFO Access Registers.

The TX Message Buffers can be accessed via the TX Message Buffer Registers.

If users read unused Message Buffer locations, the MB locations are read as unknown values.

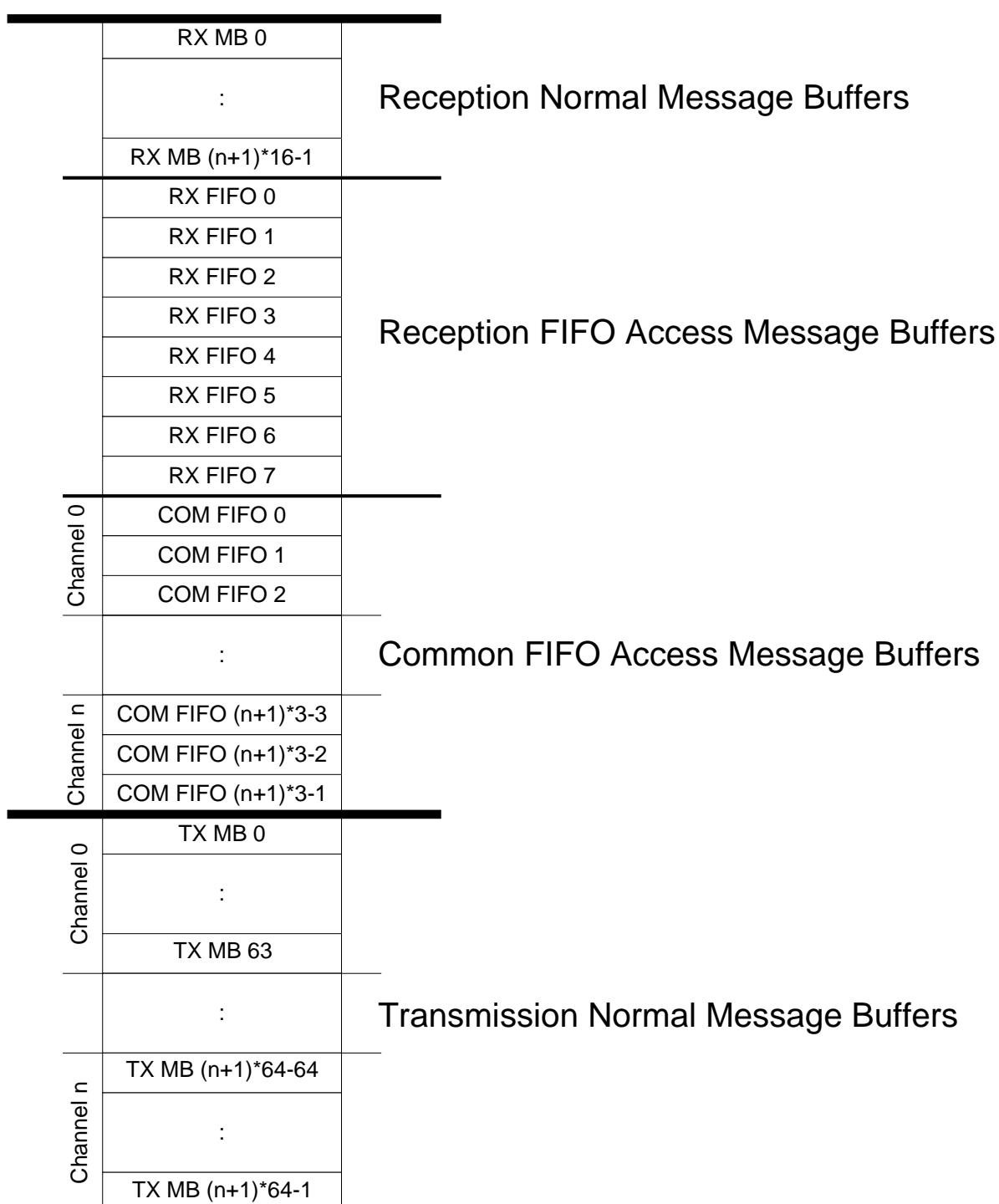


Figure 8.1: Message Buffer Configuration

8.1 Normal RX Message Buffers

In RS-CAN-FD module, the frames received by various channels can be stored in Normal RX Message Buffers based on the configuration of the AFL entries.

Additionally, the number of Normal RX Message Buffers required in the system can be chosen up to a fixed maximum limit.

8.1.1 Normal RX Message Buffer configuration

In RS-CAN-FD module, the number of normal RX Message Buffers can be configured by writing to the RX Message Buffer Number Register.

The limiting values for the configuration of number of Message Buffers are:

Minimum Value = 8'h0 (no Normal RX MB)

Maximum Value = (16 * No. of CAN channels)

$$= 8'h80 \text{ (128 Flat RX MBs for 8 channels)}$$

Users should not use values outside these limits.

The AFL entries for routing the received messages to normal RX Message Buffers should be configured to match the requirements of the system.

The AFL entries should also be configured properly. An AFL entry for normal RX Message Buffers should not exceed the number of Message Buffers configured in the RX Message Buffer Number Register.

Note: There is no internal check procedure provided in RS-CAN-FD module against wrong configuration of the AFL.

The data field size of the RX Message Buffer can be configured via the **CFDRMNB.RMPLS**. The default size is 8 Bytes. The max data payload size is 64 Bytes.

In case the receiving frame exceeds the data field size then the acceptance depends on the configuration of the **CFDGCFG.CMPOC** (message rejecting or data payload cut).

8.2 FIFO Buffers

The RS-CAN-FD module provides a fixed number of FIFO Buffers to support storage of frames for reception, transmission and gateway functions for various CAN channels.

Number of reception-only FIFO Buffers is fixed to 8.

However, 3 common FIFO Buffers per channel can be configured for storing messages for transmission or reception or gateway function.

These FIFO Buffers can be enabled or disabled and the size, Interrupt structure and Message Lost mechanism and Message overwrite mechanism of the FIFO Buffers, as well as the location of the TX FIFO or GW FIFO can be configured to match the system requirements.

In case the receiving frame exceeds the data field size then the acceptance depends on the configuration of the **CFDGCFG.CMPOC** (message rejecting or data payload cut).

8.2.1 FIFO Buffers configuration

In RS-CAN-FD module, the FIFO Buffers can be configured to match the system requirements.

The total number of FIFO Buffers = 8 RX FIFO Buffers + 24 common FIFO Buffers =32 FIFO Buffers for 8 channels and Message overwrite mechanism

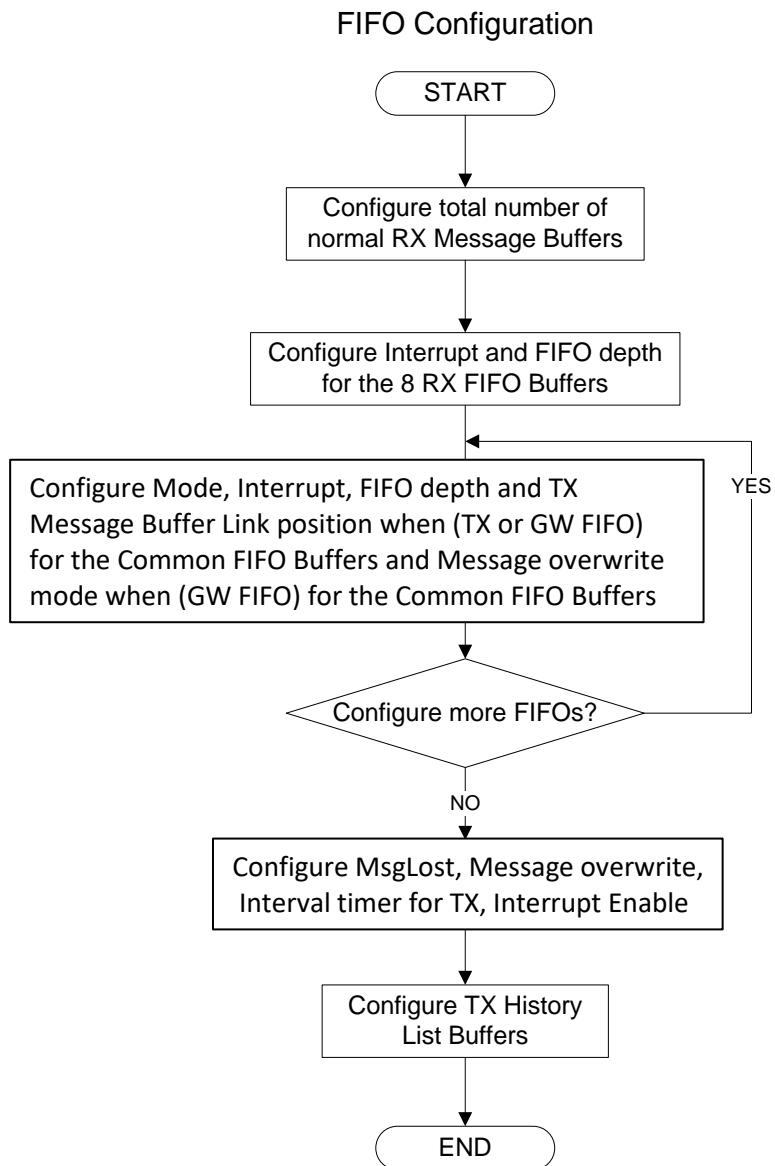


Figure 8.2: FIFO Buffer configuration flow in RS-CAN-FD module

As shown in Figure 8.2, the various FIFO Buffers can be configured by writing to the RX FIFO Configuration / Control Registers and the Common FIFO Configuration / Control Registers.

For the 8 RX FIFO Buffers, the Interrupts, the FIFO depth and the FIFO payload data size can be configured.

For the common FIFO Buffers, the mode, Interrupts FIFO depth, the FIFO payload data size and the FIFO TX link position can be configured.

8.2.1.1 FIFO Mode Configuration of Common FIFO Buffers

The mode of the common FIFO Buffers can be configured by writing to the **CFDCFCCd.CFM[1:0]** bits in the Common FIFO Configuration / Control Registers. The possible modes of configuration for Common FIFO Buffers are:

2'b00 RX mode (default mode after HW Reset)

2'b01 TX mode

2'b10 GW mode.

2'b11 Reserved (Users should not write this value to the Register bits)

Messages can only be read from the RX FIFO Buffers and the common FIFO Buffers configured in RX Mode.

Messages are stored by the CAN module in these FIFO buffers based on the AFL entries.

Messages can be read and written into the common FIFO Buffers configured in TX mode. These messages will be transmitted on the appropriate CAN channel.

Messages can only be read from the common FIFO Buffers configured in GW mode. However, the CPU read access has no impact on the read or write pointers.

The pointers can only be incremented when a new message is stored in the FIFO Buffer and decremented when a message is transmitted on the corresponding CAN channel by the RS-CAN-FD module.

After HW reset, all the common FIFO Buffers are configured in RX mode by default. Users should only enable the FIFO Buffers after configuring the common FIFO Buffers in the required modes.

In GW mode, when a transmit/receive FIFO buffer is trying to receive a new message while the transmit/receive FIFO buffer is already full of data, the oldest data of the buffer will be overwritten with the message received or the message will be discarded. The behavior is determined by setting **CFDCFCCEd.CFMOWM** bit.

When **CFDCFCCEd.CFMOWM=0**:

When writing of data is required due to reception of a new message while a transmit/receive FIFO buffer is full of data, the received message will be discarded. And **CFDCFSTSd.CFMLT** bit is set to 1.

When **CFDCFCCEd.CFMOWM=1**:

When writing of data is required due to reception of a new message while a transmit/receive FIFO buffer is full of data, the oldest data in the buffer will be overwritten with the received message. The read pointer of the transmit/receive FIFO buffer simultaneously moves to the next oldest message. Then, **CFDCFSTSd.CFMOW** bit is set to 1, which notifies that the oldest message has been overwritten with the received message.

In addition, in a case a CAN bus error or arbitration-lost for the transmitting message occurs in transmit/receive FIFO buffer full, the transmitting message is lost and re-transmission for the message is not performed. Then the read point moves to the next message automatically.

Users should not write change for this bit when the **CFDCFCCd.CFE** bit is 1'b1.

8.2.1.2 FIFO TX-Message Buffer Link configuration

When the common FIFO is configured as TX or GW FIFO, then the FIFO Buffer must be linked to a normal TX Message Buffer to participate in the transmission scan of a CAN channel.

The link to a normal TX Message Buffer should be unique i.e. the same TX Message Buffer cannot be shared between 2 or more common FIFO Buffers.

Users should not write data into a TX Message Buffer that is linked to a Common FIFO buffer.

Also, the TX Message Buffer linked to a common FIFO buffer should not be a part of the TX Queue.

The TX Message Buffer link of each common FIFO Buffer can be configured by writing to the **CFDCFCCd.CFTML[4:0]** bits in the Common FIFO Configuration / Control Registers. Available options for TX Message Buffer link configuration are:

5'b00000: TX Message Buffer 32

5'b00001: TX Message Buffer 33

: :
:

5'b11110: TX Message Buffer 62

5'b11111: TX Message Buffer 63

8.2.1.3 FIFO Depth Configuration

The depth of each FIFO Buffer can be configured by writing to the **CFDRFCCa.RFDC[2:0]** bits and **CFDCFCCd.CFDC[2:0]** bits in the RX FIFO Configuration / Control Registers and the Common FIFO Configuration / Control Registers. The 8 available options for depth configuration are:

3'b000: 0 Messages (FIFO Buffer cannot be enabled)

3'b001: 4 Messages

3'b010: 8 Messages

3'b011: 16 Messages
3'b100: 32 Messages
3'b101: 48 Messages
3'b110: 64 Messages
3'b111: 128 Messages

The RAM allocation for RX Message Buffers along with FIFO Buffers is limited to $(n+1) \times 256$ messages. Configuration of the RX Message Buffers, along with FIFO Buffers, that exceeds this maximum limit should not be done.

RS-CAN-FD module logic will not check the validity of the configuration.

Note: If the FIFO depth of a common FIFO is 4 messages or more (**CFDCFCCd.CFDC[2:0]** > 3'b000), then the common FIFO TX Message Buffer link is valid when the FIFO is disabled as well as enabled.

If FIFO depth is 0 messages, then the common FIFO TX Message Buffer link is not valid when the FIFO is disabled as well as enabled.

8.2.1.4 FIFO Payload Size Configuration

The data size of each FIFO Buffer can be configured by writing to the **CFDRFCCa.RFPLS[2:0]** bits and **CFDCFCCd.CFPLS[2:0]** bits in the RX FIFO Configuration / Control Registers and the Common FIFO Configuration / Control Registers. The 8 available options for depth configuration are:

- 3'b000: 8 Bytes
- 3'b001: 12 Bytes
- 3'b010: 16 Bytes
- 3'b011: 20 Bytes
- 3'b100: 24 Bytes
- 3'b101: 32 Bytes
- 3'b110: 48 Bytes
- 3'b111: 64 Bytes

The RAM allocation for RX Message Buffers along with FIFO Buffers is limited to $(n+1) * 256$ messages with 64 Data Bytes. Configuration of the RX Message Buffers, along with FIFO Buffers, that exceeds this maximum limit should not be done.

RS-CAN-FD module logic will not check the validity of the configuration.

8.2.1.5 FIFO Interrupt Configuration

The Interrupt generation conditions for the FIFO Buffers can be configured by writing to the **CFDRFCCa.RFIGM** and **CFDCFCCd.CFIM** bit in the RX FIFO Configuration / Control Registers and the Common FIFO Configuration / Control Registers. The 2 available options are:

- 0:
 - RX FIFO Mode: Interrupt generated when Common FIFO counter reaches **CFDRFCCa.RFIGCV / CFDCFCCd.CFIGCV** value from below values;
 - TX FIFO Mode: Interrupt generated when Common FIFO transmits last message successfully;
 - GW FIFO Mode:
 - Frame RX: Interrupt generated when message counter increments and reaches the Interrupt Threshold value;
 - Frame TX: Interrupt generated when last message is transmitted successfully from FIFO;
- 1:
 - RX FIFO Mode: Interrupt generated at the end of storage of every received message;

TX FIFO Mode: Interrupt generated for every successfully transmitted message;

GW FIFO Mode:

Frame RX: Interrupt generated when message is stored in the FIFO;

Frame TX: Interrupt generated when message is successfully transmitted from the FIFO;

If the interrupt mode bit is 1'b0 for a RX FIFO, then interrupt is generated based on the configuration of the **CFDRFCCa.RFIGCV[2:0]** bits.

Similarly, if the interrupt mode bit is 1'b0 for a Common FIFO configured in RX mode, then interrupt is generated based on the configuration of **CFDCFCCd.CFIGCV[2:0]** bits.

The 8 available options for configuring the FIFO counter value for generation of an interrupt are:

- 3'b000: Interrupt generated when FIFO is 1/8th Full
- 3'b001: Interrupt generated when FIFO is 1/4th Full
- 3'b010: Interrupt generated when FIFO is 3/8th Full
- 3'b011: Interrupt generated when FIFO is 1/2 Full
- 3'b100: Interrupt generated when FIFO is 5/8th Full
- 3'b101: Interrupt generated when FIFO is 3/4th Full
- 3'b110: Interrupt generated when FIFO is 7/8th Full
- 3'b111: Interrupt generated when FIFO is Full

In this case, an interrupt is generated when the Message Count matches the configured value.

However, there are some limitations on the configuration of the **CFDRFCCa.RFIGCV[2:0]** and **CFDCFCCd.CFIGCV[2:0]** bits depending upon the FDC[2:0] bits (FIFO Depth Configuration) see Table 8.1.

RFDC[2:0] (CFDC[2:0])	RFIGCV[2:0] (CFIGCV[2:0])							
111	110	101	100	011	010	001	000	
000	don't care (FIFO can not be enabled)							
001	allowed	not allowed	allowed	not allowed	allowed	not allowed	allowed	not allowed
010	allowed							
011	allowed							
100	allowed							
101	allowed							
110	allowed							
111	allowed							

Table 8.1 FIFO Interrupt generation counter vs. FIFO depth configuration

Common FIFO can set an interrupt output at the time of the completion of transmitting of one frame, or the completion of reception. Moreover, Common FIFO and RX FIFO can set an interrupt output, when stored to the set-up number (**CFDC/RFDC**) of FIFO stages

8.2.2 FIFO Buffers control

The FIFO Interrupt should be enabled by setting the **CFDRFCCa.RFIE** or **CFDRFCCa.RFFIE** bits in the RX FIFO Configuration / Control Registers and **CFDCFCCd.CFRXIE** or **CFDCFCCd.CFTXIE** or **CFDCFCCEd.CFFIE** or **CFDCFCCEd.CFOFRXIE** or **CFDCFCCEd.CFOFTXIE** bits in the Common FIFO Configuration / Control Registers.

After configuration is complete, each FIFO can be enabled by setting the **CFDRFCCa.RFE** and **CFDCFCCd.CFE** bit in the RX FIFO Configuration / Control Registers and the Common FIFO Configuration / Control Registers to allow transmission and reception of messages.

In the case of **CFDCFCCEd.CFBME=1**, it becomes FIFO buffering mode, send data is stored in Common FIFO, and transmission is stopped. Transmission will be started if it is set as **CFDCFCCEd.CFBME=0**.

Users should not write 1 from 0 for this bit when the **CFDCFCCd.CFE** bit is 1'b1.

9 Interrupts and DMA

9.1 Interrupts

The RS-CAN-FD module generates several Interrupts.

The interrupt output, which is connected to the Interrupt Controller Unit, can be controlled by the corresponding interrupt enable bit.

The status flag will be set independent from this enable bit.

The channel Transmission Interrupt has an additional Status flag register; these Status bits will only be set when the corresponding interrupt enables are set.

This register supports the identification of the interrupt source for the channel transmission, as this interrupt is driven by several trigger sources.

The Interrupts in the RS-CAN-FD module can be classified into 2 groups, Global Interrupts and Channel Interrupts:

Global Interrupts:

The RS-CAN-FD module can generate 2 Global Interrupts:

1.1 Global Interrupt for successful reception into the 8 RX FIFO buffers

2.1 Global Error Interrupt

Channel Interrupts:

Each channel of the RS-CAN-FD module can generate 3 Channel Interrupts:

1. Channel Transmission

Transmission completion from channel

Transmission abort from channel

Transmission from TX Queue for a channel

Channel THL Interrupt

Successful Transmission from a Common FIFO in TX or GW mode for a channel

2. Channel Error Interrupt

3. Successful Reception in a Common FIFO in RX or GW mode for a channel or Successful Reception in a TXQ

The interrupts are cleared when the corresponding flag bits are cleared or Interrupt enable bits are cleared.

If the set from the RS-CAN-FD module occurs simultaneously with the clear by the write access, then each flag bit is set. The set condition of the flag bits is prioritized.

Interrupt output in FFI mode

The FFI mode can output the existing interrupt to a virtual machine unit.

Moreover, an output is possible to virtual machine error interrupt in message lost of Global error interrupt, or message overwrite interrupt.

The interruption output to virtual machine in FFI mode is shown below.

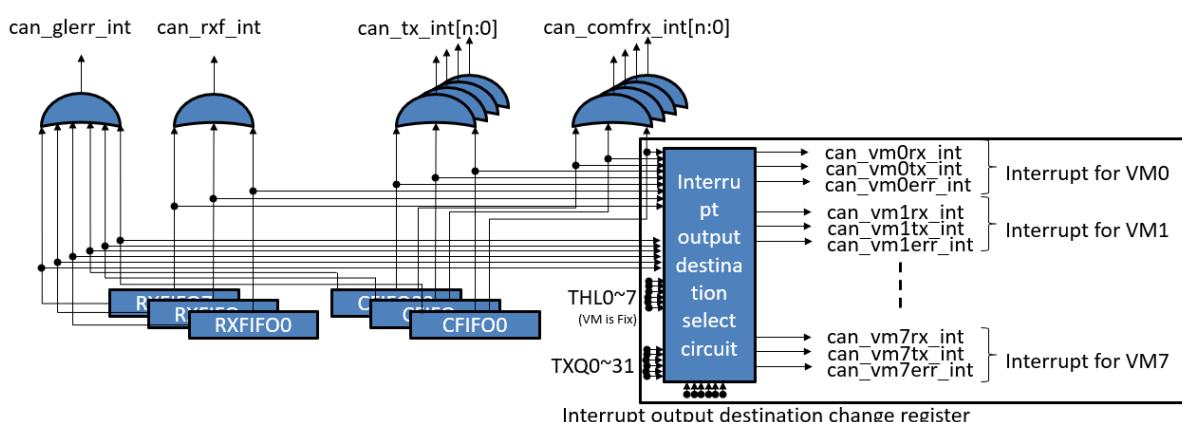


Table 9.1 below gives an overview of interrupt sources for the different interrupt outputs.

The Interrupt outputs are Active High

	interrupt terminal	interrupt	when not FFI mode	when FFI mode	interrupt source	interrupt clearing
Global Interrupts	can_rx_int	Successful reception into at least one RX FIFO	Valid	Invalid	interrupt flag of corresponding RX FIFO for which interrupt is enabled	clear the interrupt flags of corresponding RX FIFO buffer for which interrupt is enabled
		FIFO full into at least one RX FIFO			FIFO full interrupt flag of corresponding RX FIFO for which interrupt is enabled	clear the FIFO full interrupt flags of corresponding RX FIFO buffer for which interrupt is enabled
	can_glerr_int	Global Error	Valid	Valid	any of the following: •DLC Error Flag •Message Lost Status bit •Message OverWrite Status bit •TXQ Message Lost Status bit •TXQ Message OverWrite Status bit •TX History Entry Lost Status bit	clear all of: •DLC Error Flag •Message Lost Flags in all of the FIFO Status Registers •Message Overwrite Flags in all of the Common FIFO Status Registers •Message Lost Flags in all of the TXQ Status Registers •Message Overwrite Flags in all of the TXQ Status Registers •TX History List Entry Lost Flag
					•CAN-FD Message Payload overflow flag	•CAN-FD Message Payload overflow flag
	Channel Transmission Interrupts	Channel n successful transmission	Valid	Valid	any channel related TX MB Successful flag when interrupt is enabled	clear all channel related TX MB Result status bits for which the interrupt is enabled
		Channel n Abort			Note: These interrupts are only set for TX Message Buffers that do not belong to an enabled TX Queue and are not pointing to a common FIFO. Separate interrupts are provided for common FIFO buffers & TX Queue	
		Channel n transmission from TX Queue			any channel related TX MB Abort flag when interrupt is enabled	clear all channel related TX MB Result status bits for which the interrupt is enabled globally.
		Channel n THL Interrupt				
		Channel n COM FIFO TX Interrupt				
		Channel n COM FIFO One Frame TX Interrupt				
		Channel n TXQ One Frame TX Interrupt				
Channel COM RX FIFO Interrupt	can_comfrx_int[n:0]	Channel n COM FIFO RX Interrupt	Valid	Invalid	Interrupt Flag for Common FIFOs in RX or GW mode belonging to the related channel	clear the interrupt flags of Common FIFOs in RX or GW mode belonging to the related channel
		Channel n COM FIFO One Frame RX Interrupt			One Frame Reception Interrupt Flag for Common FIFOs in RX or GW mode belonging to the related channel	clear the One Frame Reception interrupt flags of Common FIFOs in RX or GW mode belonging to the related channel
		Channel n COM FIFO Full Interrupt			FIFO Full Interrupt Flag for Common FIFOs in RX or GW mode belonging to the related channel	clear the FIFO Full interrupt flags of Common FIFOs in RX or GW mode belonging to the related channel
		Channel n TXQ One Frame RX Interrupt			One Frame Reception Interrupt Flag for TXQs in GW mode belonging to the related channel	clear the One Frame Reception interrupt flags of TXQs in GW mode belonging to the related channel
		Channel n TXQ Full Interrupt			TXQ Full Interrupt Flag for TXQs in GW mode belonging to the related channel	clear the TXQ Full interrupt flags of TXQs in GW mode belonging to the related channel
Channel Error Interrupt	can_cherr_int[n:0]	Channel n Error	Valid	Valid	any channel related error flag in the Channel Error Flag Register for which interrupt is enabled in the Channel Error Interrupt Enable Register	clear all channel related error flags in the Channel Error Flag Register for which interrupt is enabled in the Channel Error Interrupt Enable Register
Virtual Machine TX interrupt	can_vmtx_int[n:0]	Virtual machine n transmission from TX Queue	Invalid	Valid	related virtual machine TX Queue Interrupt Flag	clear related virtual machine TX Queue Interrupt Flag
		Virtual machine n THL Interrupt			related virtual machine THL Interrupt status flag	clear related virtual machine THL Interrupt status flag
		Virtual machine n COM FIFO TX Interrupt			Interrupt Flag for Common FIFOs in RX or GW mode belonging to the related virtual machine	clear the interrupt flags of Common FIFOs in RX or GW mode belonging to the related virtual machine
		Virtual machine n COM FIFO One Frame TX Interrupt			One Frame Transmission Interrupt Flag for Common FIFOs belonging to the related virtual machine	clear the One Frame Transmission interrupt flags of Common FIFOs belonging to the related virtual machine
		Virtual machine n TXQ One Frame TX Interrupt			One Frame Transmission Interrupt Flag for TXQs belonging to the related virtual machine	clear the One Frame Transmission interrupt flags of TXQs belonging to the related virtual machine
Virtual Machine RX interrupt	can_vmrx_int[n:0]	Virtual machine n COM FIFO RX Interrupt	Invalid	Valid	Interrupt Flag for Common FIFOs in RX or GW mode belonging to the related virtual machine	clear the interrupt flags of Common FIFOs in RX or GW mode belonging to the related virtual machine
		Virtual machine n COM FIFO One Frame RX Interrupt			One Frame Reception Interrupt Flag for Common FIFOs belonging to the related virtual machine	clear the One Frame Reception interrupt flags of Common FIFOs belonging to the related virtual machine
		Virtual machine n COM FIFO Full Interrupt			FIFO Full Interrupt Flag for Common FIFOs in RX or GW mode belonging to the related virtual machine	clear the FIFO Full interrupt flags of Common FIFOs in RX or GW mode belonging to the related virtual machine
		Virtual machine n TXQ One Frame routing Interrupt			One Frame Routing Interrupt Flag for TXQs in GW mode belonging to the related virtual machine	clear the One Frame Routing interrupt flags of TXQs in GW mode belonging to the related virtual machine
		Virtual machine n TXQ Full Interrupt			TXQ Full Interrupt Flag for TXQs in GW mode belonging to the related virtual machine	clear the TXQ Full interrupt flags of TXQs in GW mode belonging to the related virtual machine
		Virtual machine n RX FIFO Interrupt			Interrupt Flag for RX FIFOs belonging to the related virtual machine	clear the interrupt flags of RX FIFOs belonging to the related virtual machine
Virtual Machine Error interrupt	can_vmerr_int[n:0]	Virtual machine n RX FIFO Full Interrupt			FIFO Full Interrupt Flag for RX FIFOs belonging to the related virtual machine	clear the FIFO Full interrupt flags of RX FIFOs belonging to the related virtual machine

*1:Selection is possible to either Global error interrupt or virtual machine error interrupt by CFDGVMIEIS.

Table 9.1 Interrupt source overview

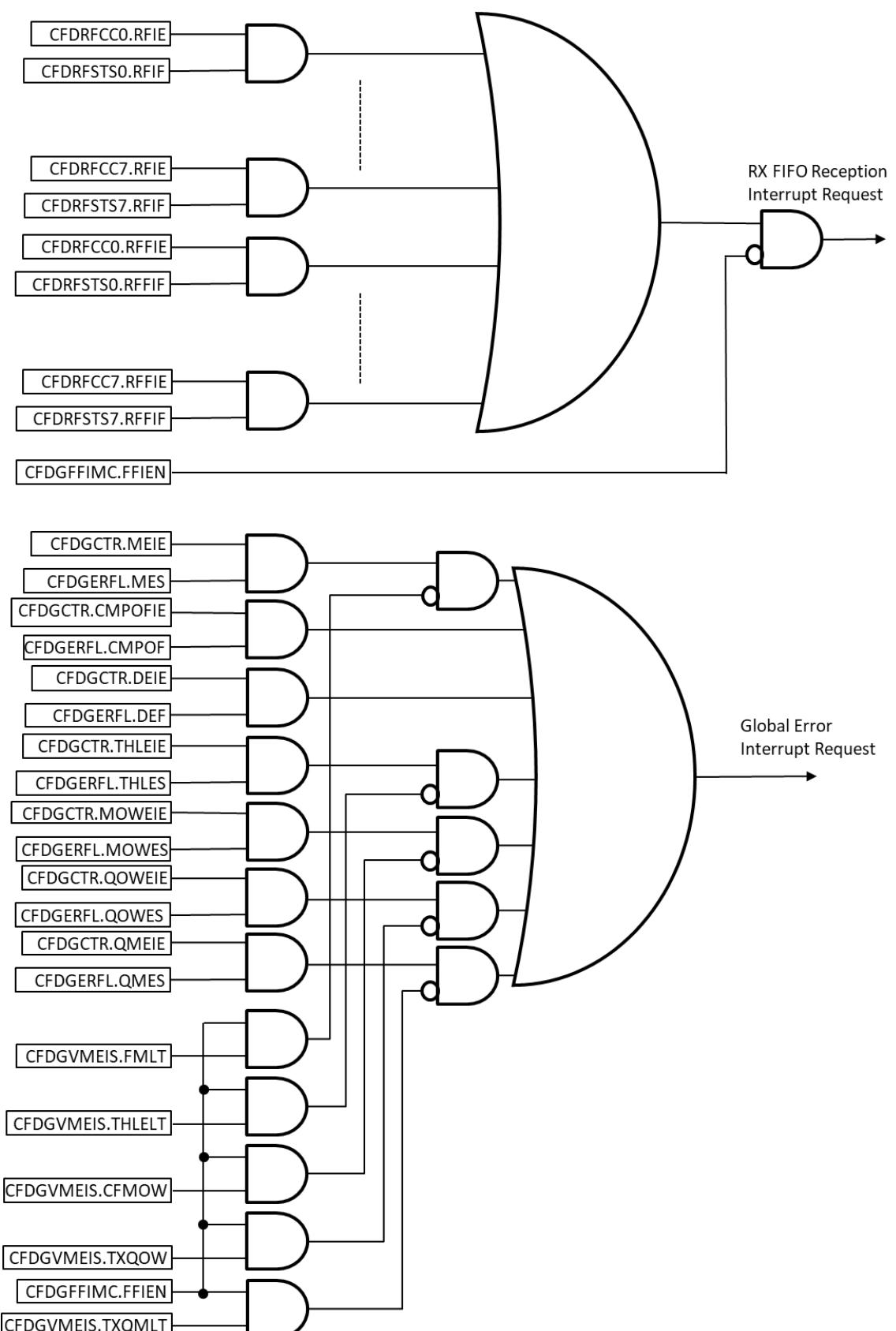
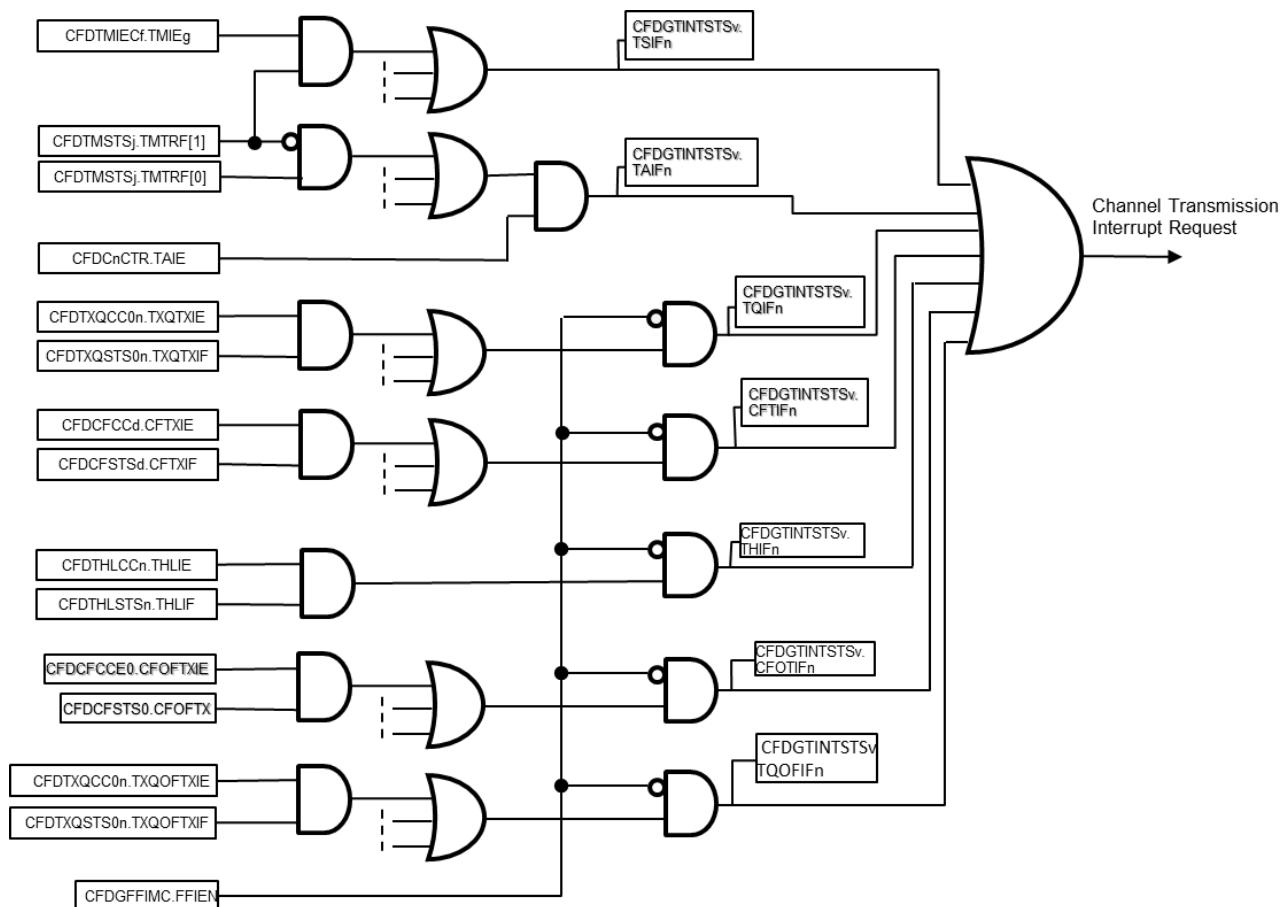


Figure 9.1 Global Interrupt Block Diagram



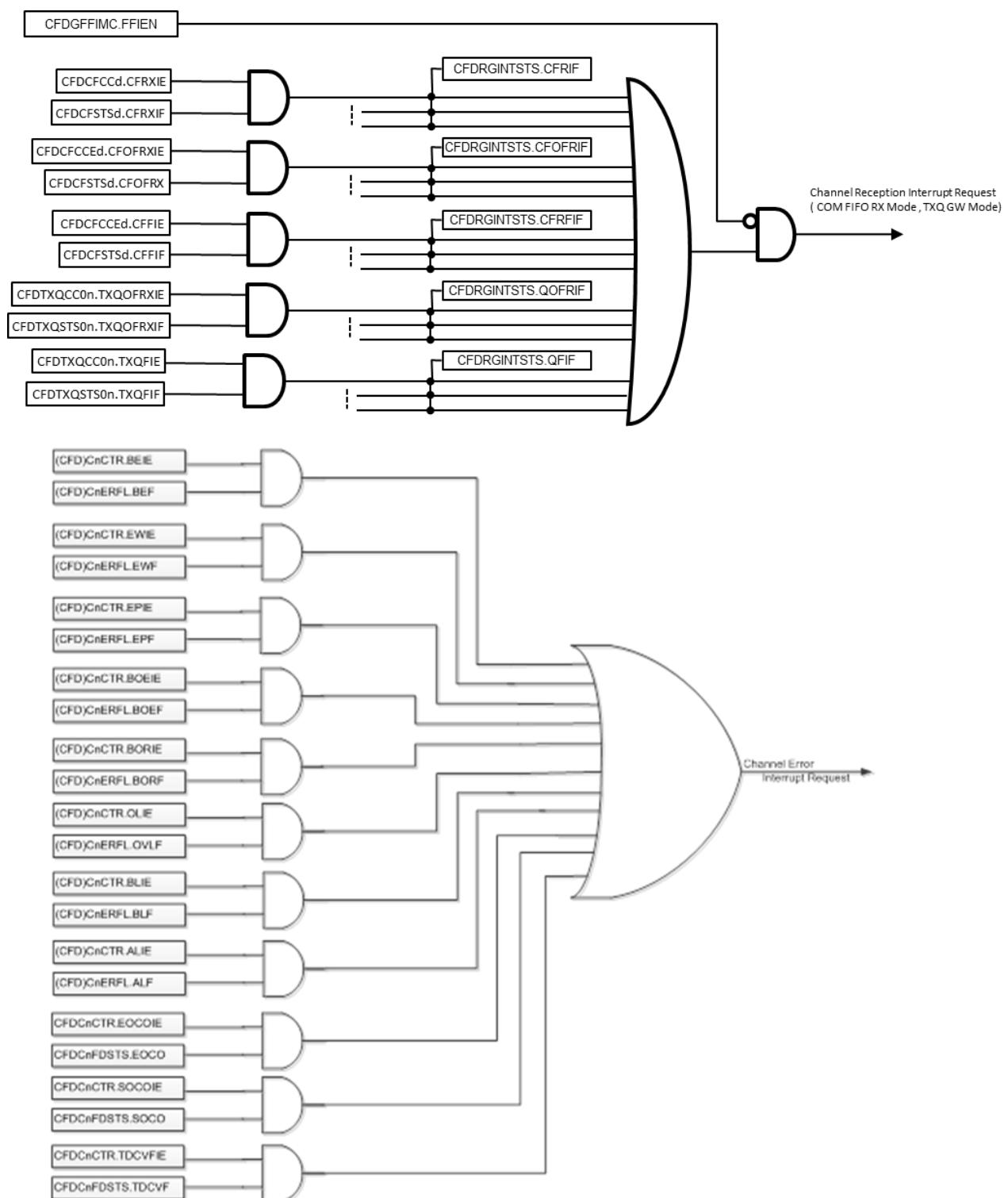
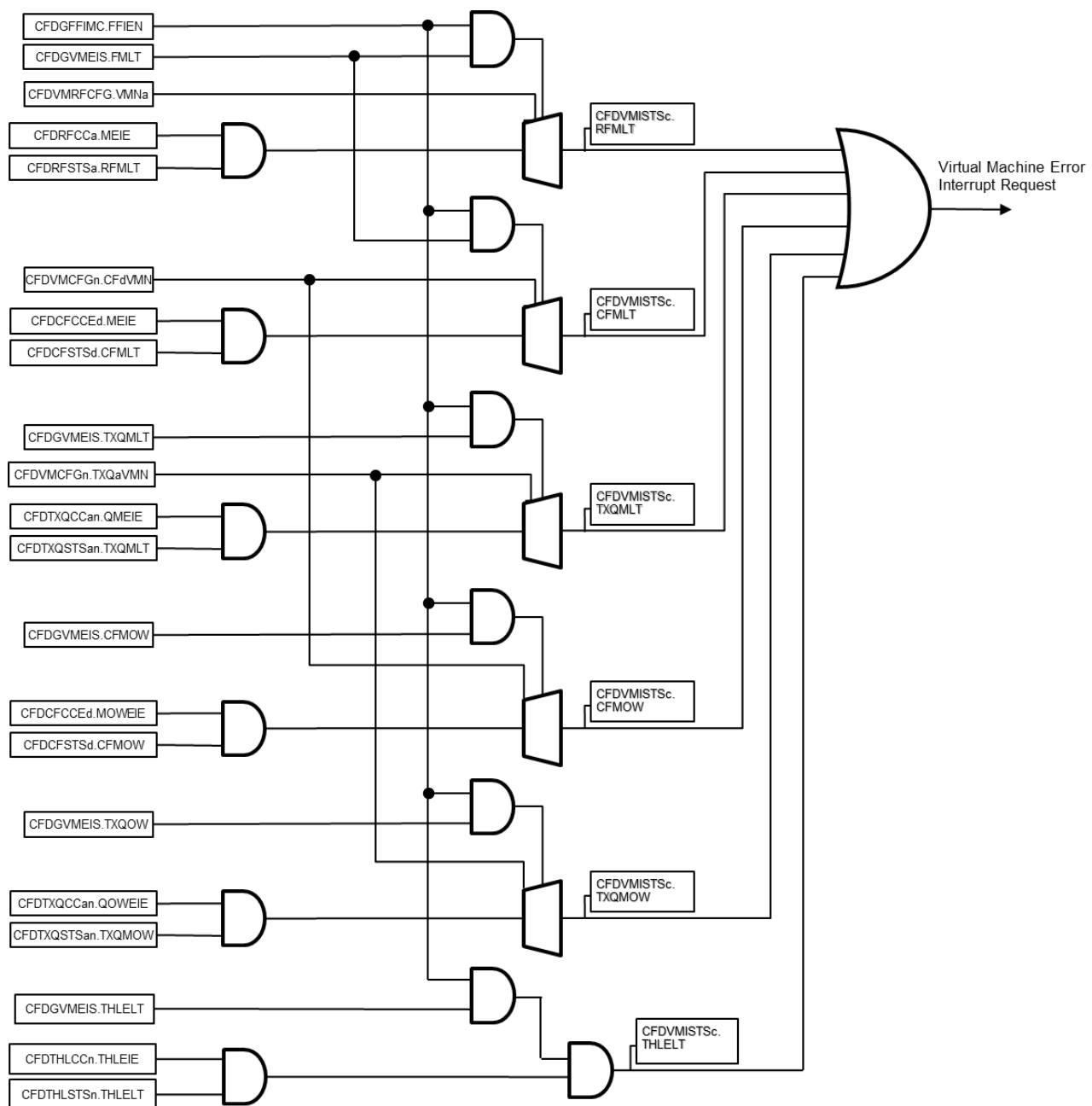


Figure 9.2 Channel Interrupt Block Diagram



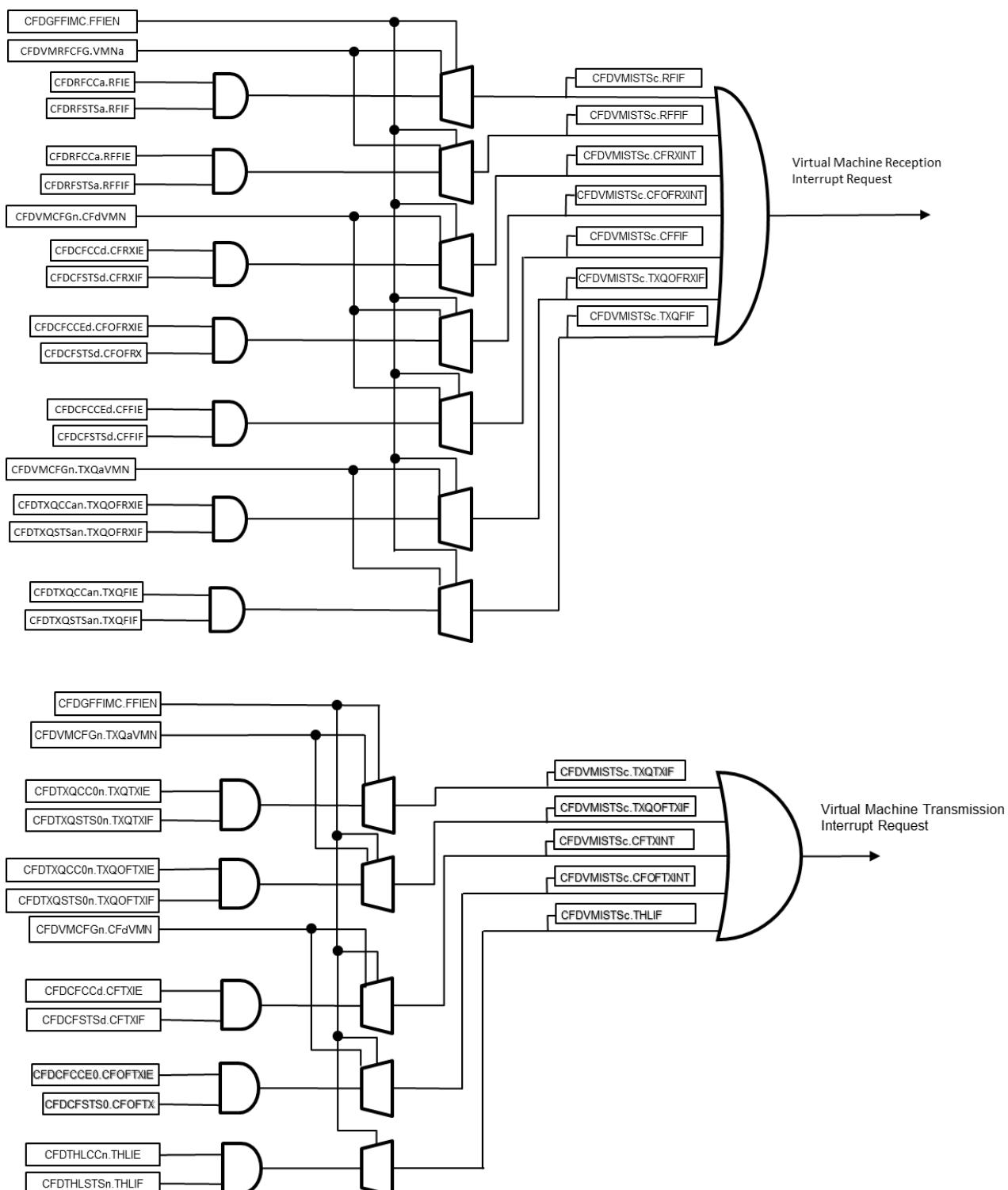


Figure 9.3 FFI mode Interrupt Block Diagram

9.2 DMA Transfer

The RS-CAN-FD module has some message buffer which can be associated with a DMA channels:

Reception DMA

- 8 RX FIFO Message Buffers

- 8 Common FIFO Message Buffers

Transmission DMA

- 16 TXQ Message Buffers (TXQ0, TXQ3)
- 8 Common FIFO Message Buffers

The figure below illustrates the potential DMA channels.

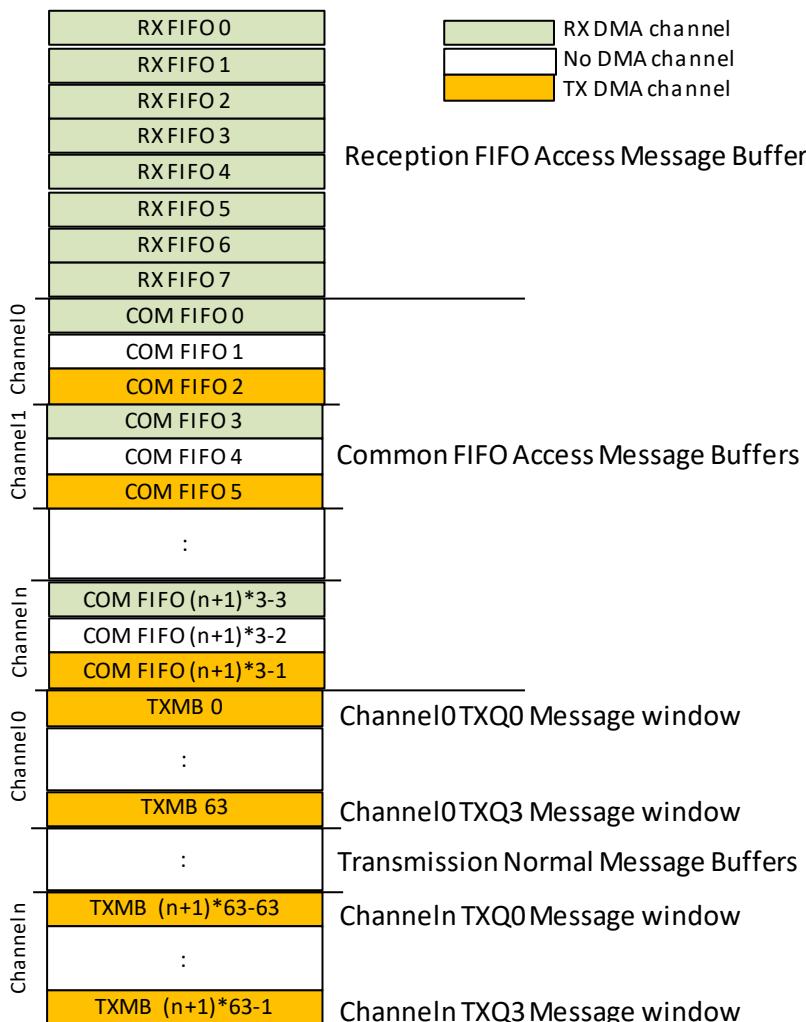


Figure 9.4 Message Buffer connectable to a DMA channel

A DMA channel transfer request will be generated for each FIFO entry to the DMAC when the related **CFDCDTCT.RFDMAE** or **CFDCDTCT.CFDMAE** is set to 1'b1 and the belonging FIFO is not empty.

Reception FIFO Interrupt should be disabled for this particular FIFO (**CFDRFCCa.RFIE** or **CFDCFCCd.CFRXIE**)

Users should use the regular start address for the DMA access window address. Users should add 8000h to the regular start address for the debugger access window. Please refer to the figure below.

b = Message Buffer Component Index	MBCP	Register	p	Regular Start Address n = [0...no_of_channels-1]	Debugger Start Address n = [0...no_of_channels-1]
[0...no_of_RFMBCPs-1]	RFMBCP b[i]	RFIDE	x	6000h + b*0080h	E000h + b*0080h
		RFPTRE	x	6004h + b*0080h	E004h + b*0080h
		RFFDSTS	x	6008h + b*0080h	E008h + b*0080h
		RFDFpE	[0...15]	600Ch + p*0004h + b*0080h	E00Ch + p*0004h + b*0080h
[0...no_of_CFMBCPs_per_channel-1]	CFMBCP b[i]	CFIDE	x	6400h + b*0080h + n*180h	E400h + b*0080h + n*180h
		CFPTRE	x	6404h + b*0080h + n*180h	E404h + b*0080h + n*180h
		CFFDCSTS	x	6408h + b*0080h + n*180h	E408h + b*0080h + n*180h
		CFDFpE	[0...15]	640Ch + p*0004h + b*0080h + n*180h	E40Ch + p*0004h + b*0080h + n*180h

Figure 9.5 DMA channel access window address

DMA FIFO pointer decrement will be done automatically with reading the last configured data payload Byte (**CFDRFCCa.RFPLS** or **CFDCFCCd.CFPLS**).

Note: The DMA should read the exact length of the configured data payload size (**CFDRFCCa.RFPLS** or **CFDCFCCd.CFPLS**), no more, no less.

The SW Debugger should access outside of the regular SFR address range.

The SW Debugger should access from E000h to EFFFh.

Users should not write to the FIFO and TXQ control registers when DMA is enabled.

The DMA enable of the particular DMA FIFO (**CFDCDTCT.RFDMAE** or **CFDCDTCT.CFDMAE**) can be set at any time, [Figure 9.6 DMA enable flow](#) below is a configuration flow for an initial set-up.

When **CFDCDTTCT.TQ0DMAE** or **CFDCDTTCT.TQ3DMAE** or **CFDCDTTCT.CFDMAE** is set, the messages of the corresponding TXQ or Common FIFO can be handled by DMA controller.

Take the following procedure when the TXQ or the Common FIFO can be handled by DMA controller.

1. CPU checks the TXQ or the Common FIFO is not full.
2. When transmit data can be used, CPU **enables DMA to sets** this data to Common FIFO or TXQ.
When using Common FIFO, transmit data is write in **CFDCFID**, **CFDCFPTR**, **CFDCFFDCSTS** and **CFDTMBCPb[i]** register.
When using TXQ, transmit data is write in **CFDTMID**, **CFDTMPTR**, **CFDTMFDFCTR** and **CFDTMDFp** register.
3. In a case of the Common FIFO, common FIFO pointer is incremented automatically when DMA controller writes the last data payload byte configured by **CFDCFCCd.CFPLS**.

In the case of TXQ, if the data of 64 data payload is written, a TXQ pointer will increase automatically. When payload data is less than 64Byte, dummy data must be written in and 64 data payload size must be done.

Note only 32-bit write-access can be possible on the DMA message handling.

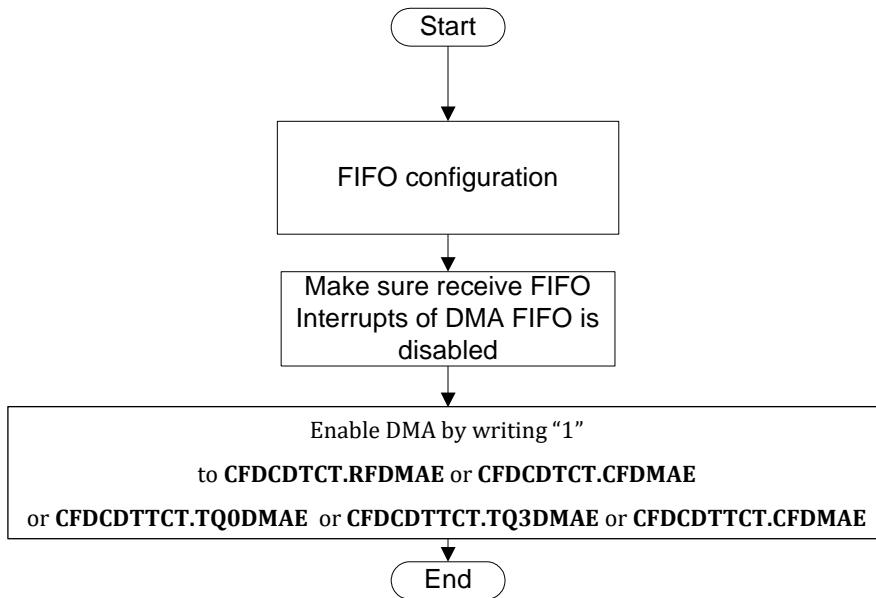


Figure 9.6 DMA enable flow

To disable a DMA transfer requested please disable the particular DMA enable bit (**CFDCDTCT.RFDMAE** or **CFDCDTCT.CFDMAE**). If the disable is made during an ongoing transfer then this must be completed first before further action should be taken. The transfer status can be identified by the **CFDCDTSTS.RFDMASTS** or **CFDCDTSTS.CFDMASTS**. For reference see the flow below. When the DMA is disabled then consider what to do with the remaining or new incoming messages to this particular reception FIFO's.

When the FIFO is not disabled then reception to the FIFO will continue.

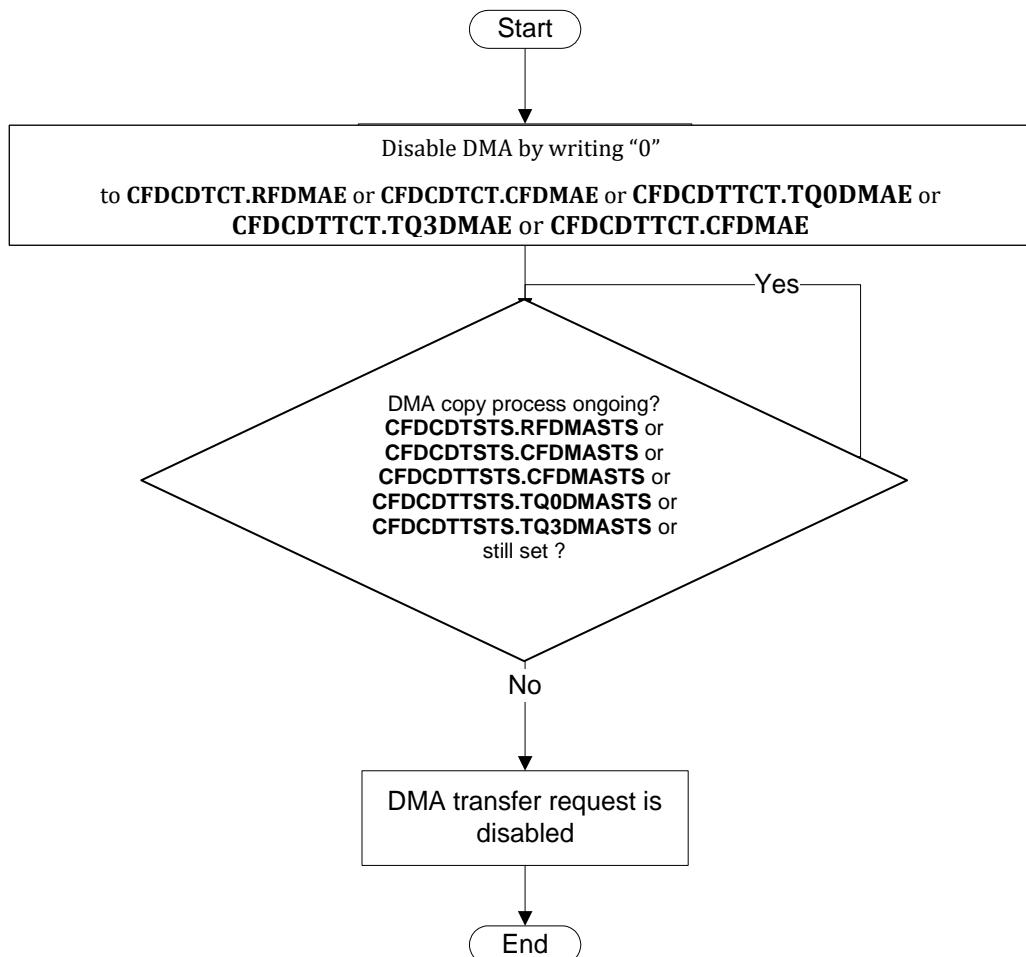


Figure 9.7 DMA disable flow

10 Reception and Transmission

10.1 Reception

In the RS-CAN-FD module, CAN messages received on any of the channels, will be stored in RX Message Buffers or in RX FIFO Buffers or Common FIFO Buffers configured in RX Mode or GW Mode depending upon the Acceptance Filter List entries:

up to $((n+1)*16)$ RX Message Buffers can be configured

8 RX FIFO Buffers available

up to $((n+1)*3)$ Common FIFO Buffers can be configured in RX mode or GW mode

up to $((n+1)*3)$ TX Queue can be configured in GW mode

10.1.1 Message storage in RX Message Buffers

When a message is successfully received and stored in a RX Message Buffer, the corresponding Newdata Flag is set in the RX Message Buffer Newdata Register.

The CAN Message can be read from the corresponding RX Message Buffer.

If a new message is stored into a RX Message Buffer before the previous message in this Message Buffer can be read, then the original message is overwritten. There is no mechanism for preventing a new message from overwriting the current message in the RX Message Buffer. If such a loss of messages is not acceptable, then RX FIFO should be used for storing related messages.

Note 1: Interrupts are not provided for the RX Message Buffers in the RS-CAN-FD module and hence the RX Message Buffer Newdata Registers should be accessed periodically to check if a new message has been stored in the RX Message Buffers.

Note 2: Unused Data Bytes will be filled with 8'h0 depending upon the DLC value.

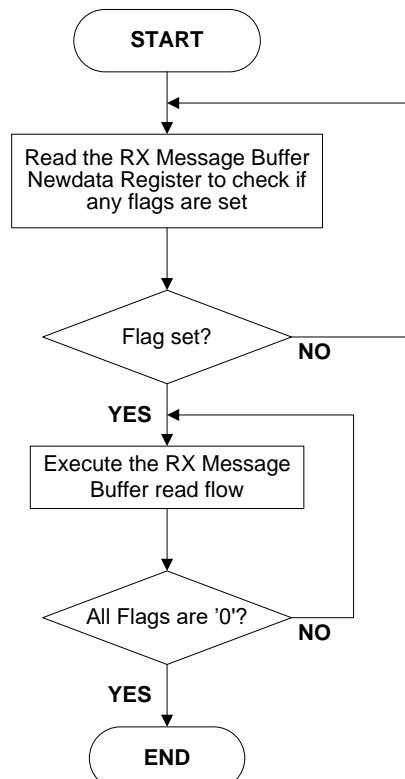


Figure 10.1: RX Message Buffer Message Access Flow

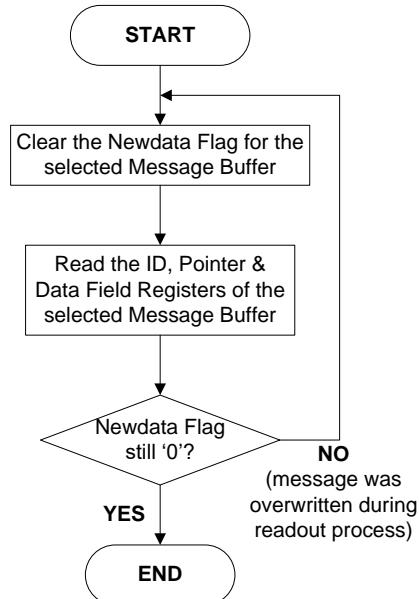


Figure 10.2: RX Message Buffer Read flow

10.1.2 Message storage in FIFO Buffers

The AFL entries for routing the received messages to RX FIFO buffers or Common FIFO Buffers configured in RX or GW Mode should be configured based on the requirements of the system.

The **CFDGAFLP1r.GAFLFDP[31:0]** field in the matching AFL entry selects the FIFO Buffers to which the related reception message will be stored.

When the received message is stored in one or more RX FIFO Buffers or Common FIFO Buffers configured in RX Mode or GW Mode, then the Message counter value is incremented in the corresponding RX FIFO Status Registers or Common FIFO Status Registers.

Depending upon the configuration of the FIFO Buffers, an Interrupt may also be generated.

The message can be read from the corresponding FIFO access registers.

Note: Since many messages can be stored in the FIFO Buffers, reading more than 1 message may be required to read the latest message stored in a FIFO Buffer.

If the Message count value matches the FIFO depth, then the FIFO Full Flag is set.

When the value 8'hFF is written to the corresponding FIFO Pointer Control Register, then the Message Count is decremented by 1.

Users should only write 8'hFF to the FIFO Pointer Control register after reading the complete message from the FIFO Access registers of the corresponding FIFO.

When all the messages stored in the FIFO are read, then the FIFO Empty flag is set.

If a new message is stored into the FIFO when the FIFO Message count matches the FIFO Depth (FIFO Full condition), the FIFO Message Lost flag is set and the new message will be lost (no overwrite of already stored messages will take place).

An appropriate value can be configured as warning level to generate an interrupt before the FIFO full condition occurs to avoid loss of a Message due to Overrun condition.

In GW mode, when a transmit/receive FIFO buffer is trying to receive a new message while the transmit/receive FIFO buffer is already full of data, the oldest data of the buffer will be overwritten with the message received or the message will be discarded. The behavior is determined by setting **CFDCFCCEd.CFMOWM** bit.

When **CFDCFCCEd.CFMOWM=0**:

When writing of data is required due to reception of a new message while a transmit/receive FIFO buffer is full of data, the received message will be discarded. And **CFDCFSTs.CFMLT** bit is set to 1.

When **CFDCFCCEd.CFMOWM=1**:

When writing of data is required due to reception of a new message while a transmit/receive FIFO buffer is full of data, the oldest data in the buffer will be overwritten with the received message. The read pointer of the transmit/receive FIFO buffer simultaneously moves to the next oldest message. Then, **CFDCFSTSd.CFMOW** bit is set to 1, which notifies that the oldest message has been overwritten with the received message.

In addition, in a case a CAN bus error or arbitration-lost for the transmitting message occurs in transmit/receive FIFO buffer full, the transmitting message is lost and re-transmission for the message is not performed. Then the read point moves to the next message automatically.

Users should not write change for this bit when the **CFDCFCCd.CFE** bit is 1'b1.

Common FIFO can set interrupt, when CAN frame reception is completed.

Common FIFO can set interrupt, when FIFO is in full status in RX mode or GW mode.

Note: The Message Lost can be set only in RX or GW mode by CAN side, the flag will not be set when the CPU side is overloading the FIFO buffers.

Note: When **CFDGAFLP0r.GAFLSRD i** (i=0to2) is set and the **CFDTXQCCin.TXQGWE** (i=0to2, n=0to7) is also set, a receiving frame is stored in the target TXQ as send data by routing.

The RX FIFO Buffers and the Common FIFO Buffers configured in RX or GW Mode can be disabled at any time by clearing the **CFDRFCCa.RFE** or **CFDCFCCd.CFE** bit in the RX FIFO Configuration / Control Registers and the Common FIFO Configuration / Control Registers.

When the **CFDRFCCa.RFE** or **CFDCFCCd.CFE** bit is cleared, then the message read and write pointers of the FIFO are cleared and are no longer active. Hence, all messages in the FIFO Buffers will be lost and no further messages can be stored into the FIFO.

When the RX FIFO Buffers or Common FIFO Buffers configured in RX Mode is assigned as DMA channel then the SW should not access the FIFO Access Register of this FIFO buffer or write 8'hFF to the FIFO Pointer Control Register (**CFDCFPCTRd.CFPC** or **CFDRFPCTRa.RFPC**), because this could lead to unintended FIFO message decrement. The DMA channel will control the FIFO decrement by automatically

Note: If the interrupt flag is set for a FIFO Buffer and then the FIFO is disabled, then the interrupt flag will not be cleared automatically. The interrupt flag should be cleared before disabling the FIFO.

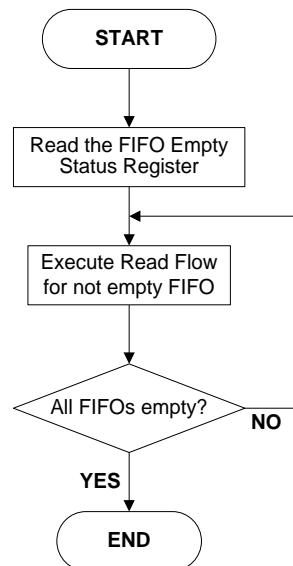


Figure 10.3: FIFO Buffer Message access Flow (example for polling case)

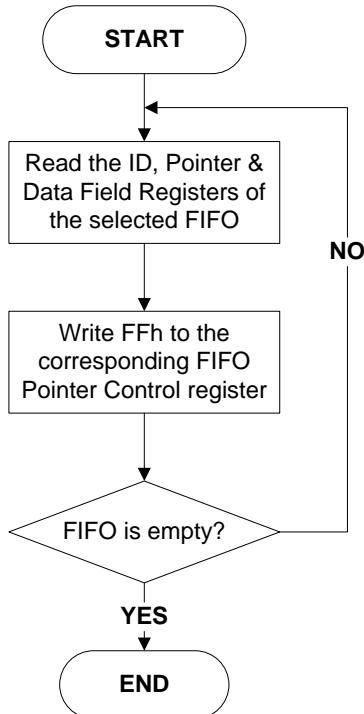


Figure 10.4: RX FIFO Buffer Read flow (example for polling case)

Note: When the next frame is received before clearing the completion interrupt flag of reception, the completion interrupt of reception is not set again.

Even if it clears an "interruption flag" after the completion processing of reception, the already received interrupt flag is not set.

It is necessary to perform the completion processing of reception even before the next completion of frame reception, and to clear an interruption flag.

When processing does not meet the deadline, after checking that receiving data is empty, interrupt flag is cleared and it checks that receiving data is empty again.

10.1.3 Timestamp

The Timestamp counter is a free-running counter that can be used to check reception time of an incoming message or transmission time of successful transmitted messages. The Timestamp counter value will be captured based on the **CFDGFDCFG.TSCCFG[1:0]** configuration (at the sample point of Start of Frame, point in time when the frame is valid, or for CAN-FD frames also at the sample point of the RES bit). For reception it is stored together with the message ID and Data into the target RX Message Buffer or RX/GW FIFO.

For transmit message the Timestamp counter value will be stored as part of the TX History List entry.

The counter can be clocked with the peripheral clock or with the CAN channel bit timing clock. The counter source clock can be configured via the **CFDGCFG.TSSS** bit of the Global Configuration Register. If it is 1'b0, the peripheral clock is used. If it is 1'b1, the selected CAN channel bit time clock is used.

The channel selection is done via the **CFDGCFG.TSBTCS** Bit of the Global Configuration Register.

Care has to be taken when using selected CAN channel bit time clock as clock source. In case of entering Channel Halt Mode or Channel Reset Mode, for this channel, the Timestamp counter is stopped. So, also for other CAN channels the Timestamp counter value will not be updated.

If peripheral clock is selected as Timestamp counter clock source Channel Modes are not influencing the Timestamp counter function.

The source clock for the Timestamp counter can be divided by a factor defined by the **CFDGCFG.TSP** bits (Timestamp Prescaler) in the Global Configuration Register.

The Timestamp counter can be reset to 16'h0000 via the **CFDGCTR.TSRST** bit (Timestamp Reset).

*Additionally, a specific value can be written into the Timestamp counter when the **CFDGCTR.TSWR** bit in the Global Control Register is set and the RS-CAN-FD module is in GL_HALT mode.*

10.2 Transmission

There are several possible transmission configurations for each channel:

Normal transmission

FIFO transmission

Gateway transmission

TX Queue transmission

A fixed number of transmission Message Buffers (64 TX Message Buffers) are dedicated for each channel. These Message Buffers are only used for transmission and cannot be configured for reception. Additionally transmission from TX Queue and/or Common FIFO in TX or GW mode can be configured in the following way (see Figure 10.5):

TX Queue: Up to 32 transmission Message Buffers for one channel can be grouped to form a TX Queue with a common access window.

Upper transmission Message Buffers are used to form the TXQ1 or TXQ3.

Lower transmission Message Buffers are used to form the TXQ0 or TXQ2.

Transmission control and status registers of these transmission Message Buffers should not be used.

One Channel has four TX Queue.

Each TXQ has each access window.

TXQ0 is transmission Message Buffer 0 of each channel.

TXQ1 is transmission Message Buffer 31 of each channel.

TXQ2 is transmission Message Buffer 32 of each channel.

TXQ3 is transmission Message Buffer 63 of each channel.

When using TXQ1 and TXQ0 simultaneously, the sum of the depths of TXQ1 and TXQ0 should not exceed 32.

When using TXQ3 and TXQ2 simultaneously, the sum of the depths of TXQ3 and TXQ2 should not exceed 32.

Common FIFO (TX/GW mode): each Common FIFO in TX or GW mode is linked to a dedicated channel.

Each channel has a fixed number of 3 Common FIFOs assigned to it. Within the channel, a Common FIFO configured in TX or GW mode, can be freely linked (assigned) between 32 and 63 transmission Message Buffers (only one FIFO to one transmission Message Buffer).

The Common FIFO Buffer then replaces the transmission Message Buffer linked to it.

Transmission control and status registers of these transmission Message Buffers should not be used.

Please refer to Figure 8.1 for information about Common FIFO Buffer assignment to related channels.

Note: Common FIFO buffers should not be linked to TX Message Buffers that are already part of a TX Queue.

Tx Message Buffer0	Tx Message Buffer0	Tx Queue 0	Tx Queue 0
Tx Message Buffer1	Tx Message Buffer1	Tx Queue 0	Tx Queue 0
Tx Message Buffer2	Tx Message Buffer2	Tx Queue 0	Tx Queue 0
Tx Message Buffer3	Tx Message Buffer3	Tx Queue 0	Tx Queue 0
Tx Message Buffer4	Tx Message Buffer4	Tx Queue 0	Tx Queue 0
Tx Message Buffer5	Tx Message Buffer5	Tx Queue 0	Tx Queue 0
Tx Message Buffer6	Tx Message Buffer6	Tx Queue 0	Tx Queue 0
Tx Message Buffer7	Tx Message Buffer7	Tx Queue 0	Tx Queue 0
Tx Message Buffer8	Tx Message Buffer8	Tx Queue 0	Tx Queue 0
Tx Message Buffer9	Tx Message Buffer9	Tx Queue 0	Tx Queue 0
Tx Message Buffer10	Tx Message Buffer10	Tx Queue 0	Tx Queue 0
Tx Message Buffer11	Tx Message Buffer11	Tx Queue 0	Tx Queue 0
Tx Message Buffer12	Tx Message Buffer12	Tx Queue 0	Tx Queue 0
Tx Message Buffer13	Tx Message Buffer13	Tx Queue 0	Tx Queue 0
Tx Message Buffer14	Tx Message Buffer14	Tx Queue 0	Tx Queue 0
Tx Message Buffer15	Tx Message Buffer15	Tx Queue 0	Tx Queue 0
Tx Message Buffer16	Tx Message Buffer16	Tx Message Buffer16	Tx Queue 0
Tx Message Buffer17	Tx Message Buffer17	Tx Message Buffer17	Tx Queue 0
Tx Message Buffer18	Tx Message Buffer18	Tx Message Buffer18	Tx Queue 0
Tx Message Buffer19	Tx Message Buffer19	Tx Message Buffer19	Tx Queue 0
Tx Message Buffer20	Tx Message Buffer20	Tx Message Buffer20	Tx Queue 0
Tx Message Buffer21	Tx Message Buffer21	Tx Message Buffer21	Tx Queue 0
Tx Message Buffer22	Tx Message Buffer22	Tx Message Buffer22	Tx Queue 0
Tx Message Buffer23	Tx Message Buffer23	Tx Message Buffer23	Tx Queue 0
Tx Message Buffer24	Tx Message Buffer24	Tx Queue 1	Tx Queue 0
Tx Message Buffer25	Tx Message Buffer25	Tx Queue 1	Tx Queue 0
Tx Message Buffer26	Tx Message Buffer26	Tx Queue 1	Tx Queue 0
Tx Message Buffer27	Tx Message Buffer27	Tx Queue 1	Tx Queue 0
Tx Message Buffer28	Tx Message Buffer28	Tx Queue 1	Tx Queue 0
Tx Message Buffer29	Tx Message Buffer29	Tx Queue 1	Tx Queue 0
Tx Message Buffer30	Tx Message Buffer30	Tx Queue 1	Tx Queue 0
Tx Message Buffer31	Tx Message Buffer31	Tx Queue 1	Tx Queue 0
Tx Message Buffer32	Tx Message Buffer32	Ch. Common FIFO 0	Tx Queue 2
Tx Message Buffer33	Ch. Common FIFO 0	Tx Message Buffer33	Tx Queue 2
Tx Message Buffer34	Tx Message Buffer34	Tx Message Buffer34	Tx Queue 2
Tx Message Buffer35	Ch. Common FIFO 1	Tx Message Buffer35	Tx Queue 2
Tx Message Buffer36	Tx Message Buffer36	Tx Message Buffer36	Tx Queue 2
Tx Message Buffer37	Tx Message Buffer37	Ch. Common FIFO 1	Tx Queue 2
Tx Message Buffer38	Tx Message Buffer38	Tx Message Buffer38	Tx Queue 2
Tx Message Buffer39	Tx Message Buffer39	Tx Message Buffer39	Tx Queue 2
Tx Message Buffer40	Tx Message Buffer40	Tx Message Buffer40	Tx Queue 2
Tx Message Buffer41	Tx Message Buffer41	Tx Message Buffer41	Tx Queue 2
Tx Message Buffer42	Tx Message Buffer42	Tx Message Buffer42	Tx Queue 2
Tx Message Buffer43	Tx Message Buffer43	Ch. Common FIFO 2	Tx Queue 2
Tx Message Buffer44	Tx Message Buffer44	Tx Message Buffer44	Tx Queue 2
Tx Message Buffer45	Tx Message Buffer45	Tx Message Buffer45	Tx Queue 2
Tx Message Buffer46	Tx Message Buffer46	Tx Message Buffer46	Tx Queue 2
Tx Message Buffer47	Tx Message Buffer47	Tx Message Buffer47	Tx Queue 2
Tx Message Buffer48	Tx Message Buffer48	Tx Message Buffer48	Tx Queue 3

Tx Message Buffer49	Tx Message Buffer49	Tx Message Buffer49	Tx Queue 3
Tx Message Buffer50	Tx Message Buffer50	Tx Message Buffer50	Tx Queue 3
Tx Message Buffer51	Tx Message Buffer51	Tx Message Buffer51	Tx Queue 3
Tx Message Buffer52	Tx Message Buffer52	Tx Queue 3	Tx Queue 3
Tx Message Buffer53	Tx Message Buffer53	Tx Queue 3	Tx Queue 3
Tx Message Buffer54	Tx Message Buffer54	Tx Queue 3	Tx Queue 3
Tx Message Buffer55	Tx Message Buffer55	Tx Queue 3	Tx Queue 3
Tx Message Buffer56	Tx Message Buffer56	Tx Queue 3	Tx Queue 3
Tx Message Buffer57	Tx Message Buffer57	Tx Queue 3	Tx Queue 3
Tx Message Buffer58	Tx Message Buffer58	Tx Queue 3	Tx Queue 3
Tx Message Buffer59	Tx Message Buffer59	Tx Queue 3	Tx Queue 3
Tx Message Buffer60	Tx Message Buffer60	Tx Queue 3	Tx Queue 3
Tx Message Buffer61	Tx Message Buffer61	Tx Queue 3	Tx Queue 3
Tx Message Buffer62	Tx Message Buffer62	Tx Queue 3	Tx Queue 3
Tx Message Buffer63	Ch. Common FIFO 2	Tx Queue 3	Tx Queue 3
Normal Transmission Message Buffer Configuration	Normal and FIFO Transmission Configuration (example for Common FIFO linking)	Normal, FIFO and Tx-Queue Transmission Configuration (example for Common FIFO linking and Tx Queue 0 depth of 16 and Tx Queue 1 depth of 8 Tx Queue 3 depth of 12 and Tx Message Buffers)	Tx-Queue Transmission Configuration (example for Tx Queue 0 depth of 32 and Tx Queue 2 depth of 16 and Tx Queue 3 depth of 16)

Figure 10.5 Channel Transmission Message Buffer Configuration

10.2.1 Transmission Priority

If two or more transmission Message Buffers of a channel are configured for transmission, then the transmission priority in the RS-CAN-FD module can be selected from the following two modes:

CAN ID priority

Message Buffer number priority

The transmission priority mode is common for all Message Buffers and all CAN channels. It can be configured via the **CFDGCFG.TPRI** bit in the Global Configuration Register.

For Message Buffer number priority transmission, the smallest Message Buffer number with transmission request has the highest priority for transmission. This also includes the TX Message Buffers linked to the Common FIFO Buffers configured in TX mode or GW mode.

However, Message Buffer number priority should not be used if TX Queue is enabled.

For CAN ID priority transmission, ID priority complies with the CAN bus arbitration rule (as specified in ISO 11898-1 specification). All TX Message Buffers can enter the ID priority comparison for Message Buffers configured for transmission. This also includes the TX Message Buffers linked to the Common FIFO Buffers configured in TX mode or GW mode and includes the TX Queue Message Buffers.

If the ID of two or more Message Buffers is the same, then the smaller Message Buffer number will have higher priority for transmission.

Note: For Common FIFO Buffers configured in TX mode or GW mode, only the message currently being pointed to by the FIFO Read Pointer can be included in the transmission arbitration.

If the message is being transmitted from the FIFO, then the next pending Message within the same FIFO will be considered in the transmission arbitration.

In contrast to this, all transmission Message Buffers of a TX Queue will participate in internal transmission arbitration.

Figure 10.6 below shows the transmission configuration flow.

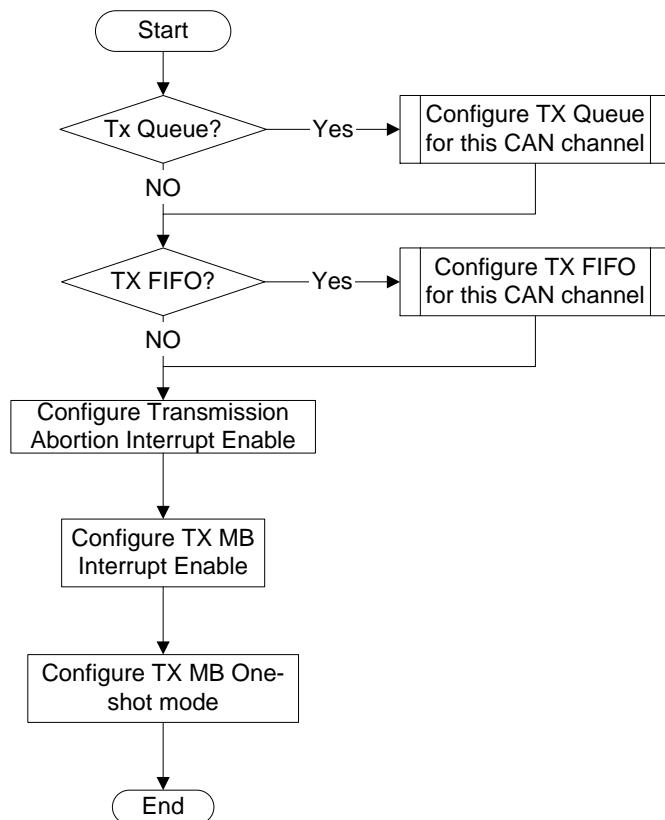


Figure 10.6 Transmission Configuration Flow

10.2.2 Normal Transmission

Each transmission Message Buffer has two modes of message transmission:

1. Regular Transmission Mode

If the Message Buffer is placed in regular transmission mode, the data frame or remote frame set in that Message Buffer can be transmitted.

Completion of regular transmission can be checked through the related TX Message Buffer Transmission Result Flag bits (**CFDTMSTSj.TMTRF**) in the TX Message Buffer Status Registers. These bits are set to 2'b10 or 2'b11 when the regular transmission is successful.

When arbitration is lost or an error occurs, message transmission will be attempted further if no transmission abort request is set for this transmission Message Buffer.

New internal transmission arbitration for this channel will be performed considering all Message Buffers with transmission request.

2. One Shot Transmission Mode

When the **CFDTMCi.TMOM** bit of the TX Message Buffer Control Registers is set for a transmission Message Buffer, then the Message Buffer is placed in one-shot transmission mode and attempts to transmit a message only once.

Completion of one shot transmission can be checked through the related TX Message Buffer Transmission Result Flag bits (**CFDTMSTSj.TMTRF**) in the TX Message Buffer Status Registers. The **CFDTMSTSj.TMTRF** bits are set to 2'b10 or 2'b11 when the one shot transmission is successful.

The **CFDTMSTSj.TMTRF** bits are set to 2'b01 when arbitration is lost or an error occurs during the transmission of the related Message Buffer.

Further message transmission will not be attempted in this case.

The regular transmission request procedure after a configuration is shown in Figure 10.7.

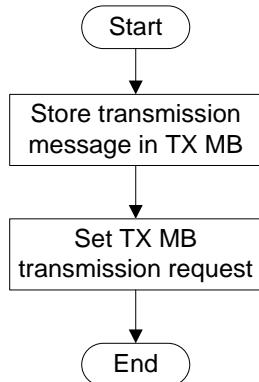


Figure 10.7: Transmission request procedure using normal TX Message Buffer mode

10.2.2.1 TX Message Buffer control register setting

Table 10.1 shows configuration of the normal CAN transmission mode.

Transmission Request CFDTMCi.TMTR	Transmission Abortion Request CFDTMCi.TMTAR	One Shot Enable CFDTMCi.TMOM	Communication activity
0	0	0	Message Buffer disabled
0	0	1	Message Buffer disabled
1	0	0	configured as a transmission Message Buffer for a data frame or a remote frame
1	0	1	configured as a one shot transmission Message Buffer for a data frame or a remote frame
1	1	0	Transmission abortion requested
1	1	1	One shot transmission abortion requested

Table 10.1 Configuration of CAN transmission mode

The Configuration bits can be configured in the TX Message Buffer Control Registers.

The Figure 10.8 shows timings for successful transmission for two Message Buffers of one channel.

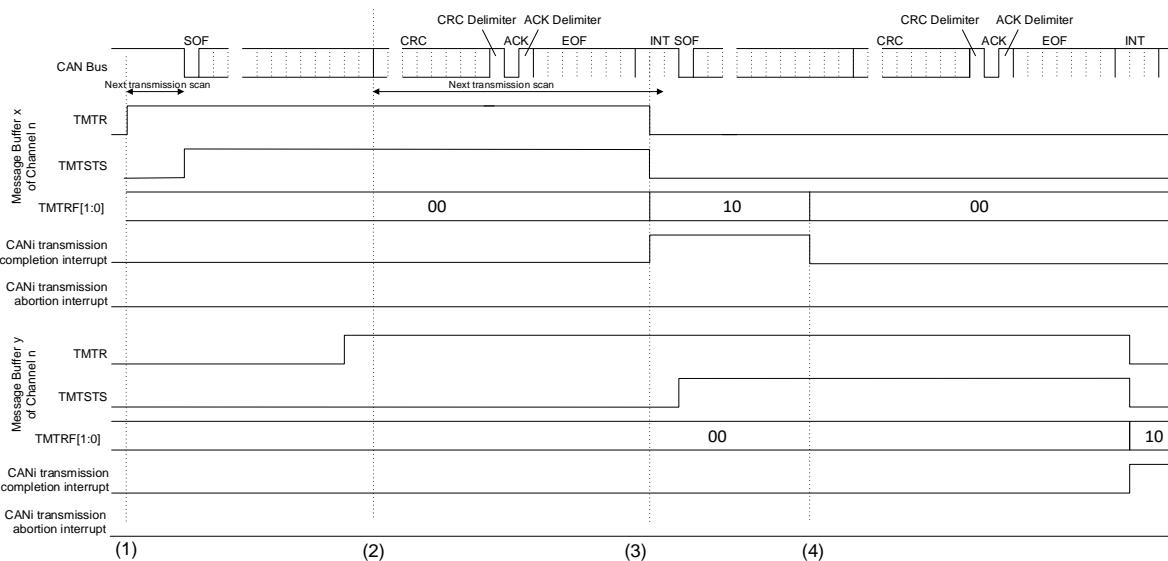


Figure 10.8 Timing of request and flag bits for successful transmission

- If the **CFDTMCi.TMTR** bit in the TX Message Buffer Control Registers is set in the bus idle state, Message Buffer scanning procedure starts to decide the highest priority Message Buffer for transmission.

When the transmission Message Buffer is decided, the **CFDTMSTSj.TMTSTS** bit in the related TX Message Buffer Status Registers is set (Transmitting/Transmitter), and CAN channel starts the transmission**.

- At 1st Bit of CRC, the transmission scanning procedure starts for the next possible transmission when pending transmission requests exist.

The scan time could be delayed due to other transmission scan on other channels, but it will be finished before Intermission 3 to be able to continue transmission without any gaps.

- If the message has been successfully transmitted, the **CFDTMSTSj.TMTRF[1:0]** bits in the corresponding TX Message Buffer Status Registers are set to 2'b10 and **CFDTMSTSj.TMTSTS** and the **CFDTMCi.TMTR** bits are cleared.

When the TMIE bits in the TX Message Buffer Interrupt Enable Configuration Registers is set (Interrupt enabled), the CAN successful transmission interrupt request is generated.

To clear the related interrupt line the **CFDTMSTSj.TMTRF** flag bits have to be cleared.

- Before starting the next transmission, clear the **CFDTMSTSj.TMTRF** bits. Load the next message in the transmission Message Buffer and set the **CFDTMCi.TMTR** bit again.

CFDTMCi.TMTR bit cannot be set again before **CFDTMSTSj.TMTRF[1:0]** bits are cleared.

** If arbitration is lost after the CAN channel starts the transmission, the **CFDTMSTSj.TMTSTS** bit is cleared. The transmission scanning procedure is performed again to search for the highest priority transmission Message Buffer from the beginning of the 1st CRC bit.

If an error occurs either during the transmission or following the loss of arbitration, then during Error Frame, the transmission scanning procedure is performed again to search for the highest priority transmission Message Buffer.

Note: The setting point of **CFDTMSTSj.TMTSTS** is not always fixed at the start of the SOF. It may be delayed up to the start of the standard ID due to the synchronisation logic implemented for the PLL bypass.

The Figure 10.9 shows timings for transmission abort for two Message Buffers of one channel.

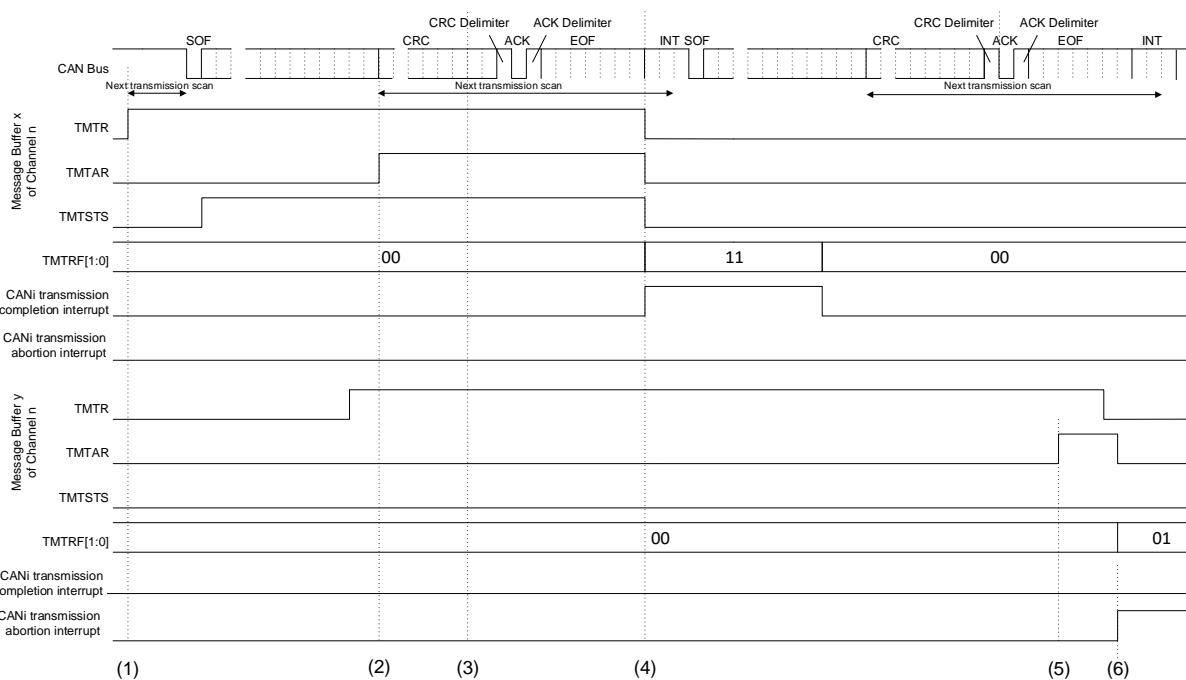


Figure 10.9 Timing of request and flag bits for transmission abort

- (1) If the **CFDTMCi.TMTR** bit in the TX Message Buffer Control Registers is set in the bus idle state, Message Buffer scanning procedure starts to decide the highest priority Message Buffer for transmission.

When the transmission Message Buffer is decided, the **CFDTMSTSj.TMTSTS** bit in the TX Message Buffer Status Registers is set (Transmitting/Transmitter), and CAN channel starts the transmission**.

- (2) If the **CFDTMCi.TMTAR** bit is set when the related Message Buffer is already selected for transmission or currently transmitting then the message will not be aborted, if no error occurs or arbitration is lost.

(3) At 1st CRC bit, the transmission scanning procedure starts for the next transmission. In this example timing chart Message Buffer y is not selected as next transmission Message Buffer.

The scan time could be delayed due to other transmission scan on other channels, but it will be finished before Intermission 3 to be able to continue transmission without any gaps.

- (4) If the message has been successfully transmitted, the **CFDTMSTSj.TMTRF[1:0]** bits in the corresponding TX Message Buffer Status Registers are set to 2'b11 and **CFDTMSTSj.TMTSTS** and the **CFDTMCi.TMTR** bits are cleared.

When the TMIE bits in the TX Message Buffer Interrupt Enable Configuration Registers is set (Interrupt enabled), the CAN successful transmission interrupt is generated.

To clear the related interrupt line the **CFDTMSTSj.TMTRF[1:0]** bits has to be cleared.

- (5) Another CAN node is transmitting on the CAN bus (**CFDTMSTSj.TMTSTS** not set!): if the **CFDTMCi.TMTAR** bit is set when the related channel is under transmission scan then the transmission request cannot be cleared.

- (6) After internal processing time the transmission is aborted and the **CFDTMSTSj.TMTRF[1:0]** bits are set to 2'b01.

If the Message Buffer is not transmitting or selected as next transmission Message Buffer or under transmit scan, then the abort is immediately accepted and the corresponding **CFDTMSTSj.TMTRF[1:0]** bits in the TX Message Buffer Status Registers are set to 2'b01.

In addition, **CFDTMCi.TMTR**, and **CFDTMCi.TMTAR** bits are cleared automatically.

When the transmission abort interrupt enable TAIE bit of the related Channel Control Register is set then an interrupt is generated for successful transmission abort.

To clear the related interrupt line the **CFDTMSTSj.TMTRF[1:0]** bits have to be cleared.

** If arbitration is lost after the CAN channel starts the transmission, the **CFDTMSTSj.TMTSTS** is cleared.

The transmission scanning procedure is performed again to search for the highest priority transmission Message Buffer from the beginning of the 1st CRC bit.

If an error occurs, either during the transmission, or following the loss of arbitration, then during Error Frame, the transmission scanning procedure is performed again to search for the highest priority transmission Message Buffer.

10.2.3 TX FIFO or GW FIFO Transmission

Three common FIFO buffers are assigned to each channel. The 3 FIFO buffers could be linked to any normal TX Message Buffer position for this channel by the **CFDCFCCd.CFTML** bits in the Common FIFO Configuration / Control Register if configured in TX or GW mode.

When the transmission scan starts and the FIFO Buffer corresponding to this TX Message Buffer is enabled, then the relevant message in the FIFO Buffer will participate in the transmission scan.

Configuration of a TX Message Buffer linked to a FIFO Buffer configured in TX or GW mode should not be done.

10.2.3.1 TX FIFO Operation

CAN Messages can be written into the TX FIFO by writing to the corresponding FIFO Access registers.

When the value 8'hFF is written into the corresponding FIFO Pointer Control Register, then the Message Count of the related FIFO is incremented by 1.

Users should only write to the FIFO Pointer Control register after writing the complete message to the corresponding FIFO Access registers.

If the Message count matches the FIFO Depth, then the FIFO Full flag is set.

The oldest message in the TX FIFO is included in the scan for transmission by the corresponding RS-CAN-FD module channel logic.

When a message is successfully transmitted from the TX FIFO, the Message Count value is decremented by 1.

When all the messages from the FIFO are transmitted, the FIFO Empty flag is set.

The Interrupt generation conditions for the TX FIFO buffers can be configured by configuring the **CFDCFCCd.CFIM** bit in the corresponding Common FIFO Configuration / Control Registers.

If **CFDCFCCd.CFIM** bit is 1'b0, then interrupt is generated when last message is successfully transmitted from the TX FIFO buffer.

If **CFDCFCCd.CFIM** bit is 1'b1, then interrupt is generated for every successfully transmitted message from the TX FIFO Buffer.

Common FIFO can set interrupt, when CAN frame transmitted is completed.

The Common FIFO Buffers configured in TX Mode can be disabled by clearing the **CFDCFCCd.CFE** bit in the Common FIFO Configuration / Control Registers. If this bit is cleared to 1'b0, the FIFO empty flag is set as described below:

immediately if the message from the TX FIFO is neither scheduled for the next transmission nor in transmission

following the transmission completion, the detection of an error on the CAN bus, loss of arbitration or transition to Channel or Global Halt Mode if the transmission from the TX FIFO is already scheduled for transmission or already in transmission

Note: The Common FIFO buffer is considered as disabled after clearing the **CFDCFCCd.CFE** bit only when the Empty flag is set for the corresponding Common FIFO Buffer.

Other possible messages pending from the TX FIFO are lost and their transmission needs to be requested again. Before **CFDCFCCd.CFE** is set again ensure that **CFDCFSTSd.CFEMP** bit is set and that there is no pending abort from the TX FIFO.

When the **CFDCFCCd.CFE** bit is cleared, the message read and write pointers of the FIFO are cleared and are no longer active. Hence, all messages in the FIFO buffers will be lost and no further message can be stored into the FIFO.

The FIFO transmission request procedure after a configuration is shown in Figure 10.10.

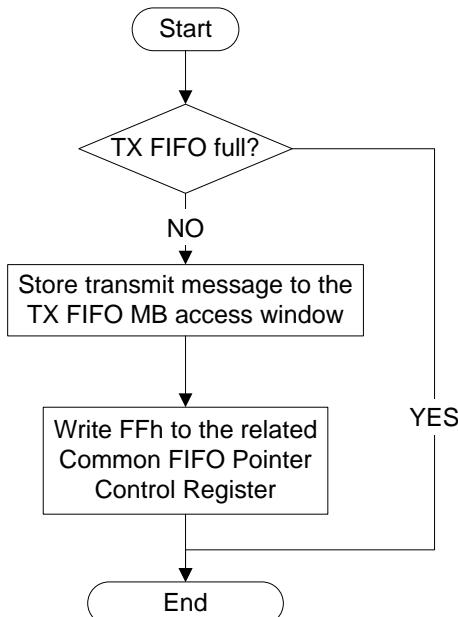


Figure 10.10 : TX FIFO transmission request procedure

10.2.3.2 GW FIFO Operation

The AFL entries for routing the received messages to GW FIFO buffers should be configured based on the requirements of the system. The matching AFL entry selects the GW FIFO Buffer for storage of a received message on any of the CAN channels.

When a message is successfully received and stored in a GW FIFO Buffer, then the FIFO Message Count in the corresponding FIFO Status Register is incremented by 1.

If the Message Count matches the FIFO Depth, then the FIFO Full flag is set.

The oldest message in the GW FIFO is included in the scan for transmission by the corresponding RS-CAN-FD module channel logic.

When a message is successfully transmitted from the GW FIFO, the Message Count value is decremented by 1.

When all the messages from the GW FIFO are transmitted, the FIFO Empty flag is set.

If a new message is stored into the FIFO when the FIFO Message count matches the FIFO Depth (FIFO Full condition), the FIFO Message Lost flag is set and the new message will be lost (no overwrite of already stored messages will take place).

In GW mode, when a transmit/receive FIFO buffer is trying to receive a new message while the transmit/receive FIFO buffer is already full of data, the oldest data of the buffer will be overwritten with the message received or the message will be discarded. The behavior is determined by setting **CFDCFCCEd.CFMOWM** bit.

When **CFDCFCCEd.CFMOWM=0**:

When writing of data is required due to reception of a new message while a transmit/receive FIFO buffer is full of data, the received message will be discarded. And **CFDCFSTSd.CFMLT** bit is set to 1.

When **CFDCFCCEd.CFMOWM=1**:

When writing of data is required due to reception of a new message while a transmit/receive FIFO buffer is full of data, the oldest data in the buffer will be overwritten with the received message. The read pointer of the transmit/receive FIFO buffer simultaneously moves to the next oldest message. Then, **CFDCFSTSd.CFMOW** bit is set to 1, which notifies that the oldest message has been overwritten with the received message.

In addition, in a case a CAN bus error or arbitration-lost for the transmitting message occurs in transmit/receive FIFO buffer full, the transmitting message is lost and re-transmission for the message is not performed. Then the read point moves to the next message automatically.

The Interrupt generation conditions for the GW FIFO buffers can be configured by configuring the **CFDCFCCd.CFIM** bit in the corresponding Common FIFO Configuration / Control Registers.

If **CFDCFCCd.CFIM** bit is 1'b0, then RX interrupt flag is set when FIFO counter increments and reaches value configured by **CFDCFCCd.CFIGCV** and the TX interrupt flag is set when FIFO transmits the last message successfully

If **CFDCFCCd.CFIM** bit is 1'b1, then RX Interrupt flag is set at the end of storage of every received message and TX interrupt flag is set if a message is successfully transmitted from the FIFO.

Common FIFO can set interrupt, when CAN frame transmitted is completed.

Common FIFO can set interrupt, when CAN frame reception is completed.

Common FIFO can set interrupt, when FIFO is in full status in RX mode or GW mode.

In the case of **CFDCFCCEd.CFBME=1**, it becomes FIFO buffering mode, send data is stored in Common FIFO, and transmission is stopped. Transmission will be started if it is set as **CFDCFCCEd.CFBME=0**.

The Common FIFO Buffers configured in GW Mode can be disabled by clearing the **CFDCFCCd.CFE** bit in the Common FIFO Configuration / Control Register. If this bit is cleared, the GW FIFO becomes empty as described below:

immediately if the message from the GW FIFO is neither scheduled for the next transmission nor in transmission

following the transmission completion, the detection of an error on the CAN bus, loss of arbitration or transition to Channel or Global Halt Mode if the transmission from the GW FIFO is already scheduled for transmission or already in transmission

Other possible messages pending from the GW FIFO are lost.

Before **CFDCFCCd.CFE** is set again ensure that the **CFDCFSTSd.CFEMP** bit is set and that there is no pending abort from the GW FIFO.

When the **CFDCFCCd.CFE** bit is cleared and the **CFDCFSTSd.CFEMP** bit is set, the message read and write pointers of the GW FIFO are cleared and are no longer active . Hence all messages in the GW FIFO buffers will be lost and no further message can be stored into the GW FIFO.

In applications intended to be used as CAN-to-CAN gateways it would be useful if the Error State Indication (ESI) information of the routing messages is not replaced by the sending node Error State Indication. For this, each channel has the control function register **CFDCnFDCFG.ESIC** to replace their own ESI information by the routing ESI information.

Note: If the sending node is error passive then the ESI bit will be sent anyway as Error passive (ESI =1).

10.2.3.3 Interval Timer for FIFO Transmission

For each Common FIFO in TX or GW mode it is possible to specify a delay between two consecutive messages that are configured for transmission from the same FIFO buffer. This delay is, called interval time. This interval time starts after the first message has been successfully transmitted from the FIFO buffer after the **CFDCFCCd.CFE** bit is set.

When the Common FIFO in TX or GW mode is enabled, then the first message will be transmitted without considering this interval time.

The Interval Timer will stop counting when:

FIFO is disabled by clearing the **CFDCFCCd.CFE** bit.

CAN channel is in CH_RESET mode.

The interval time is specified by the **CFDCFCCd.CFITT** value in the Common FIFO Configuration / Control Register and can be specified from 0 to 255 timer units.

The timer unit can be defined based on two different source clocks for the interval timer. To disable the interval timer for FIFO transmission a value of 0 should be selected.

The timer source can be selected by the configuration bit CFITSS in the Common FIFO Configuration / Control Register. For the timer source the CAN Bit Timing clock of the FIFO related channel or a global

reference clock could be selected.

If CAN channel bit time clock is configured as clock source and the CAN channel enters CH_HALT or CH_RESET or CH_SLEEP mode, then the Interval Timer is stopped for that channel.

If peripheral clock is selected as Interval Timer clock source, then the Interval Timer is stopped only when the CAN channel is in CH_RESET or CH_SLEEP mode.

The reference clock can be used to configure the interval time in fixed time units. It is based on the peripheral clock. The reference clock prescaler value **CFDGCFG.ITRCP** in the Global Configuration Register defines the relation between the peripheral clock frequency/period and the reference clock period.

Please refer to Table 10.2 for **CFDGCFG.ITRCP** configuration values to achieve different reference clock periods based on the peripheral clock frequency/period.

Reference clock Peripheral clock	1µs	100µs	500µs
16MHz / 62.5 ns	16	1600	8000
20MHz / 50 ns	20	2000	10000
32MHz / 31.25 ns	32	3200	16000
50MHz / 20 ns	50	5000	25000

Table 10.2 Configuration example for the FIFO interval timer reference clock

Additionally the reference clock resolution can be specified by the Interval Timer Reference Clock Resolution Value **CFDCFCCd.CFITR** in the Common FIFO Configuration / Control Register.

The interval time is based on the reference clock period multiplied by the configured value (x1 or x10).

The reference clock based interval timer can be used to follow the requirements of the ISO 15765-2 Separation Time. The whole range for the Separation Time from 100µs to 127 ms can be covered.

The specified interval time starts after successful transmission event (after EOF7 state of the CAN protocol). When the interval time has elapsed, the next transmission request is raised by the related TX/GW FIFO. Hence, the interval time defines the minimum time between two messages transmitted from one FIFO.

The next message will earliest be sent after this interval time.

The Figure 10.11 shows an example timing of the internal processing.

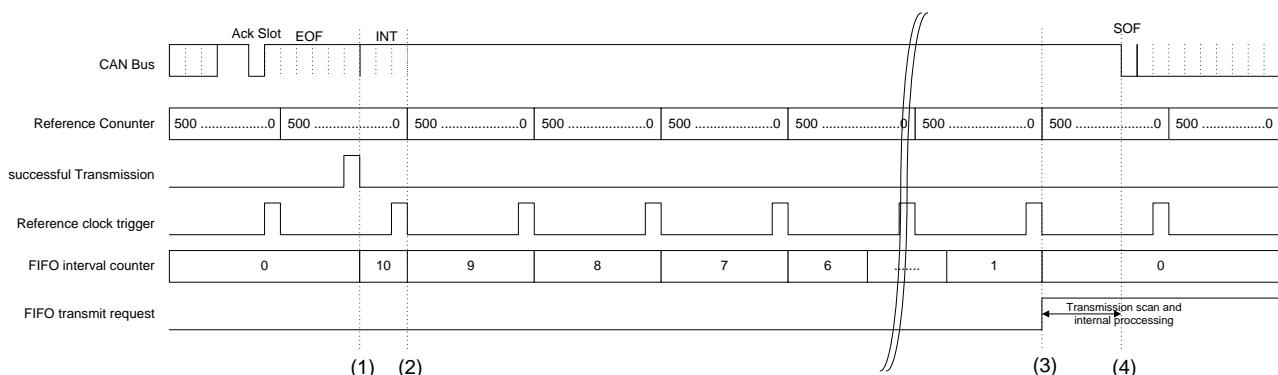


Figure 10.11 : Example for interval processing time

The configuration for this timing above is following:

Peripheral clock frequency = 50MHz

Interval Timer Reference clock (**CFDGCFG.ITRCP**)= 500 times

Reference clock due to the settings above = 10µs

Common FIFO interval Timer Source Selection (**CFDCFCCd.CFITSS**)= 0

Common FIFO Interval Timer Resolution (**CFDCFCCd.CFITR**)= 0

Common FIFO Interval Transmission time (**CFDCFCCd.CFITT**)= 10 times

Theoretical Message separation interval = 100 µs

(1) Internal FIFO interval timer is restarted with the occurrence of Successful transmission result. This restart is not synchronized to the Reference clock trigger. Therefore the first interval is counting less or equal to one Reference clock interval.

(2) With the next Reference clock trigger the FIFO interval timer is decremented and so on.

(3) When the FIFO interval timer reached the value “zero” the FIFO Transmit request is set.

(4) When the FIFO is selected for transmission then the transmission will start soon. Due to internal processing this usually takes less than 3 CAN bit time, between internal FIFO transmit request set (3) and actual transmission.

In worst case when multi events like reception scan, internal message routing, transmit scan on all channels happen, then it could take up to 1152 peripheral clock cycles.

$$\begin{aligned} \text{Calculation: } & (1 * \text{Routing} + (n+1) * \text{OTB} + (n+1) * \text{TXSCAN} + (n+1) * \text{THL}) \\ & = (192 + (n+1) * 22 + (n+1) * 94 + (n+1) * 4) \end{aligned}$$

As shown in Figure 10.11, it is not guaranteed that the minimum interval is always equal to the configured value. If a minimum time must never be breached, users should configure **CFDCFCCd.CFITT** to required minimum value+1

If further TX Message Buffers or TX/GW FIFOs are configured for transmission for the same channel the real delay between two messages transmitted from a TX FIFO can be much longer than specified by the interval time due to higher priority message transmission from these TX Message Buffers or TX/GW FIFOs.

Figure 10.12 shows a block diagram of the FIFO interval time generation circuit.

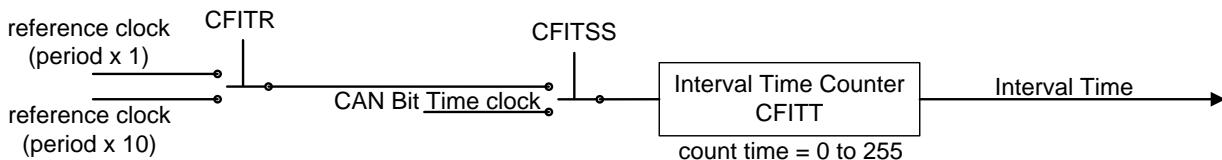


Figure 10.12 : Block Diagram of FIFO interval timer

10.2.4 TX Queue

Each enabled TX Queue for a specific channel consists of 3 to 32 TX Message Buffers, which are accessed via one access window. One Channel has four TX Queue. One TX Queue can be configured with a depth of 3 up to 32 buffer and it is using the TX Message Buffer No. 0 as access window (Referred to TXQ0). The second TX Queue can be configured with a depth of 3 up to 32 buffer and it is using the TX Message Buffer No. 31 as access window (Referred to TXQ1). The third TX Queue can be configured with a depth of 3 up to 32 buffer and it is using the TX Message Buffer No. 32 as access window (Referred to TXQ2). The fourth TX Queue can be configured with a depth of 3 up to 32 buffer and it is using the TX Message Buffer No. 63 as access window (Referred to TXQ3). All the Message of TXQ0, TXQ1, TXQ2 and TXQ3 enter the priority comparison for the transmission, which should be only ID Priority (**CFDGCFG.TPRI = 1'b0**). The registers for TXQ0 are **CFDTXQCC0 [n]**, **CFDTXQSTS0 [n]**, and **CFDTXQPCTR0 [n]**. The registers for TXQ1 are **CFDTXQCC1 [n]**, **CFDTXQSTS1 [n]**, and **CFDTXQPCTR1 [n]**. The registers for TXQ2 are **CFDTXQCC2 [n]**, **CFDTXQSTS2 [n]**, and **CFDTXQPCTR2 [n]**. The registers for TXQ3 are **CFDTXQCC3 [n]**, **CFDTXQSTS3 [n]**, and **CFDTXQPCTR3 [n]**. As access window TX Message Buffer No.63 (TXQ3) or TX Message Buffer No.32 (TXQ2) or TX Message Buffer No.31 (TXQ1) or TX Message Buffer No.0 (TXQ0) is used, refer to related access registers TX Message Buffer ID Registers (**TMID[m]**), TX Message Buffer Pointer Registers (**TMPTR[m]**), TX Message Buffer Data Field 0 Registers and TX Message Buffer Data Field 1 Registers (**TMDF[0:1][m]**)

The depth of each TXQ0 Buffer can be configured by writing to the **CFDTXQCC0n.TXQDC[4:0]** bits of the

D012250_111_S01 VERSION 1.021.01

STATUS: Issued

©RENESAS SYSTEM DESIGN

FORM EM-PRJ3A (APRIL 2017) VERSION 4.1A

LEVEL 3 – CONFIDENTIAL

PAGE 588 OF 648

TX Queue Configuration / Control Register. TXQ0 can set from TXMB0 to TXMB31 as a queue buffer at the maximum.

The 31 available options for depth configuration are:

5'b00000: TX Queue disabled

5'b00001: reserved

5'b00010: 3 Messages

o:

o:

5'b11101: 30 Messages

5'b11110: 31 Messages

5'b11111: 32 Messages

The depth of each TXQ1 Buffer can be configured by writing to the **CFDTXQCC1n.TXQDC[4:0]** bits of the TX Queue Configuration / Control Register.

TXQ1 can set from TXMB31 to TXMB0 as a queue buffer at the maximum.

The 31 available options for depth configuration are:

5'b00000: TX Queue disabled

5'b00001: reserved

5'b00010: 3 Messages

o:

o:

5'b11101: 30 Messages

5'b11110: 31 Messages

5'b11111: 32 Messages

The depth of each TXQ2 Buffer can be configured by writing to the **CFDTXQCC2n.TXQDC[4:0]** bits of the TX Queue Configuration / Control Register.

TXQ2 can set from TXMB32 to TXMB63 as a queue buffer at the maximum.

The 31 available options for depth configuration are:

5'b00000: TX Queue disabled

5'b00001: reserved

5'b00010: 3 Messages

o:

o:

5'b11101: 30 Messages

5'b11110: 31 Messages

5'b11111: 32 Messages

The depth of each TXQ3 Buffer can be configured by writing to the **CFDTXQCC3n.TXQDC[4:0]** bits of the TX Queue Configuration / Control Register.

TXQ3 can set from TXMB63 to TXMB32 as a queue buffer at the maximum.

The 31 available options for depth configuration are:

5'b00000: TX Queue disabled

5'b00001: reserved

5'b00010: 3 Messages

o:

o:

5'b11101: 30 Messages

5'b11110: 31 Messages

5'b11111: 32 Messages

When using TXQ1 and TXQ0 simultaneously, the depth of TXQ is 32 or less in total.

When using TXQ3 and TXQ2 simultaneously, the depth of TXQ is 32 or less in total.

Users should not access all the TX Message Buffers forming the TX Queue directly (except TX Message Buffer No. 63, TX Message Buffer No. 32, TX Message Buffer No. 31 and TX Message Buffer No. 0, which acts as TX Queue access window).

When **CFDGAFLP0r.GAFLSRD i** (i=0to2) is set and the **CFDTXQCCin.TXQGWE** (i=0to2, n=0to7) is also set, a receiving frame is stored in the target TXQ as send data by routing.

When **CFDTXQCCn.TXQOWE** bit is 1'b1, the TX queue is in TX queue overwrite mode. If the message of the same ID is stored in TX Queue when a frame is received and it is stored in TX Queue, an old message will be overwritten by a new message. Therefore, an old message is not transmitted. When the old message of the same ID is transmitting and a CAN bus error and an arbitration-lost occur, the message of old ID is not resent.

When users use the function in GW mode and TX queue overwrite mode, the depth of TXQ (**CFDTXQCC0n.TXQDC**) should be configured to the value which is the various number of ID which is used in the TX queue plus 3. If it accesses by routing in gateway mode when a TXQ buffer is full, **CFDTXQSTS.TXQMLT** will be set and send data will be thrown away. Then the function is valid for the standard ID frame and is invalid for the extended ID frame.

Explanation of operation of the TXQ same ID over-writing function in GW mode is shown below.

TXQ0 depth = 6 buffer , kind of ID is 3, TXQ0 Gwmode

①Three frames are stored.

now transmitting	ID0	TXMB0
next transmission	ID1	TXMB1
waiting for txscan	ID2	TXMB2
	TXMB3	
	TXMB4	
	TXMB5	



②ID0 is stored in TXMB3 and abort is set as TXMB0.

transmitting	ID0	TXMB0	←set abort reqesut
next transmission	ID1	TXMB1	
waiting for txscan	ID2	TXMB2	
Entry New ID →	ID0	TXMB3	
	TXMB4		
	TXMB5		



③ID1 is stored in TXMB4 and abort is set as TXMB1.

transmitting	ID0	TXMB0	←wait for abort
next transmission	ID1	TXMB1	←set abort reqesut
waiting for txscan	ID2	TXMB2	
	ID0	TXMB3	
Entry New ID →	ID1	TXMB4	
	TXMB5		



④ID2 is stored in TXMB5 and the transmit request of TXMB2 is cleared.

transmitting	ID0	TXMB0	←wait for abort
next transmission	ID1	TXMB1	←wait for abort
	TXMB2		←clear transmission request
	ID0	TXMB3	
Entry New ID →	ID1	TXMB4	
	TXMB5		



⑤Transmission of TXMB0 is completed and transmission of ID1 of TXMB1 is started.

completion of transmitting	TXMB0	
transmitting	ID1	TXMB1
	TXMB2	←wait for abort
	ID0	TXMB3
Entry New ID →	ID1	TXMB4
	ID2	TXMB5



When a system writes in TXQ, a system should write in send data, after checking the state of TXQ.

Users should also not access or configure the related TX Message Buffer Control Registers.

The messages stored to the TX Queue access window are internally stored to a free buffer of the TX Queue.

When the buffer is full then no further access should be done to the Queue, until it is no longer full. If it accesses by software writing in when the buffer of TXQ is full, send data will be overwritten.

The TX Queue can be disabled by clearing the TXQE bit in the TX Queue Configuration / Control Register. If this bit is cleared, the TX Queue empty flag is set as described below:

immediately if the message from the TX Queue is neither scheduled for the next transmission nor in transmission

following the transmission completion, the detection of an error on the CAN bus, loss of arbitration or transition to Channel or Global Halt Mode if the transmission from the TX Queue is already scheduled for

transmission or already in transmission

Note: The TX Queue is disabled only when the Empty flag is set after clearing the TXQE bit for the corresponding TX Queue.

Other possible messages pending from the TX Queue are lost and their transmission needs to be requested again.

Before TXQE is set again ensure that the **CFDTXQSTS_n.TXQEMP** bit is set and that there is no pending abort from the TX Queue.

When the TXQE bit is cleared all messages in the TX Queue buffers will be lost and no further message should be stored into the TX Queue.

When a message has been stored to the TX Queue, 8hFF must be written in to the TX Queue Pointer Control Register. This will set the transmit request automatically and change the internal Message Buffer pointer to the next free Message Buffer location of the TX Queue.

Note: If two messages with the same Identifier are stored in the TX Queue, then the order of transmission of these messages could be different from the order in which they were stored in the TX Queue.

To avoid this condition, it is important to confirm that the previous message with the same ID was successfully transmitted before a new Message with the same Identifier is stored in the TX Queue.

Or if TX queue overwrite mode is used, the frame of the same ID will be rewritten on a new frame.

For the TX Queue a dedicated interrupt can be enabled by setting the TXQIE bit of the TX Queue Configuration / Control Register.

The interrupt mode can be configured with the **CFDTXQCC_n.TXQIM** bit of the same register either to generate an interrupt for every transmitted message or for the last transmitted message.

The TX Queue transmission request procedure after configuration is shown in Figure 10.13.

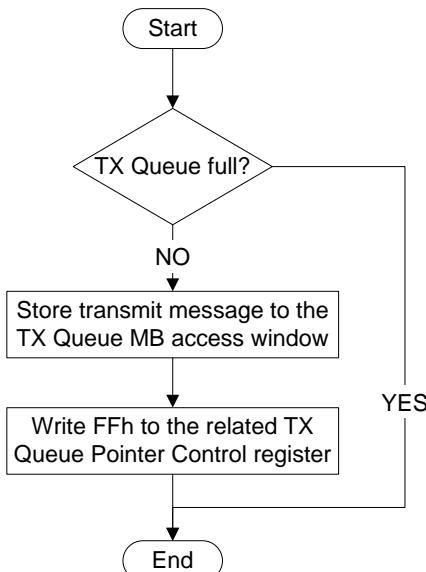


Figure 10.13 : TX Queue transmission request

TXQ Name	Access window	Range width	Direction	HW Routing access point	CPU access point	DMA access point	TX queue overwrite mode	Note
TXQ0	TXMB0	0..3-32	TXMB0 → TXMB31	Yes	Yes	Yes	CPU/DMA/Routing	When using both TXQ0/1, the number sum total of stages is 32 or less.
TXQ1	TXMB31	0..3-32	TXMB31 → TXMB0	Yes	Yes	No	CPU/Routing	
TXQ2	TXMB32	0..3-32	TXMB32 → TXMB63	Yes	Yes	No	CPU/Routing	When using both TXQ2/3, the number sum total of stages is 32 or less.
TXQ3	TXMB63	0..3-32	TXMB63 → TXMB32	No	Yes	Yes	CPU/DMA	

TXQ0 can use HW Routing, CPU access, and DMA access.

HW Routing access, CPU access, and DMA access should not be used simultaneously.

The one access method is chosen.

10.2.5 TX History List

The TX History List function records the information of the successfully transmitted message in the TX History List Buffers for each CAN channel. Two THL Buffer are provided for each CAN channel and each THL Buffer can store up to 32 THL entries for a CAN channel.

The **CFDTHLCCn.THLDTE** bit of the TX History List Configuration / Control Register can be used to configure if only message information from TX FIFOs / TX Queue is stored or if all transmit message information from TX Queue, TX FIFO or normal TX Message Buffers should be stored in the TX History List for a CAN channel.

When a **CFDTHLCCn.THLDGE** bit is set up, the information on all the frames transmitted in GW mode is stored in THL.

Each transmit message can be individually configured for acceptance to the TX History List by the **CFDCFID.THLEN** bit in the Message Buffer Pointer Register.

The message information is stored to the TX History List Buffer of a CAN channel after the message is successfully transmitted on that CAN channel.

Storing to the List is not synchronized with the status of **CFDTMSTSj.TMTRF[1:0]** bits in the TX Message Buffer Status Register.

Due to internal processing, the storage to the List could happen with a delay after the successful transmission indication.

Storing the TX History List data can be recognized by the condition that the THLIF is set to 1 when the THLIE is configured to 1 or when the TX History List counter **CFDTHLSTS_n.THLMC[5:0]** is increased.

The delay time is dependent on the number of channels due to internal processing.

The delay due to internal processing can be calculated as follows:

Maximum delay time from setting the **CFDTMSTSj.TMTRF** to storing the TX History List data is 224 peripheral bus clock cycles.

*Peripheral bus clock delay = Routing process time + (THL process time * No. of channels)*

*peripheral bus clock delay = (192 + (4 * (n+1)))*

The History list records following information of the transmitted message:

Buffer Type:

001: TX Message Buffer

010: TX FIFO

100: TX Queue

Buffer Number:

TX Message Buffer, TX Queue Message Buffer or TX Message Buffer Link for the Common FIFO Buffer from which the transmission occurred. The number depends upon the Buffer Type, refer to Table 10.3.

Transmission ID:

Transmission Pointer stored in the transmission message

Transmit Timestamp:

Message timestamp captured at capture point as configured by **CFDGFD CFG.TSCCFG**.

Transmission Information Label:

Transmission information label stored in the transmission message.

Transmit Gateway Buffer indication:

In the case of the data transmitted from Gateway, **CFDTHLACC0n.TGW** bit is set to 1.

BufferType BufferNumber	1	10	100
000000	Message Buffer 0		
000001	Message Buffer 1		
000010	Message Buffer 2		
000011	Message Buffer 3		
0000100	Message Buffer 4		
0000101	Message Buffer 5		
0000110	Message Buffer 6		
0000111	Message Buffer 7		
0001000	Message Buffer 8		
0001001	Message Buffer 9		
0001010	Message Buffer 10		
0001011	Message Buffer 11		
0001100	Message Buffer 12		
0001101	Message Buffer 13		
0001110	Message Buffer 14		
0001111	Message Buffer 15		
0010000	Message Buffer 16		
0010001	Message Buffer 17		
0010010	Message Buffer 18		
0010011	Message Buffer 19		
0010100	Message Buffer 20		
0010101	Message Buffer 21		
0010110	Message Buffer 22		
0010111	Message Buffer 23		
0011000	Message Buffer 24		
0011001	Message Buffer 25		
0011010	Message Buffer 26		
0011011	Message Buffer 27		
0011100	Message Buffer 28		
0011101	Message Buffer 29		
0011110	Message Buffer 30		
0011111	Message Buffer 31		
0100000	Message Buffer 32		
0100001	Message Buffer 33		
0100010	Message Buffer 34		
0100011	Message Buffer 35		
0100100	Message Buffer 36		
0100101	Message Buffer 37		
0100110	Message Buffer 38		
0100111	Message Buffer 39		
0101000	Message Buffer 40		
0101001	Message Buffer 41		
0101010	Message Buffer 42		
0101011	Message Buffer 43		
0101100	Message Buffer 44		
0101101	Message Buffer 45		
0101110	Message Buffer 46		
0101111	Message Buffer 47		
0110000	Message Buffer 48		
0110001	Message Buffer 49		
0110010	Message Buffer 50		
0110011	Message Buffer 51		
0110100	Message Buffer 52		
0110101	Message Buffer 53		
0110110	Message Buffer 54		
0110111	Message Buffer 55		
0111000	Message Buffer 56		
0111001	Message Buffer 57		
0111010	Message Buffer 58		
0111011	Message Buffer 59		
0111100	Message Buffer 60		
0111101	Message Buffer 61		
0111110	Message Buffer 62		
0111111	Message Buffer 63		
1000000	Message Buffer 64		
1000001	Message Buffer 65		
1000010	Message Buffer 66		
1000011	Message Buffer 67		
1000100	Message Buffer 68		
1000101	Message Buffer 69		
1000110	Message Buffer 70		
1000111	Message Buffer 71		
1001000	Message Buffer 72		
1001001	Message Buffer 73		
1001010	Message Buffer 74		
1001011	Message Buffer 75		
1001100	Message Buffer 76		
1001101	Message Buffer 77		
1001110	Message Buffer 78		
1001111	Message Buffer 79		
1010000	Message Buffer 80		
1010001	Message Buffer 81		
1010010	Message Buffer 82		
1010011	Message Buffer 83		
1010100	Message Buffer 84		
1010101	Message Buffer 85		
1010110	Message Buffer 86		
1010111	Message Buffer 87		
1011000	Message Buffer 88		
1011001	Message Buffer 89		
1011010	Message Buffer 90		
1011011	Message Buffer 91		
1011100	Message Buffer 92		
1011101	Message Buffer 93		
1011110	Message Buffer 94		
1011111	Message Buffer 95		

Table 10.3 TX History List Buffer Number entry

The Transmission ID entry is used to identify which message of a TX FIFO or TX Queue has been successfully transmitted because the TX FIFO or TX Queue number alone is not sufficient.

Therefore, a unique number can be attached to each transmission message stored in a TX FIFO or TX Queue. This unique identification number should be written to the **CFDCFFDCSTS.CFPTR[15:0]** part of the Common FIFO Access Pointer Register for a TX FIFO or to the **CFDTMFCTR.TMPTR[15:0]** part of the TX Message Buffer Pointer Register of the TX Queue access window Message Buffer.

When the message is successfully transmitted then this identification number is stored together with the other message related information to the TX History List and can be read via the Transmission ID (TID) of the TX History List Access Register.

Also for normal TX Message Buffers, the **CFDTMFCTR.TMPTR[15:0]** part of the TX Message Buffer Pointer Register will be stored in the Transmission History List. Information label is the same.

Figure 10.14 shows a transmission preparation flow when TX History List is used.

Read access to the TX History List Access Register will be done for every single entry.

After reading one entry, 8'hFF has to be written to the corresponding TX History List Pointer Control Register to be able to access the next entry until TX History List is empty.

Figure 10.15 shows an example flow for processing the TX History List information.

The TX History Lists have dedicated interrupts, which can be configured with the **CFDTHLCCn.THLIM** bit of the corresponding TX History List Configuration / Control Registers and enabled with the

CFDTHLCCn.THLIE bit of the same registers, either to generate an interrupt when the History List reached a filling level of 75% or for every new TX History List entry.

An entry lost indication is flagged by the **CFDTHLSTS_n.THLELT** bit in the TX History List Status Register.

Status of this bit is also shown by the THLES bit in the Global Error Flag Register.

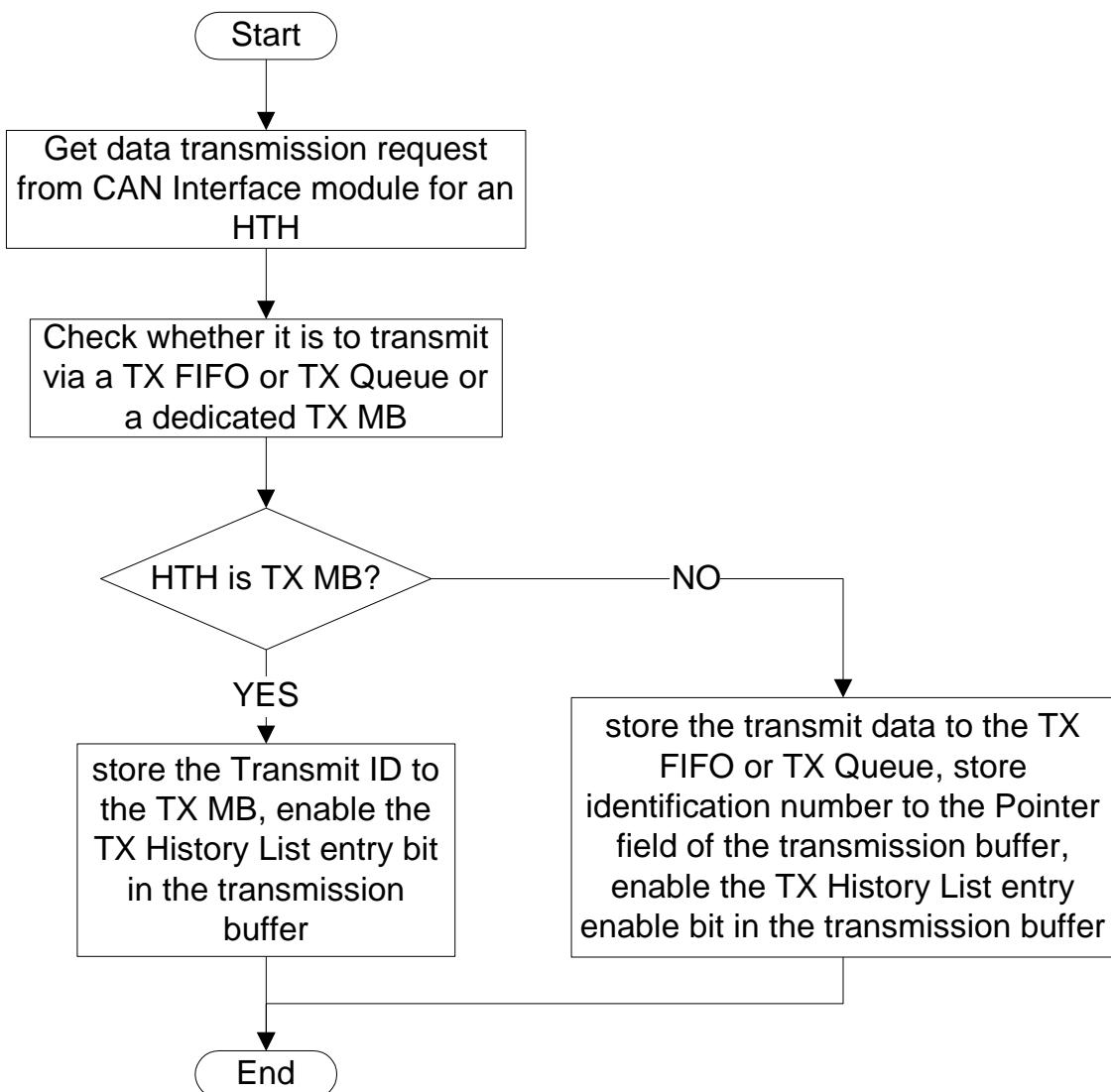


Figure 10.14 TX History List preparation flow

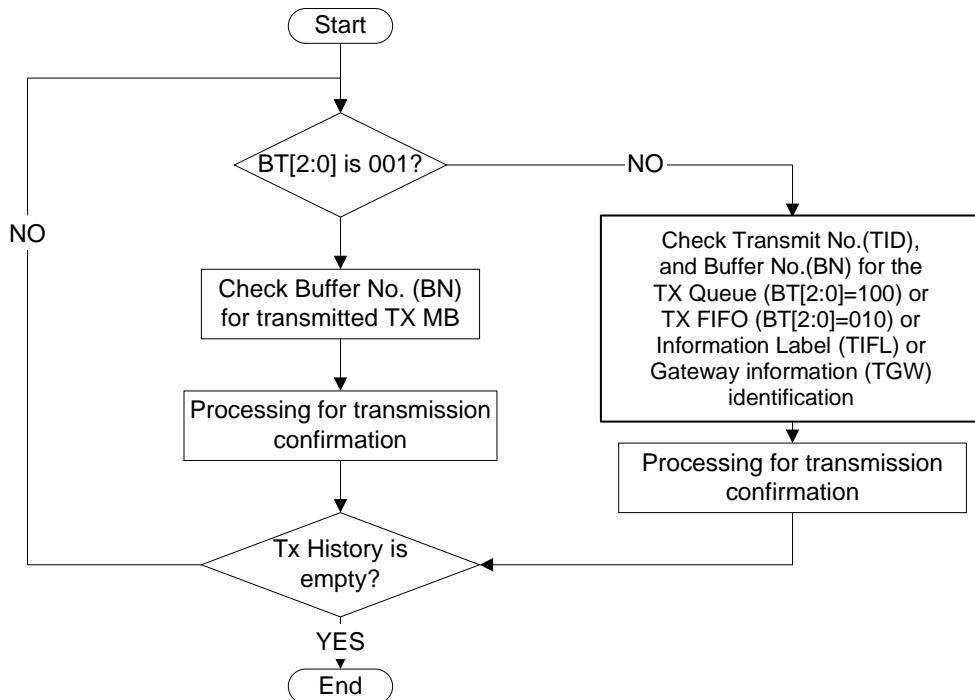


Figure 10.15 TX History List processing flow

10.2.6 TX data padding

If the data length code (DLC) of the transmitting message is higher number of data bytes than the buffer size. Then the data bytes beyond the restricted range will be replaced by bytes with the value of CC HEX.

This could happen for Common FIFOs configured as (TX or GW) when the transmit message DLC is higher than the **CFDCFCCd.CFPLS**.

This could also happens in FD only mode, if a Classical Frame is configured with a DLC bigger than 8.

11 ECC Check

An ECC check mechanism is implemented in RS-CAN-FD to check consistency of data in the RAM.

Each memory location in the RAM consists of 39 bits. 32 bits are used for storage of user data and 7 bits are used for storage of ECC data. For each long-word (32 bits) memory location, ECC data is allocated. The value of this ECC data is calculated by the ECC generation logic.

The ECC check mechanism checks the data during read access from CAN side or CPU side to the RAM. If a single bit error is detected, then it is corrected and a flag is set in the ECC Status Registers (Register is outside of CAN IP).

If multiple bit error is detected from CAN side read during the transmission scan, then the transmit request of the CAN channel which is under transmission scan will be suppressed and the corresponding **CFDGERFL.EEFn** bit is set. This is done to avoid transmission of corrupted data.

In general if multiple bit errors are detected then perform RAM test to check the RAM status. When no problem is found, make a transition to global reset mode and then make communication initialisation again.

Note: If an error is detected by the ECC logic then the error address of the RAM access will be captured in the capture register which is part of the ECC macro (not RS-CAN-FD).

The captured RAM address will not match with the SFR address.

Example for 8 channel version:

If ECC error occurs when CPU is reading Identifier of RX (**CFDRMID**), then the SFR address of **CFDRMID3** is 2180h. However, the RAM Address for the same is A560h (if **CFDRMN.B.RMPLS=3d**), refer to Section 13.

Based on the RAM address given by the ECC module when an ECC error occurred the related buffer can be determined with the information given in Section 13. As example the configuration given at Figure 13.2 is considered.

Since the data of each RAM is 32 bits, a RAM address is an address of 4 byte boundaries. Therefore, as for the address of an ECC error, the address of 4 byte boundaries is shown.

When the ECC module returns the address 2A38h (10808d) as address where the ECC error occurred, then the error occurred at COM FIFO 0.

In order to access the MRAM address in RAM test mode the related page number (pn) must be calculated.

$$Pn = \text{floor}(\text{address}/(256/4)) = \text{floor}(10808/(256/4)) = 168$$

The page number is written to the **CFDGTSTCFG.RTMPS[9:0]** register. Because in RAM test mode AFL RAM and MRAM are considered together, 96 pages must be added to the calculated one (refer to section 12.2.1). Therefore 264 must be written.

The address to be accessed at the RAM test pages **CFDRPGACCk** is calculated as follows.

$$k = \text{mod}(\text{address}/(256/4)) = \text{mod}(10808/(256/4)) = 56$$

Acceptance filter list RAM consists of two RAM. The relation between an AFL entry related register, a RAM address, and an AFL page number is shown.

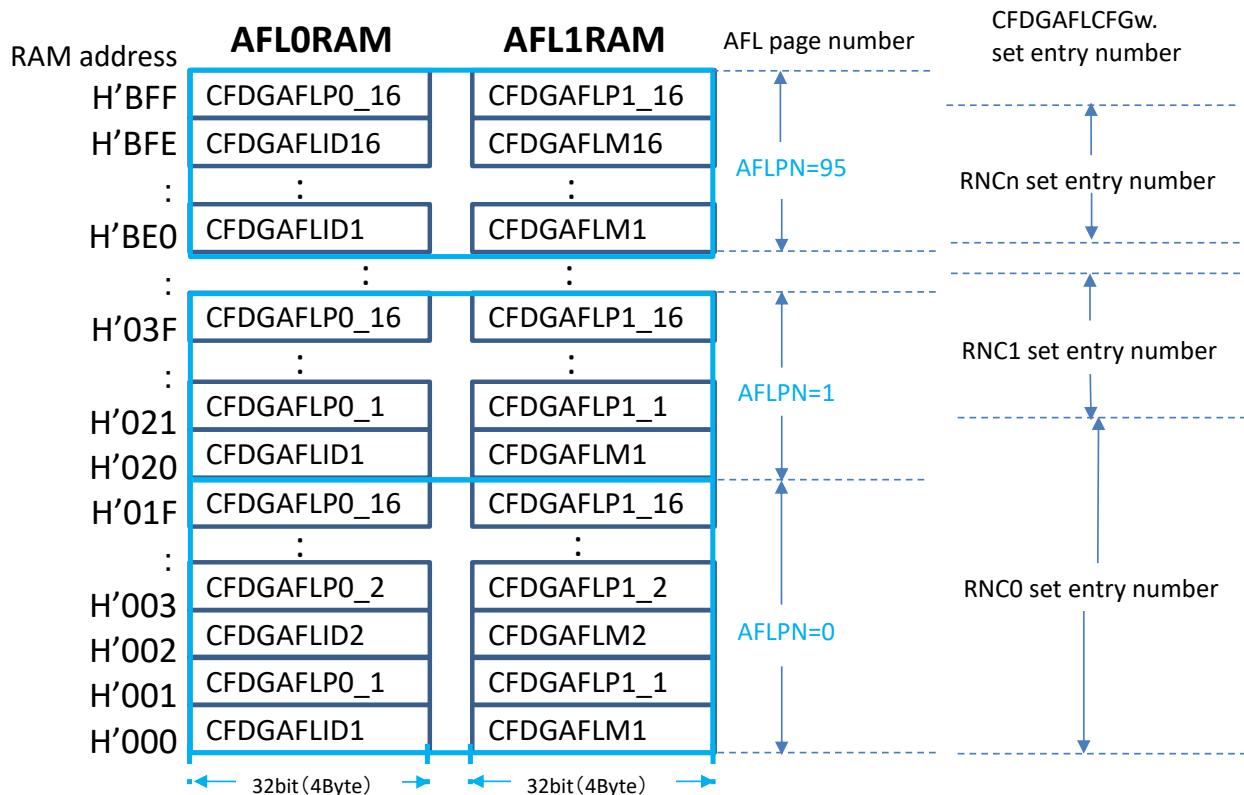


Figure 11.1: Relation of AFLRAM and a register

The relation between Register **CFDRPGACC_k** ($k = 0$ to 63), and AFL entry page number and the number of AFL entry is shown.

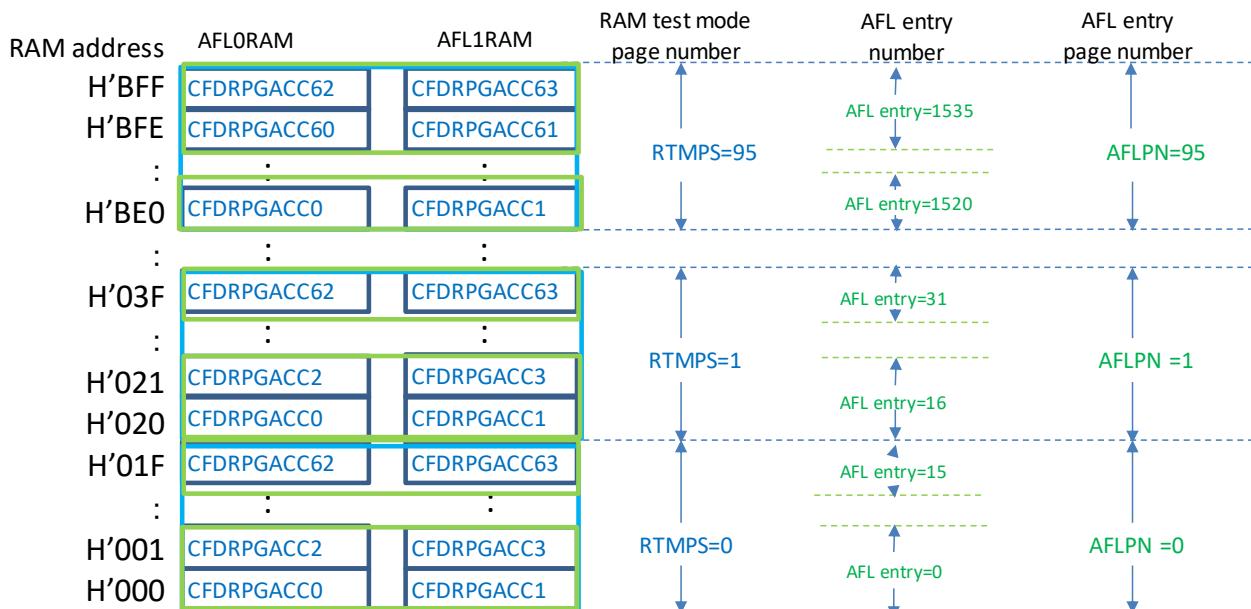


Figure 11.2: Relation of AFLRAM and a RAM test mode

The access sequence of AFLRAM of a RAM test mode accesses AFL0RAM and AFL1RAM by turns. MBRAM area is accessed after accessing the final address of AFL1RAM.

```

H' 000 AFL0RAM→ H' 000 AFL1RAM→
H' 001 AFL0RAM→ H' 001 AFL1RAM→
:
H' BFF AFL0RAM→ H' BFF AFL1RAM→
H' 0000 MBRAM→
:
H' C13E MBRAM→ H' C13F MBRAM

```

The access sequence in a RAM test mode is shown.

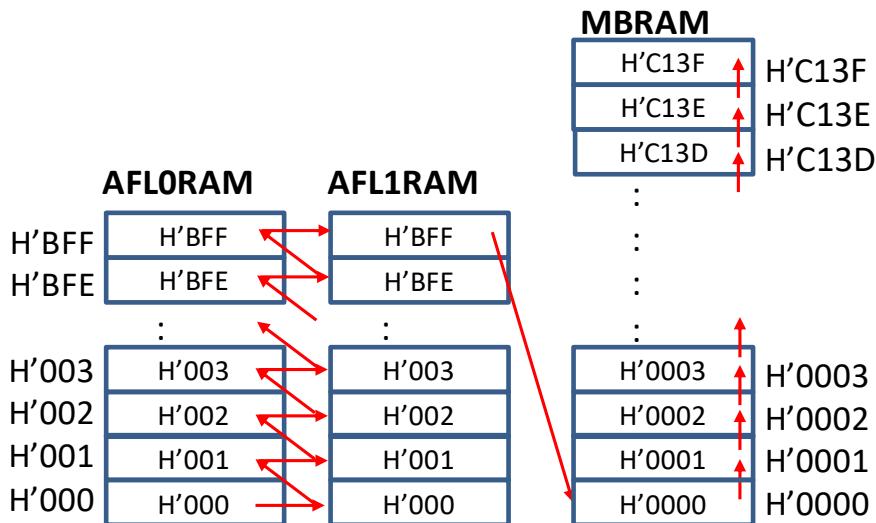


Figure 11.3: RAM test mode access sequence

The case which the ECC error generated in H'BFE of AFL1RAM explains the access method of a RAM test mode, and the calculation method of the number of pages. In a RAM test mode, page settings are performed to **CFDGTSTCFG.RTMPS**. RAM is accessed from **CFDRPGACC0~CFDRPGACC63** register. By the following sequences, the number of pages calculates and it asks for an access register.

1. Calculation of the number of pages
In a RAM test mode, in order to access AFL0RAM/AFL1RAM by turns, it doubles H'BFE of an ECC error address. The error case of AFL1RAM adds 1 to the result.
It asks for the number of pages in the following calculation formulae from the result.
$$(H'BFE*2+1)/(256/4)=H'17FD/64=95$$

95 is set to **CFDGTSTCFG.RTMPS**.
2. Calculation of the position in a page asks for remainder from a previous formula.
$$\text{Mod}=(H'BFE*2+1)/(256/4)=H'17FD/64=61$$
3. If 95 is set to **CFDGTSTCFG.RTMPS** and **CFDRPGACC61** is accessed, H'BFE of AFL1RAM will be accessed.

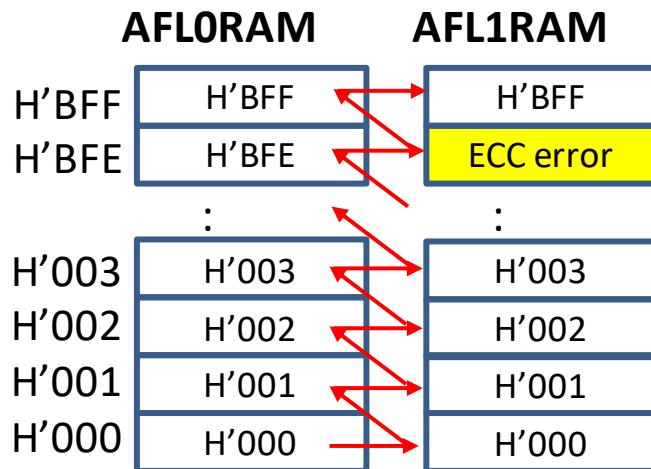


Figure 11.4: RAM error part access

The relation between a PFL entry related register, RAM address, and PFL page number is shown below.

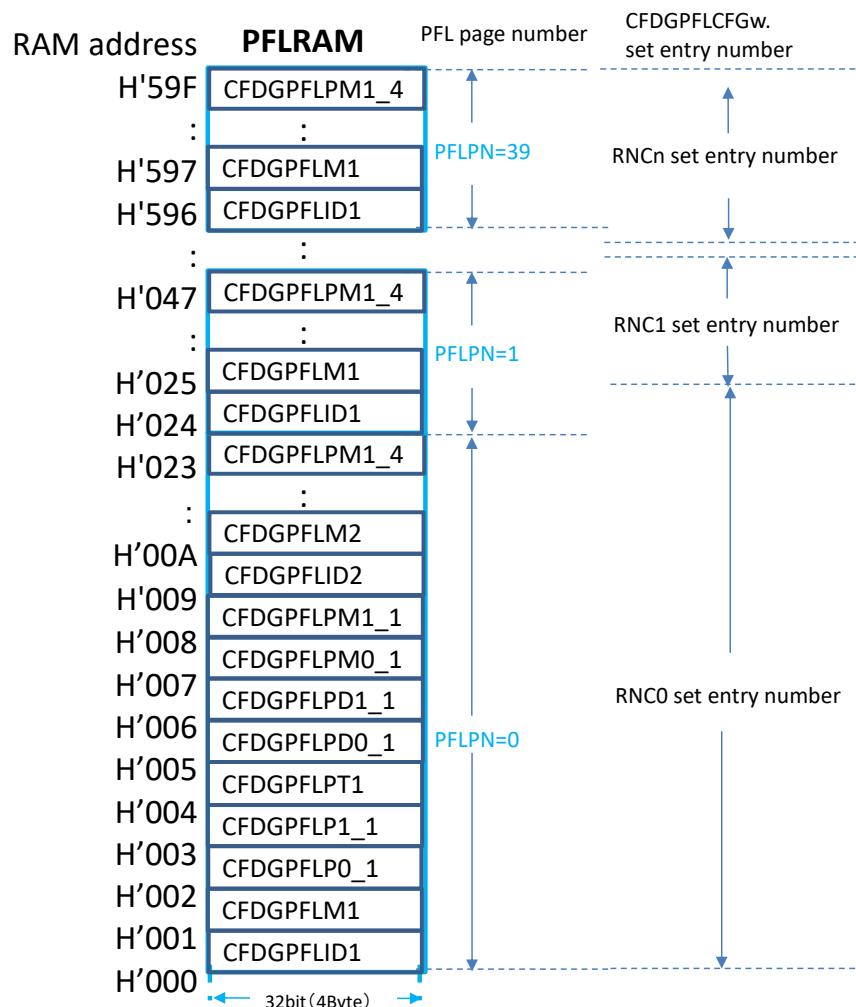


Figure 11.5: Relation of PFLRAM and a register

The relation between Register CFDRPGACC k ($k = 0$ to 63), and PFL entry page number and the number of PNF entries is shown.

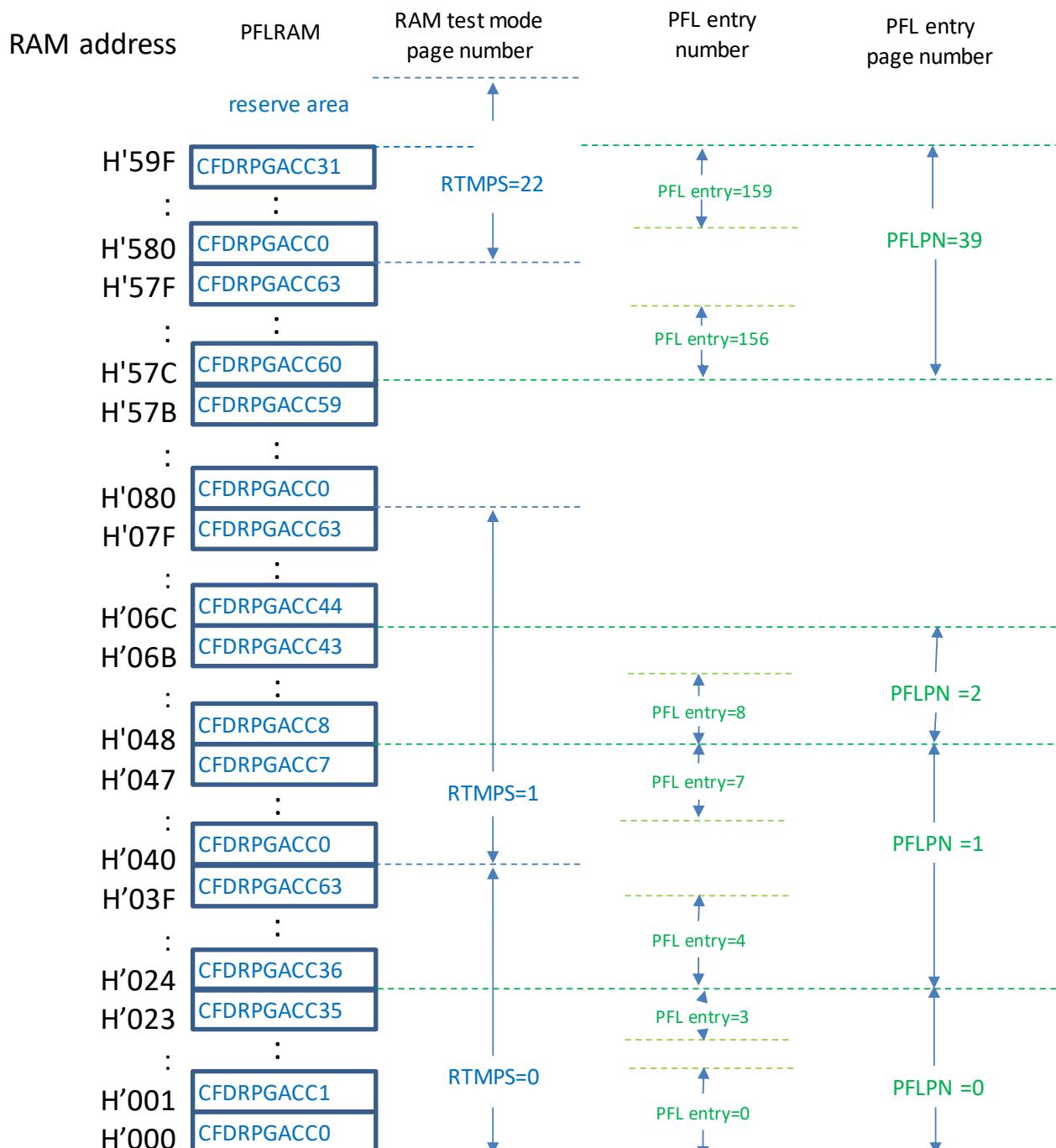


Figure 11.6: Relation of PFLRAM and a RAM test mode

12 Test Mode

The RS-CAN-FD module can be configured into Test Modes to allow testing of certain features. These features are provided only for special purposes and care must be taken when configuring the RS-CAN-FD module in the test modes.

Note that all Test modes are mutually exclusive unless it is explicitly stated that some functions can be enabled across other test modes.

Users should not enable any combinations of the various Test modes specified in this Section.

The Test modes can be broadly split into 2 groups:

Channel specific test modes
Global test modes

12.1 Channel specific test modes

Each CAN channel can be configured into following test modes:

Basic test mode
Listen-only mode
Self test mode 0 (External Loop back mode)
Self test mode 1 (Internal Loop back mode)
Restricted Operation Mode

Additionally the CFDCnCTR.TRWE, CFDCnCTR.TRH and CFDCnCTR.TRR bits can be configured in the corresponding Channel Control Registers in any channel test mode.

12.1.1 Basic test mode

The Basic test mode should be used when a particular test setting needs to be enabled other than when in Listen-Only and Self-test modes.

12.1.2 Listen-only mode

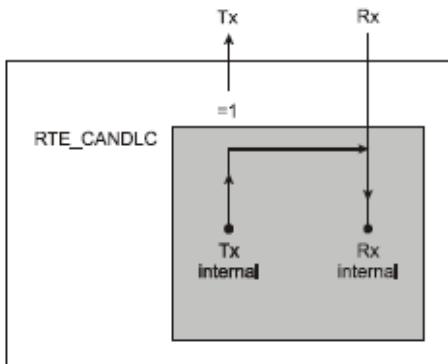
The ISO 11898-1 recommends an optional bus-monitoring mode. In this mode, the CAN channel is able to receive valid data frames and valid remote frames. However, it sends only recessive bits on the CAN bus and is not allowed to transmit.

If the CAN engine is required to send a dominant bit (ACK bit, overload flag, active error flag), the bit is routed internally so that the CAN engine monitors this as dominant. The external TX pin will remain in recessive state.

This mode can be used for Baud Rate detection. In this mode, an error interrupt is generated if a bus error occurs and the interrupt is enabled.

In this mode, it is not permitted to request transmission from any normal TX Message Buffer or TX- / GW-FIFO of this channel.

Note: If a message is stored in GW FIFO or Routing TXQ then users should ensure that the transmitting channel is not in Listen-only mode so that transmission is not requested for this channel from the GW FIFO or Routing TXQ.



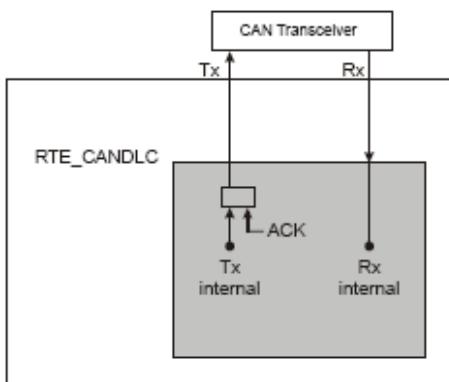
12.1.3 Self test mode 0 (External Loop back mode)

In Self Test Mode 0, the CAN engine treats its own transmitted messages as received messages via the CAN transceiver and can store them into its receive Message Buffers.

To be independent from external stimulation the engine generates its own Acknowledge bit.

This test can be used for CAN transceiver tests.

The Rx/TX pins should be connected to the transceiver.



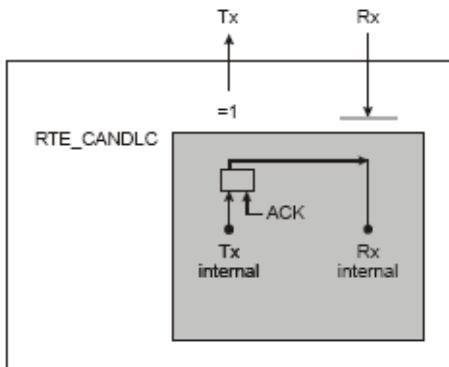
12.1.4 Self test mode 1 (Internal Loop back mode)

In Self Test Mode 1, the CAN engine treats its own transmitted messages as received messages and stores them into the receive buffer. This mode is provided for self-test functions. To be independent from external stimulation the CAN engine generates its own Acknowledge bit. In this mode the CAN engine performs an internal feedback from TX internal to Rx internal. The actual value of the external Rx input is disregarded by the CAN engine.

The external TX pin outputs only recessive bits.

The Rx/TX pins do not need to be connected to the CAN bus or any external device.

Note: The channel Pins are also disconnected from the Internal CAN Bus Communication line.



12.1.5 Restricted Operation Mode

In Restricted Operation Mode the CAN node is able to receive valid data and remote frames generating the Acknowledge bit.

Active Error and Overload frames cannot be transmitted instead it waits for the occurrence of bus idle condition to resynchronise itself to the CAN communication after an error or overload condition occurs

Moreover the Receive and Transmit Error Counter (REC and TEC) are frozen independently from the occurrence of errors.

The mode is specified as in ISO 11898-1; however it is permitted to set any transmit requested.

12.2 Global test modes

The RS-CAN-FD module can be configured into following test modes:

RAM Test Mode

Internal CAN Bus Communication Mode

CRC Error Test

OTB Test Mode

For following test modes are protected by a special SW procedure to enable the mode. This SW procedure enables the write access to the test mode by specific unlock Key, the related unlock key can be seen in the table below:

Test Mode	Unlock key 1	Unlock key 2
RAM Test Mode	7575h	8A8Ah
OTB FIFO Disable	CCCCh	3333h

If the SW sequence of the two consecutive unlock key write accesses (word or long-word accesses) is interrupted by any other write access to the SFR or if incorrect data is written to the Global Unlock Key Register then the corresponding Test mode cannot be set and the sequence should be re-started.

After the two unlock key write accesses, the next write access should be to set the corresponding Testmode Enable bit. If this is not followed, the unlock mechanism resets and the Testmode enable bit cannot be set and then the unlock sequence should be restarted.

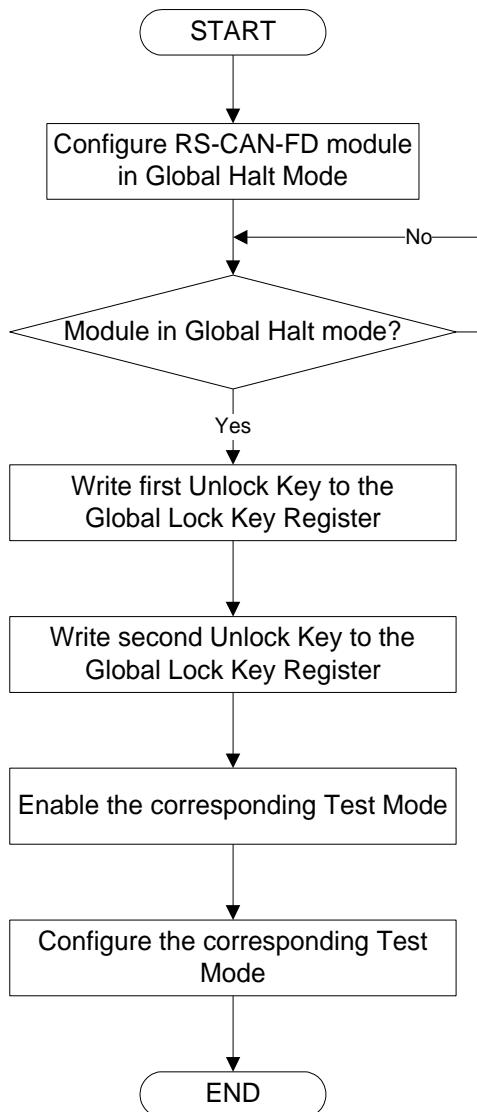


Figure 12.1: Unlock SW protection routine

12.2.1 RAM Test Mode

The RS-CAN-FD module can be configured in RAM Test Mode by setting the **CFDGTSTCTR.RTME** bit in the Global Test Control Register when the corresponding lock key is written before. This is a special test mode, in which, the complete RAM area can be accessed.

Note: The actual RAM size is bigger than the RAM area initialised after HW Reset. Hence, ECC error flag (of the ECC macro) may be set if CPU reads data from this un-initialised RAM area while RS-CAN-FD module is in RAM Test Mode.

In this mode, the RAM area is split into number of pages (**pn**) of 256 Bytes each. Which can be accessed via **CFDRPGACCK** register.

The page should be selected for read / write access by writing to the **CFDGTSTCFG.RTMPS[9:0]** bits in the Global Test Control Register. Then, data can be read from or written in to the RAM Test Page Access Registers.

Figure 12.2 shows the structure of the pages in the RAM when performing a RAM Test Mode.

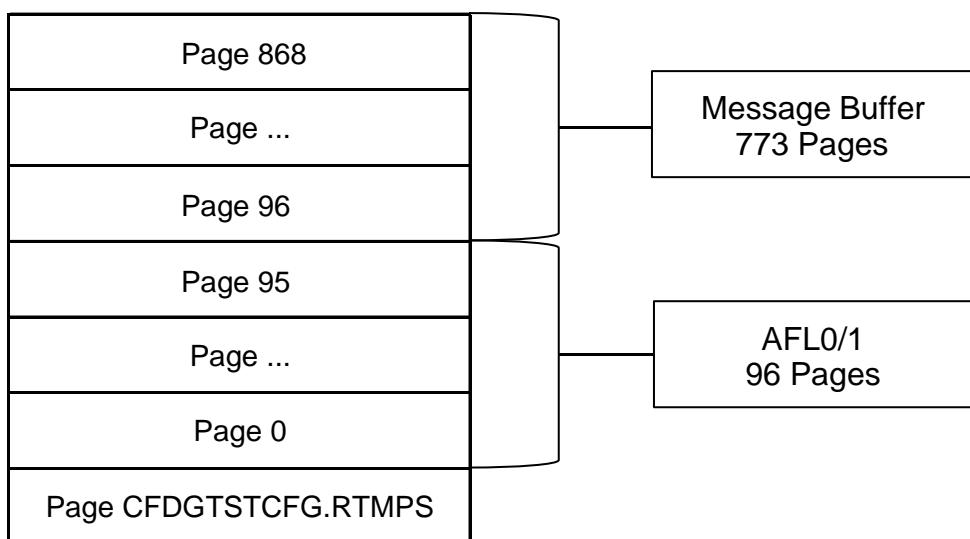


Figure 12.2: RAM page structure

The total available RAM size for a 8 CAN channel version is, 24576 Bytes for the AFL RAM and 197888 Bytes for the Message Buffer RAM.

AFL RAM0/1 can treat RAM Test mode as one RAM.

The RAM size is calculated according to the following formula:

AFL RAM: 16 Bytes * 192 * number of channel

MB RAM : 24736 Bytes * number of channel

The **pn** and **CFDGTSTCFG.RTMPS[9:0]** values for the AFL and MB RAMs are calculated in the following way:

pn = ceil(Total RAM size in Bytes / Number of Bytes per page)

AFL RAM:

pn = ceil(24576 / 256) = 96 Pages

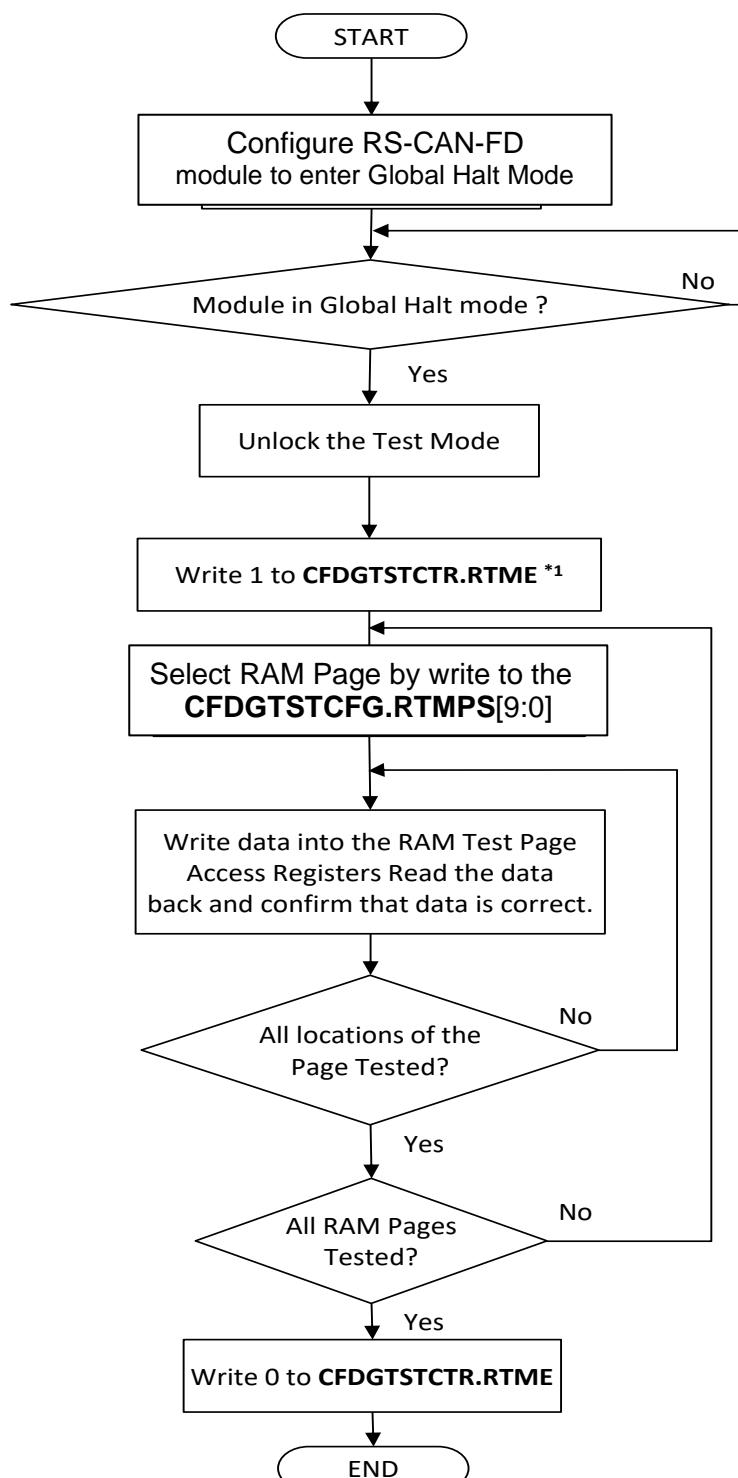
CFDGTSTCFG.RTMPS[9:0] = 0 to 95 (10'h5F) inclusive

MB RAM:

pn = ceil(197888 / 256) = 773 Pages

CFDGTSTCFG.RTMPS[9:0] = 96 to 868 (10'h364) inclusive

Figure 12.3 below shows the SW flow for RAM Test mode.



*1 Change into the following status before changing to RAMTEST.
Cancel of a request of transmission.
Disable of all the FIFO and TXQ.
Clear of the receiving flag of a receiving buffer.

Figure 12.3: RAM Test Mode SW flow

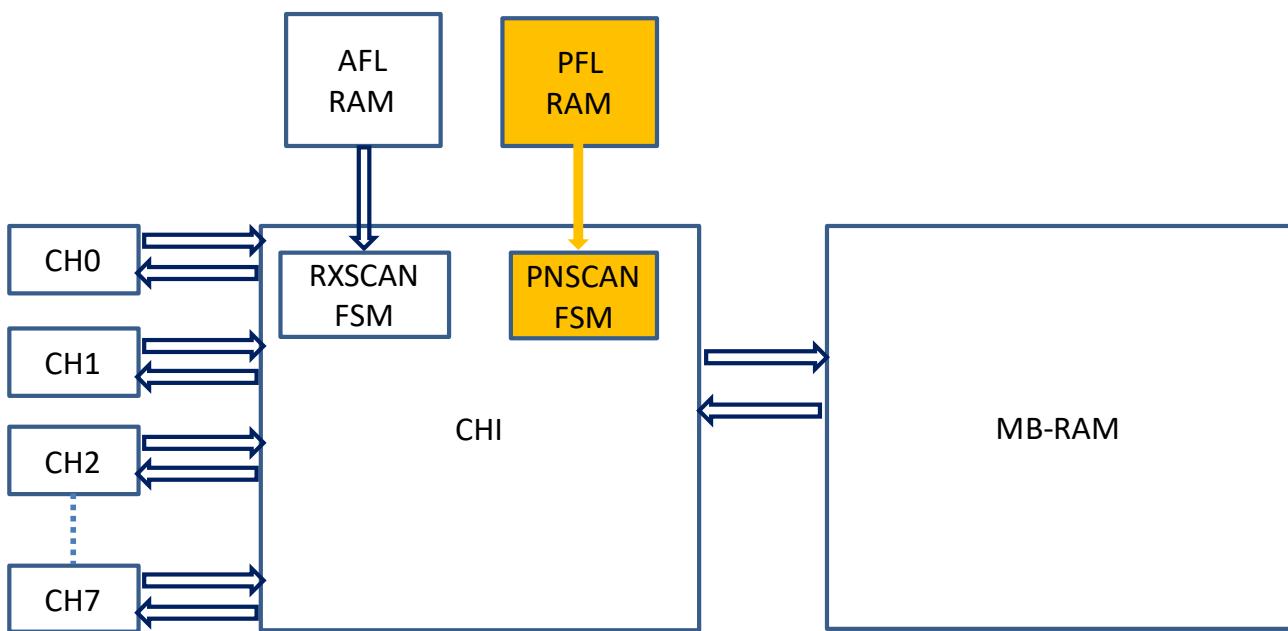
To exit this Test mode, the **CFDGTSTCTR.RTME** bit must be cleared. The **CFDGTSTCTR.RTME** bit is cleared by writing 1'b0 to it.

The **CFDGTSTCTR.RTME** bit is cleared automatically when the RS-CAN-FD module enters Global Reset mode from the Test mode.

12.2.1.1 RAM Test Mode for PFLRAM

The test mode of RAM for PN filters is explained.

As for PFLRAM, a part of configuration of a test mode differs between AFLRAM and MRAM.



The RAM test of RAM for PN becomes the following procedures.

1. Transition to Global_HALT mode.
2. PFLRAM selection bit is set. (**CFDGTSTCFG.PNFS** = 1'b1) *1
3. Transition to a RAM test mode. (**CFDGTSTCTR.RTME** = 1'b1)
4. RAM test page is set to **CFDGTSTCFG.RTMPS**.
5. RAM for PN is accessed. *2

*1: AFLRAM and MRAM cannot be accessed when a PFLRAM selection bit is valid.

*2: Refer to Section 13.3 for the RAM test page number which can be accessed.

12.2.2 Internal CAN Bus Communication Test Mode

The RS-CAN-FD module can be configured in Internal CAN Bus Communication Test Mode by setting the **CFDGTSTCTR.ICBCTME** bit in the Global Test Control Register. This is a special test mode, in which the CAN channels can be connected together internally to generate a CAN cluster within the RS-CAN-FD module.

Users should only use following sequence to enter Internal CAN Bus Communication Test Mode:

1. Configure all channels in Halt Mode and check that all channels have entered Halt mode (Global Halt mode).
2. Write data into the Global Test Configuration Register to select the channels participating in the Internal CAN Bus Communication Test.

3. Set the **CFDGTSTCTR.ICBCTME** bit of the Global Test Control Register.
4. Check that **CFDGTSTCTR.ICBCTME** bit is set in the Global Test Control Register.

In this mode, the TxD outputs of the channels participating (configured) in Internal CAN Bus Communication Mode are connected together using AND gate. The output of the AND gate is connected to the RxD inputs of all participating channels to create a CAN cluster within the RS-CAN-FD module. The channels are isolated from the external CAN bus while the RS-CAN-FD module is in this test mode.

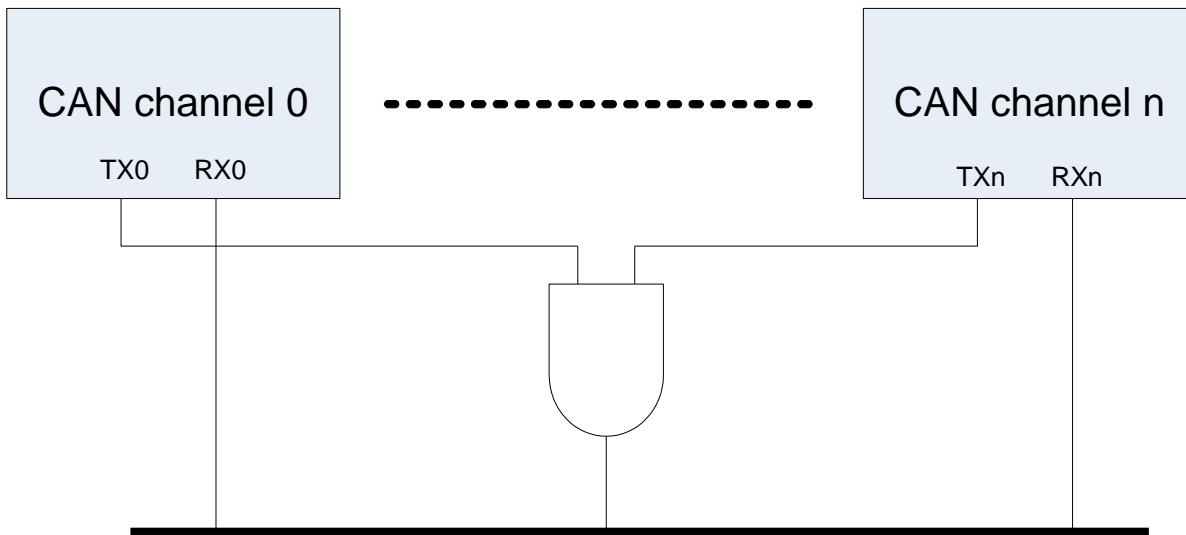


Figure 12.4: Internal CAN Bus connections

The AFL, Flat RX Message Buffers, FIFO Buffers, Flat TX Message Buffers and various registers can now be configured as normal to start communication between channels.

The channels not participating in Internal CAN Bus should only be configured in Halt Mode.

12.2.2.1 CRC Error Test

After the RS-CAN-FD module has been configured in Internal CAN Bus Communication Test Mode, the following sequence should be used to perform CRC Error testing. In the sequence below channel x is the reference transmitter RS-CAN-FD module and channel y is the receiver RS-CAN-FD module where ($x,y = [0...n]$ and $x \neq y$):

1. Configure channel x node to transmit 1 reference message
2. Set the **CFDCyCTR.CRCT** bit to 1'b1, in order to invert the first bit of the incoming bit stream from channel x
3. Set the **CFDTMCx.TMTR**
4. Wait for the **can_cherr_int[y]** output signal to set to 1'b1
5. Read either the **CFDCyERFL.CRCREG** or the **CFDCyCRC.CRCREG** (depending on the received frame type: Classical or FD). The value should be different from the received CRC value of the reference message from channel x.
6. Check that **CFDCyERFL.CERR** is 1'b1

As the CRC generator logic is shared for RX and TX there is no need to create a separate TX CRC Error test.

12.2.3 OTB Test Mode

*The RS-CAN-FD module can be configured in OTB Test Mode for disabling the OTB FIFO buffer. If the **CFDGLOTB.OTBFE** bit is cleared, then all data stored in OTB FIFO Buffer will be lost and the pointers will be cleared.*

Users should only use following sequence to enter the OTB Test Mode:

1. *Configure all channels in Halt Mode and check that all channels have entered Halt mode (Global Halt mode).*
2. *Write unlock key for OTB Test Mode as shown in Section 12.2.*
3. *Clear the **CFDGLOTB.OTBFE** bit of the Global OTB FIFO Configuration / Status Register.*
4. *Check that the **CFDGLOTB.OTBFE** bit is cleared in the Global OTB FIFO Configuration / Status Register.*
5. *Check that the **CFDGLOTB.OTBEMP** flag is set in the Global OTB FIFO Configuration / Status Register.*

*Note: The OTB contains data that will be stored in the various RX Message Buffers or FIFO Buffers and hence this test mode must be handled very carefully. Clearing the **CFDGLOTB.OTBFE** may result in loss of Data stored from received frames from various channels.*

*To exit this Test mode, the **CFDGLOTB.OTBFE** bit must be set. The **CFDGLOTB.OTBFE** bit is set by writing 1'b1 to it. The **CFDGLOTB.OTBFE** bit is set automatically when the RS-CAN-FD module enters Global Reset mode from the Test mode.*

13 RAM area configuration

The RAM area used in RS-CAN-FD can be split into the following groups as shown below in Figure 13.1:

Rule Table area

RX MB + FIFO Buffer area

TX MB area

THL area

OTB area (used for internal purpose)

PFL Rule Table area

Physically the RAM is split in two RAMs the AFL (Rule Table) RAM0 (**CFDGAFLID** and **CFDGAFLP0**), AFL (Rule Table) RAM1 (**CFDGAFLM** and **CFDGAFLP1**), the Message Buffer RAM (RX, RX FIFO, Common FIFO, TX, THL and OTB), and PFL (Rule Table) RAM.

The size of AFL RAM, MRAM and PFLRAM changes with the number of channels.

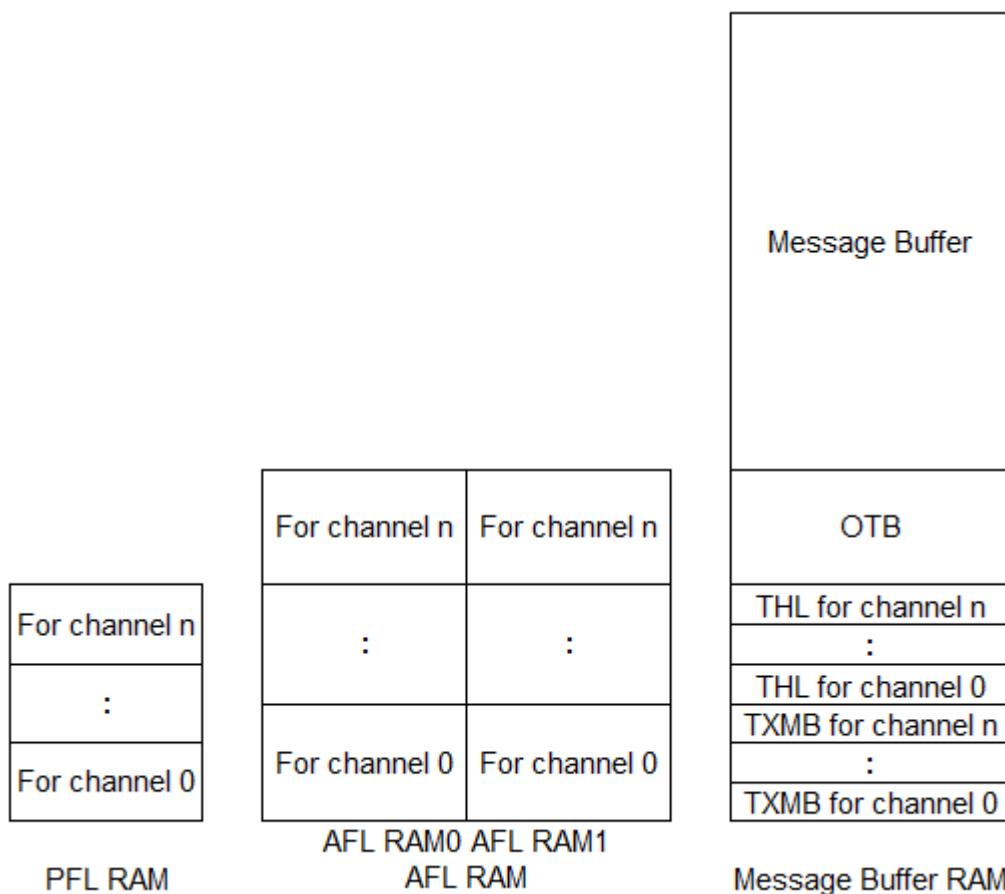


Figure 13.1: RAM area grouping

The Rule Table area always starts at AFL RAM address 0000h and has a fixed size for a given number of channels.

The MRAM area starts with the TX-MB area at address 0x0000. The TX-MB area is followed immediately by the THL area which is then followed immediately by the OTB area. The size of the TX-MB, THL and OTB area is fixed for a given number of channels.

The OTB area is followed by the message buffer area. The message buffer area size depends on the configuration of the flat RX MBs, RXFIFOs and CFIFOs. When all are configured the RX-MB area is

followed by the RXFIFO area which is followed by the CFIFO area.

The configured MRAM area can be calculated then as follows.

$$\text{MRAM_cfg} = \text{RXMB_MRAM_cfg} + \text{RXFIFO_MRAM_cfg} + \text{CFIFO_MRAM_cfg} + \\ \text{TXMB_MRAM_cfg} + \text{THL_MRAM_cfg} + \text{OTB_MRAM_cfg}$$

$$\text{RXMB_MRAM_cfg} = (12 \text{ Bytes} + \text{CFDRMND.RMPLS}) * \text{CFDRMNBB.NRXMB}$$

$$\text{RXFIFO_MRAM_cfg} = \text{SUM}((12 \text{ Bytes} + \text{CFDRFCCa.RFPLS}) * \text{CFDRFCCa.RFDC})$$

$$\text{CFIFO_MRAM_cfg} = \text{SUM}((12 \text{ Bytes} + \text{CFDCFCCd.CFPLS}) * \text{CFDCFCCd.CFDC})$$

$$\text{TXMB_MRAM_cfg} = 4864 \text{ Bytes} * n$$

$$\text{THL_MRAM_cfg} = 256 \text{ Bytes} * n$$

$$\text{OTB_MRAM_cfg} = 160 \text{ Bytes} * n$$

"a" means RX FIFO index = [0...no_of_RFIFOs-1]

"d" means Common FIFO index = [0 .. no_of_CFIFOs -1]

no_of_RFIFOs : Number of configured RX FIFOs

no_of_CFIFOs: Number of configured CFIFOs

Note: For **CFDRFCCa.RFDC**, **CFDCFCCd.CFDC**, **CFDRMNBB.RMPLS**, **CFDRMNBB.NRXMB**, **CFDRFCCa.RFPLS** and **CFDCFCCd.CFPLS** the related number of bytes must be used.

The PFL Rule Table area always starts at PFL RAM address 0000h and has a fixed size for a given number of channels.

The Table 13.1 shows the calculation of the different RAM areas used for the AFL entries, OTB buffers, TX/RX message buffers, RX/Common FIFOs and PFL entries.

RAM Name	RAM Property	RAM Area Calculation Method	AFL RAM Values											
	No. of Channels	(n+1)	8 6 4 2											
AFL	Avg. rule entries per channel	Parameter (192/ch, 128/ch, 64/ch)	192	128	64	192	128	64	192	128	64	192	128	64
	Max Rule entries	(n+1) * Avg. rule entries per channel	1536	1024	512	1152	768	384	768	512	256	384	256	128
	No. of Bytes in a rule entry	Fixed	16	16	16	16	16	16	16	16	16	16	16	16
	Number of Bytes AFL RAM	Max Rule entries * No. of Bytes in a rule entry	24576	16384	8192	18432	12288	6144	12288	8192	4096	6144	4096	2048

RAM Name	RAM Property	RAM Area Calculation Method	PFL RAM Values			
	No. of Channels	(n+1)	8 6 4 2			
PFL	Avg. rule entries per channel	Fixed	20			
	Max Rule entries	(n+1) * Avg. rule entries per channel	160 120 80 40			
	No. of Bytes in a rule entry	Fixed	36			
	Number of Bytes PNF RAM	Max Rule entries * No. of Bytes in a rule entry	5760	4320	2880	1440

RAM Name	RAM Property	RAM Area Calculation Method	MRAM Values																					
	No. of Channels	(n+1)	8																					
TX MB	No. of TX MBs per channel	Parameter (64/ch, 32/ch, 16/ch)	64		32		16																	
	Max no. of TX MBs	(n+1) * No. of TX MBs per channel	512		256		128																	
	No. of Bytes needed for each TX MB	Fixed	76		76		76																	
	Number of Bytes in TX MB area	(n+1) * No. of TX MBs per channel * No. of Bytes needed for each TX MB	38912		19456		9728																	
THL	No. of entries in 1 THL buffer	Fixed	32																					
	Max no. of THL entries	(n+1) * No. of entries in 1 THL buffer	256																					
	No. of Bytes needed for each THL entry	Fixed	8																					
	Number of Bytes in THL area	Max no. of THL entries * No. of Bytes needed for each THL entry	2048																					
OTB	Avg. number of buffers for each channel		2																					
	Max no. of OTB entries	(n+1) * Avg. number of buffers for each channel	16																					
	No. of Bytes for OTB entry	Fixed	80																					
	Number of Bytes in OTB area	Max no. of OTB entries * No. of Bytes for OTB entry	1280																					
Message Buffer	No. of RX MBs per channel	Fixed	16																					
	Max no. of RX MBs	(n+1) * No. of RXMBs per channel	128																					
	No. of RX FIFOs	Fixed	8																					
	No. of Common FIFOs per channel	Fixed	3																					
	Max no. of Common FIFOs	(n+1) * No. of Common FIFOs per channel	24																					
	Avg. number of messages for RXMB and FIFO buffers for each channel	Parameter (256/ch, 128/ch, 64/ch, 48/ch, 32/ch)	256	128	64	48	32	256	128	64	48	32	256	128	64	48	32							
	No. of Bytes for each stored message	Fixed	-																					
	Average size of a Message Buffer in Bytes		76																					
	Number of Bytes in Message Pool area	(n+1) * Avg. number of messages for RXMB and FIFO buffers for each channel * (No. of Bytes for each stored message or Average size of a Message Buffer in Bytes based on CAN_FD_MODE)	155648	77824	38912	29184	19456	155648	77824	38912	29184	19456	155648	77824	38912	29184	19456							
	Number of Bytes Message RAM	Number of Bytes in Message Pool area + Number of Bytes in OTB area + Number of Bytes in THL area	197888	120064	81152	71424	61696	178432	100608	61696	51968	42240	168704	90880	51968	42240	32512							



RAM Name	RAM Property	RAM Area Calculation Method	MRAM Values																			
	No. of Channels	(n+1)	6																			
TX MB	No. of TX MBs per channel	Parameter (64/ch, 32/ch, 16/ch)	64				32				16											
	Max no. of TX MBs	(n+1) * No. of TX MBs per channel	384				192				96											
	No. of Bytes needed for each TX MB	Fixed	76				76				76											
	Number of Bytes in TX MB area	(n+1) * No. of TX MBs per channel * No. of Bytes needed for each TX MB	29184				14592				7296											
THL	No. of entries in 1 THL buffer	Fixed	32																			
	Max no. of THL entries	(n+1) * No. of entries in 1 THL buffer	192																			
	No. of Bytes needed for each THL entry	Fixed	8																			
	Number of Bytes in THL area	Max no. of THL entries * No. of Bytes needed for each THL entry	1536																			
OTB	Avg. number of buffers for each channel		2																			
	Max no. of OTB entries	(n+1) * Avg. number of buffers for each channel	12																			
	No. of Bytes for OTB entry	Fixed	80																			
	Number of Bytes in OTB area	Max no. of OTB entries * No. of Bytes for OTB entry	960																			
Message Buffer	No. of RX MBs per channel	Fixed	16																			
	Max no. of RX MBs	(n+1) * No. of RXMBs per channel	96																			
	No. of RX FIFOs	Fixed	8																			
	No. of Common FIFOs per channel	Fixed	3																			
	Max no. of Common FIFOs	(n+1) * No. of Common FIFOs per channel	18																			
	Avg. number of messages for RXMB and FIFO buffers for each channel	Parameter (256/ch, 128/ch, 64/ch, 48/ch, 32/ch)	256	128	64	48	32	256	128	64	48	32	256	128	64	48	32					
	No. of Bytes for each stored message	Fixed	-																			
	Average size of a Message Buffer in Bytes		76																			
	Number of Bytes in Message Pool area	(n+1) * Avg. number of messages for RXMB and FIFO buffers for each channel * (No. of Bytes for each stored message or Average size of a Message Buffer in Bytes based on CAN_FD_MODE)	116736	58368	29184	21888	14592	116736	58368	29184	21888	14592	116736	58368	29184	21888	14592					
	Number of Bytes Message RAM	Number of Bytes in Message Pool area + Number of Bytes in OTB area + Number of Bytes in THL area	148416	90048	60864	53568	46272	133824	75456	46272	38976	31680	126528	68160	38976	31680	24384					



RAM Name	RAM Property	RAM Area Calculation Method	MRAM Values											
	No. of Channels	(n+1)	4											
TX MB	No. of TX MBs per channel	Parameter (64/ch, 32/ch, 16/ch)	64		32								16	
	Max no. of TX MBs	(n+1) * No. of TX MBs per channel	256		128								64	
	No. of Bytes needed for each TX MB	Fixed	76		76								76	
	Number of Bytes in TX MB area	(n+1) * No. of TX MBs per channel * No. of Bytes needed for each TX MB	19456		9728								4864	
THL	No. of entries in 1 THL buffer	Fixed			32									
	Max no. of THL entries	(n+1) * No. of entries in 1 THL buffer			128									
	No. of Bytes needed for each THL entry	Fixed			8									
	Number of Bytes in THL area	Max no. of THL entries * No. of Bytes needed for each THL entry			1024									
OTB	Avg. number of buffers for each channel				2									
	Max no. of OTB entries	(n+1) * Avg. number of buffers for each channel			8									
	No. of Bytes for OTB entry	Fixed			80									
	Number of Bytes in OTB area	Max no. of OTB entries * No. of Bytes for OTB entry			640									
Message Buffer	No. of RX MBs per channel	Fixed			16									
	Max no. of RX MBs	(n+1) * No. of RXMBs per channel			64									
	No. of RX FIFOs	Fixed			8									
	No. of Common FIFOs per channel	Fixed			3									
	Max no. of Common FIFOs	(n+1) * No. of Common FIFOs per channel			12									
	Avg. number of messages for RXMB and FIFO buffers for each channel	Parameter (256/ch, 128/ch, 64/ch, 48/ch, 32/ch)	256	128	64	48	32	256	128	64	48	32	256	128
	No. of Bytes for each stored message	Fixed											-	
	Average size of a Message Buffer in Bytes												76	
	Number of Bytes in Message Pool area	(n+1) * Avg. number of messages for RXMB and FIFO buffers for each channel * (No. of Bytes for each stored message or Average size of a Message Buffer in Bytes based on CAN_FD_MODE)	77824	38912	19456	14592	9728	77824	38912	19456	14592	9728	77824	38912
	Number of Bytes Message RAM	Number of Bytes in Message Pool area + Number of Bytes in OTB area + Number of Bytes in THL area	98944	60032	40576	35712	30848	89216	50304	30848	25984	21120	84352	45440

RAM Name	RAM Property	RAM Area Calculation Method	MRAM Values																			
	No. of Channels	(n+1)	2																			
TX MB	No. of TX MBs per channel	Parameter (64/ch, 32/ch, 16/ch)	64				32				16											
	Max no. of TX MBs	(n+1) * No. of TX MBs per channel	128				64				32											
	No. of Bytes needed for each TX MB	Fixed	76				76				76											
	Number of Bytes in TX MB area	(n+1) * No. of TX MBs per channel * No. of Bytes needed for each TX MB	9728				4864				2432											
THL	No. of entries in 1 THL buffer	Fixed	32																			
	Max no. of THL entries	(n+1) * No. of entries in 1 THL buffer	64																			
	No. of Bytes needed for each THL entry	Fixed	8																			
	Number of Bytes in THL area	Max no. of THL entries * No. of Bytes needed for each THL entry	512																			
OTB	Avg. number of buffers for each channel		2																			
	Max no. of OTB entries	(n+1) * Avg. number of buffers for each channel	4																			
	No. of Bytes for OTB entry	Fixed	80																			
	Number of Bytes in OTB area	Max no. of OTB entries * No. of Bytes for OTB entry	320																			
Message Buffer	No. of RX MBs per channel	Fixed	16																			
	Max no. of RX MBs	(n+1) * No. of RXMBs per channel	32																			
	No. of RX FIFOs	Fixed	8																			
	No. of Common FIFOs per channel	Fixed	3																			
	Max no. of Common FIFOs	(n+1) * No. of Common FIFOs per channel	6																			
	Avg. number of messages for RXMB and FIFO buffers for each channel	Parameter (256/ch, 128/ch, 64/ch, 48/ch, 32/ch)	256	128	64	48	32	256	128	64	48	32	256	128	64	48	32					
	No. of Bytes for each stored message	Fixed	-																			
	Average size of a Message Buffer in Bytes		76																			
	Number of Bytes in Message Pool area	(n+1) * Avg. number of messages for RXMB and FIFO buffers for each channel * (No. of Bytes for each stored message or Average size of a Message Buffer in Bytes based on CAN_FD_MODE)	38912	19456	9728	7296	4864	38912	19456	9728	7296	4864	38912	19456	9728	7296	4864					
	Number of Bytes Message RAM	Number of Bytes in Message Pool area + Number of Bytes in OTB area + Number of Bytes in THL area	49474	30018	20290	17858	15426	44610	25154	15426	12994	10562	42176	22720	12992	10560	8128					

Table 13.1 AFL RAM, MRAM and PFL RAM area calculation

RAM initialization cycle with each parameter is shown below.

Parameter Item	Parameter set														
No. of Channels	8														
No. of TX MBs per channel	64					32					16				
No. of Pool buffers per channel	256	128	64	48	32	256	128	64	48	32	256	128	64	48	32
RAM initialisation cycles (pclk cycle)	49474	30018	20290	17858	15426	44610	25154	15426	12994	10562	42176	22720	12992	10560	8128

Parameter Item	Parameter set														
No. of Channels	6														
No. of TX MBs per channel	64					32					16				
No. of Pool buffers per channel	256	128	64	48	32	256	128	64	48	32	256	128	64	48	32
RAM initialisation cycles (pclk cycle)	37106	22514	15218	13394	11570	33458	18866	11570	9746	7922	31634	17042	9746	7922	6098

Parameter Item	Parameter set														
No. of Channels	4														
No. of TX MBs per channel	64					32					16				
No. of Pool buffers per channel	256	128	64	48	32	256	128	64	48	32	256	128	64	48	32
RAM initialisation cycles (pclk cycle)	24738	15010	10146	8930	7714	22306	12578	7714	6498	5282	21090	11362	6498	5282	4066

Parameter Item	Parameter set														
No. of Channels	2														
No. of TX MBs per channel	64					32					16				
No. of Pool buffers per channel	256	128	64	48	32	256	128	64	48	32	256	128	64	48	32
RAM initialisation cycles (pclk cycle)	12370	7506	5074	4466	3858	11154	6290	3858	3250	2642	10546	5682	3250	2642	2034

Table 13.2 RAM initialisation cycle

		30500h
	Unused area	B9B0h
CFDCFCC10.CFDC =4d (32 Message) CFDCFCC10.CFPLS =7d (64byte) →76byte per Message	COM FIFO 10	B030h
CFDCFCC5.CFDC =6d (64 Message) CFDCFCC5.CFPLS =2d (16byte) →28byte per Message	COM FIFO 5	A930h
CFDCFCC0.CFDC =1d (4 Message) CFDCFCC0.CFPLS =0d (8byte) →20byte per Message	COM FIFO 0	A8E0h
CFDRFCC4.RFDC = 2d (8 Message) CFDRFCC4.RFPLS =0d (8byte) →20byte per Message	RX FIFO 4	A840h
CFDRFCC0.RFDC =3d (16 Message) CFDRFCC0.RFPLS =5d (32byte) →44byte per Message	RX FIFO 0	A580h
RXMB: CFDRMMB.NRXMB =4d (4 Message) CFDRMNB.RMPLS =3d(20byte) →32byte per RXMB	RX MB	A500h
	OTB	A000h
	THL 7	
	:	
	THL 0	9800h
	TXMB[511]	
	:	
	TXMB[0]	0000h

13.1 Examples

The Figure 13.2 below shows one possible configuration of a 8 channel version.

(unit : Byte)

Figure 13.2: RX MB + FIFO buffers RAM area configuration examples of a RS-CAN-FD 8 channel

version

13.2 OTB Area

The OTB area starts immediately after the area allocated for THL Buffers. The OTB is a special purpose buffer used by the RS-CAN-FD modules. This section of RAM area can be accessed only by the CPU in RAM Test mode. In RAM Test mode, this area does not function as OTB. The average number of buffers for each channel is 2. Each Buffer needs 80 Bytes. Hence, the total number of Bytes allocated for the OTB is $((n+1)*2)*80$ Bytes.

13.3 num_ram_chan dependencies

Depending on the num_ram_chan[3:0] configuration the sizes of the AFLRAM and MRAM differs. Further the number of initialization cycles of the memory and the number of RAM test pages is different. Dedicated values can be found in the following table. (When parameter maximum specification)

CH	num_ram_chan [3:0]	AFLRAM area size	MRAM area size	RAM initialisation cycles	RAM Test RTMPS range (*1)			
1	4'b0001	3072	24736	6186	00h .. 6Ch	(108d)		
2	4'b0010	6144	49472	12370	00h .. D9h	(217d)		
3	4'b0011	9216	74208	18554	00h .. 145h	(325d)		
4	4'b0100	12288	98944	24738	00h .. 1B2h	(434d)		
5	4'b0101	15360	123680	30922	00h .. 21Fh	(543d)		
6	4'b0110	18432	148416	37106	00h .. 28Bh	(651d)		
7	4'b0111	21504	173152	43290	00h .. 2FBh	(760d)		
8	4'b1000	24576	197888	49474	00h .. 364h	(868d)		
8	4'b0000	24576	197888	49474	00h .. 364h	(868d)		

(pclk cycle)

*1

4'b0001 (1Ch) : User should not access more than 160 Bytes in the last page

4'b0010 (2Ch) : User should not access more than 64 Bytes in the last page

4'b0011 (3Ch) : User should not access more than 224 Bytes in the last page

4'b0100 (4Ch) : User should not access more than 128 Bytes in the last page

4'b0101 (5Ch) : User should not access more than 32 Bytes in the last page

4'b0110 (6Ch) : User should not access more than 192 Bytes in the last page

4'b0111 (7Ch) : User should not access more than 96 Bytes in the last page

4'b1000 (8Ch) : User can access all area in the last page

4'b0000 (8Ch) : User can access all area in the last page

Other values are prohibited.

If **CFDGTSTCFG.PNFS=1'b1**, User can access the following area of the PFLRAM.

CH	num_ram_chan [3:0]	PFLRAM size (byte)	RAM Test page range (*2)		
1	4'b0001	720	0	...	2h (2d)
2	4'b0010	1440	0	...	5h (5d)
3	4'b0011	2160	0	...	8h (8d)
4	4'b0100	2880	0	...	Bh (11d)
5	4'b0101	3600	0	...	Eh (14d)
6	4'b0110	4320	0	...	10h (16d)
7	4'b0111	5040	0	...	13h (19d)
8	4'b1000	5760	0	...	16h (22d)
8	4'b0000	5760	0	...	16h (22d)

*2

4'b0001 (1Ch) : User should not access more than 208 Bytes in the last page

4'b0010 (2Ch) : User should not access more than 160 Bytes in the last page

4'b0011 (3Ch) : User should not access more than 112 Bytes in the last page

4'b0100 (4Ch) : User should not access more than 64 Bytes in the last page

4'b0101 (5Ch) : User should not access more than 16 Bytes in the last page

4'b0110 (6Ch) : User should not access more than 224 Bytes in the last page

4'b0111 (7Ch) : User should not access more than 176 Bytes in the last page

4'b1000 (8Ch) : User should not access more than 128 Bytes in the last page

4'b0000 (8Ch) : User should not access more than 128 Bytes in the last page

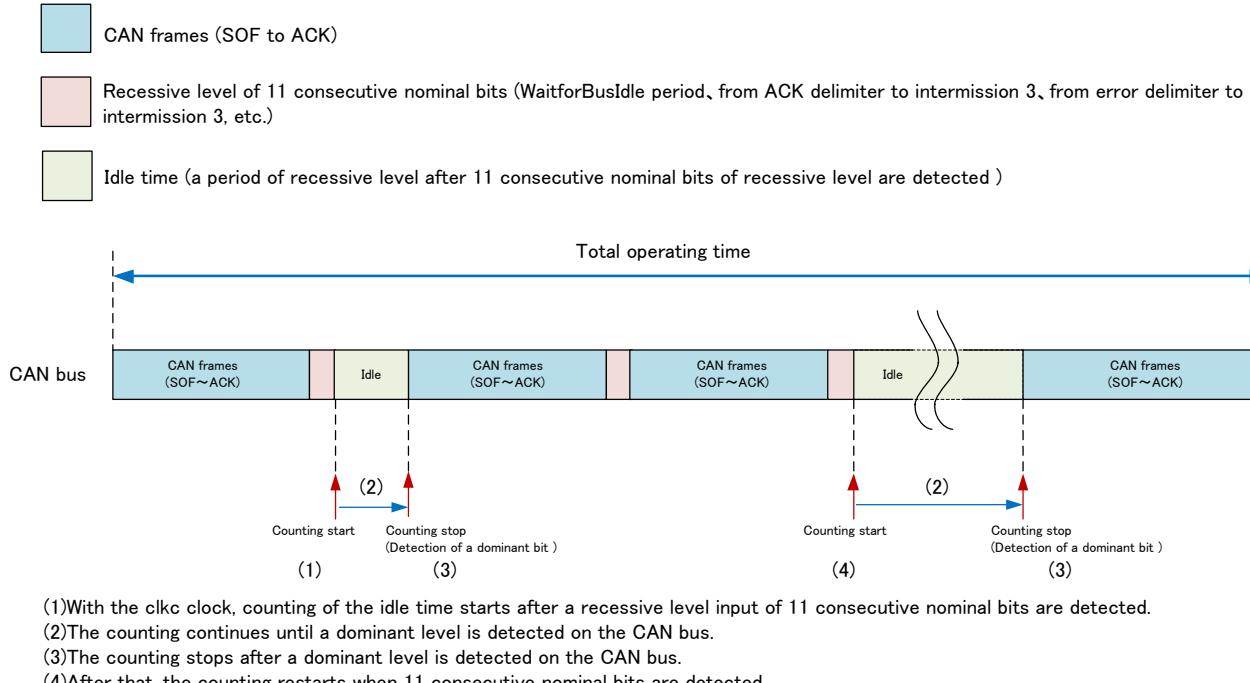
Other values are prohibited.

14 Bus traffic measurement

The idle time of the CAN bus can be measured using the clkc clock or clk_xincan clock. Bus traffic can be calculated based on the measurement results.

14.1 How to count the CAN bus idle time

Below is the concept of measuring the idle time of the CAN bus.



14.2 Operations and measurement procedure

The following are the procedure for measuring the idle time of the CAN bus.

- 1 The channel to be measured transits to operation mode.
- 2 Write 1 to **CFDCnBLCT.BLCE** bit to set the measuring counter to operating mode.
- 3 Write 1 to **CFDCnBLCT.BLCLD** bit to clear the counter register.
- 4 Detect a recessive level input of 11 consecutive nominal bits.
- 5 Start counting the bus idle time.
- 6 Detect a dominant level.
- 7 The counting stops.
- 8 Detect a recessive level input of 11 consecutive nominal bits.
- 9 The counting starts.
- 10 Write 1 to **CFDCnBLCT.BLCLD** bit to clear the counter register and simultaneously load the counter value to **CFDCnBLSTS**.
- 11 Read the value of **CFDCnBLSTS**.

To stop the measurement counter, write 0 to **CFDCnBLCT.BLCE** bit.

To initialize the counter, write 1 to **CFDCnBLCT.BLCLD** bit.

This measurement will be enabled when the channel to be measured is in operation mode.

When the relevant channels are in reset mode, the counter will not operate.

Also, accurate measurements are not available in test mode.

Write 1 to **CFDCnBLCT.BLCLD** bit to clear the counter register and simultaneously load the value of the counter to **CFDCnBLSTS**.

The lower three bits of **CFDCnBLSTS** are fixed to 0.

Based on the values of the counter, software can calculate the CAN bus traffic according to the following formulas.

$$\frac{(\text{total operating time} - \text{total idle time})}{\text{Total operating time}} = \frac{\text{bus operating time}}{\text{total operating time}} = \text{Bus usage ratio}$$

Total idle time: a value read from **CFDCnBLSTS** × a clock cycle of clkc

Total operating time: a setting interval of **CFDCnBLCT.BLCLD** bit

Example) Below is a calculation example under the following conditions.

Conditions: nominal bit rate = 1Mbps

clkc clock = 40MHz (=25ns)

a setting interval of **CFDCnBLCT.BLCLD** bit = cycle of 1ms

a read value of **CFDCnBLSTS** register = H'4E20 (D'20000)

$$\frac{(\text{total operating time} - \text{total idle time})}{\text{Total operating time}} = \frac{(1000000ns - 20000 * 25ns)}{1000000ns} = 50\%$$

15 Flexible CAN mode

This is a mode in which it is possible to connect the CAN modules of 2 channels to a single CAN driver.

The pair of Channel in flexible CAN mode is as follows.

When **CFDGFCMC.FLXC0** bit is set, Channel 0 and Channel 1 of a RS-CAN-FD module are Flexible CAN mode.

When **CFDGFCMC.FLXC1** bit is set, Channel 2 and Channel 3 of a RS-CAN-FD module are Flexible CAN mode.

When **CFDGFCMC.FLXC2** bit is set, Channel 4 and Channel 5 of a RS-CAN-FD module are Flexible CAN mode.

When **CFDGFCMC.FLXC3** bit is set, Channel 6 and Channel 7 of a RS-CAN-FD module are Flexible CAN mode.

Channel n+1 uses TX/RX terminal of Channel n.

The TX/RX terminal of Channel n+1 cannot use.

In Flexible CAN mode, each channel performs communication processing independently.

However, when one of the channels transmits, the other channel will not return an acknowledge bit.

Note: When operating in Flexible CAN mode the error counters (TEC/REC) of the two CAN nodes are not synchronised with each other.

Flexible transmission buffer assignment configured in **CFDGFTBAC** register and Flexible CAN mode configured in **CFDGFCMC** register should not be used simultaneously.

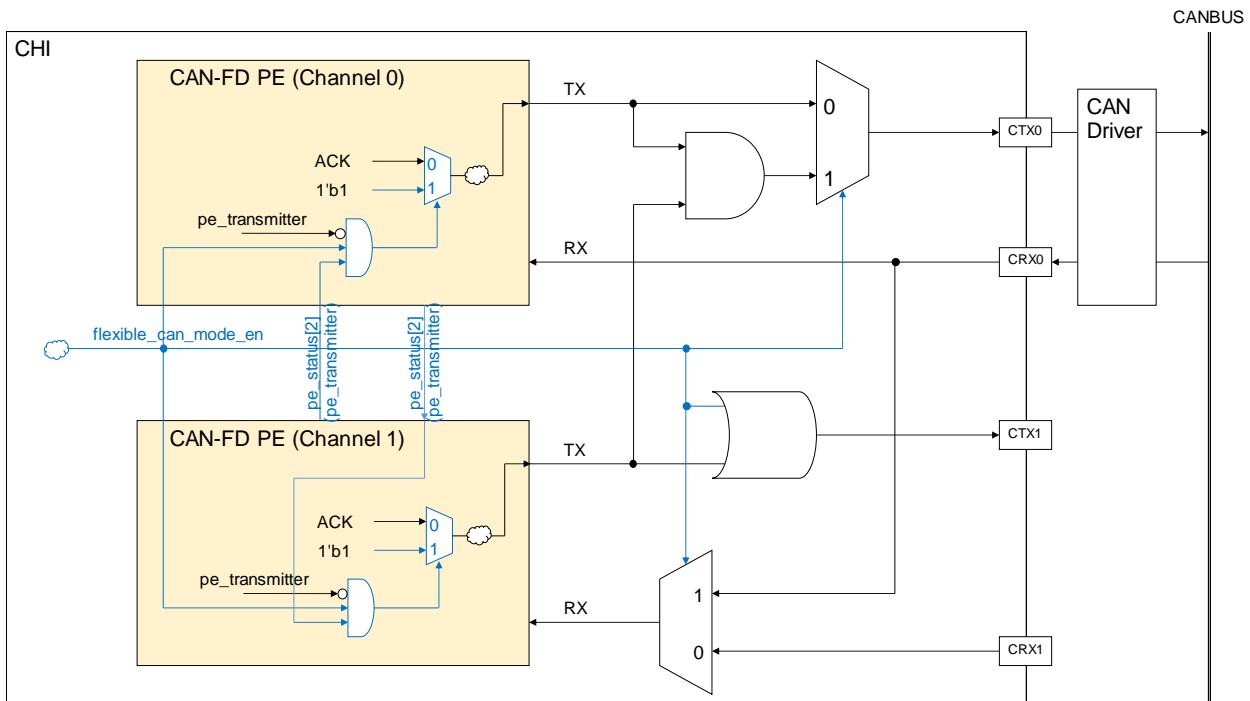


Figure 15–1 Diagram of the Flexible CAN

16 Flexible transmission buffer assignment

Each Channel has 64 transmit buffers by exclusive use.

In order to correspond to an insufficient case by a TXMB, this Channel can rent a maximum of 32 TXMBs from the next Channel.

The buffer which can be rented becomes every four units by 4 to 32 buffers.

The pair of Channel in flexible CAN mode is as follows.

When CFDGFTBAC.FLXMB0 bit is set, Flexible transmission buffer assignment between Channel 0 and Channel 1.

When CFDGFTBAC.FLXMB1 bit is set, Flexible transmission buffer assignment between Channel 2 and Channel 3.

When CFDGFTBAC.FLXMB2 bit is set, Flexible transmission buffer assignment between Channel 4 and Channel 5.

When CFDGFTBAC.FLXMB3 bit is set, Flexible transmission buffer assignment between Channel 6 and Channel 7.

Flexible transmission buffer assignment configured in **CFDGFTBAC** register and Flexible CAN mode configured in **CFDGFCMC** register should not be used simultaneously.

Interrupt of the rented buffer is outputted to interrupt of the rented channel.

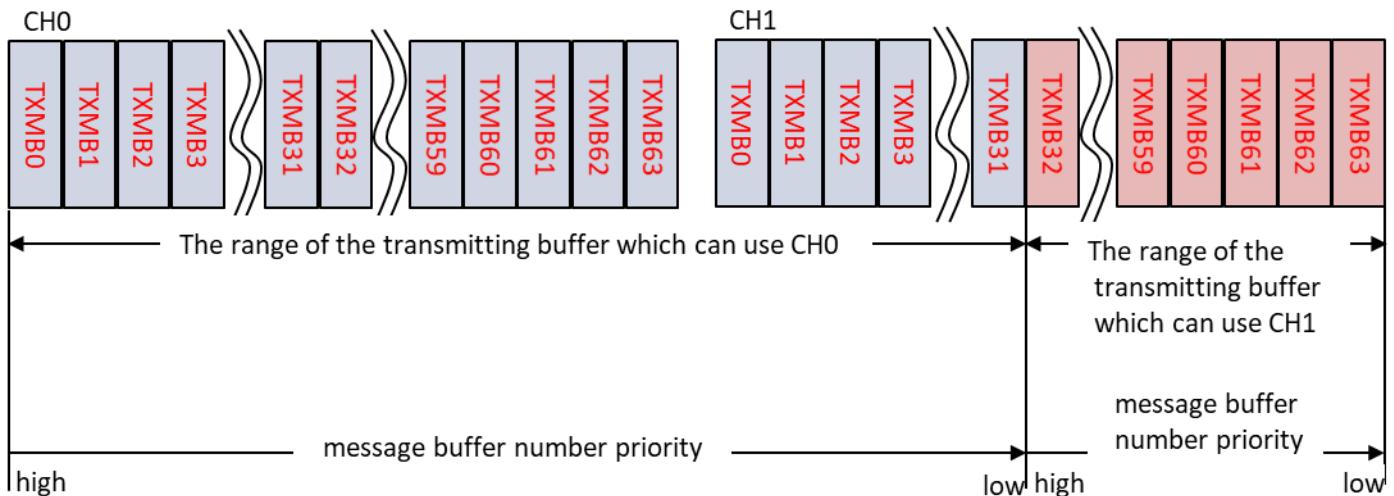
When using TXQ by the rented buffer, TXQ should only set within the rented range.

The TXMB lent out operates in the mode of the channel to be used.

For example, when the channel 1 is a reset mode, using rented TXMB, transmission of the channel 0 is possible.

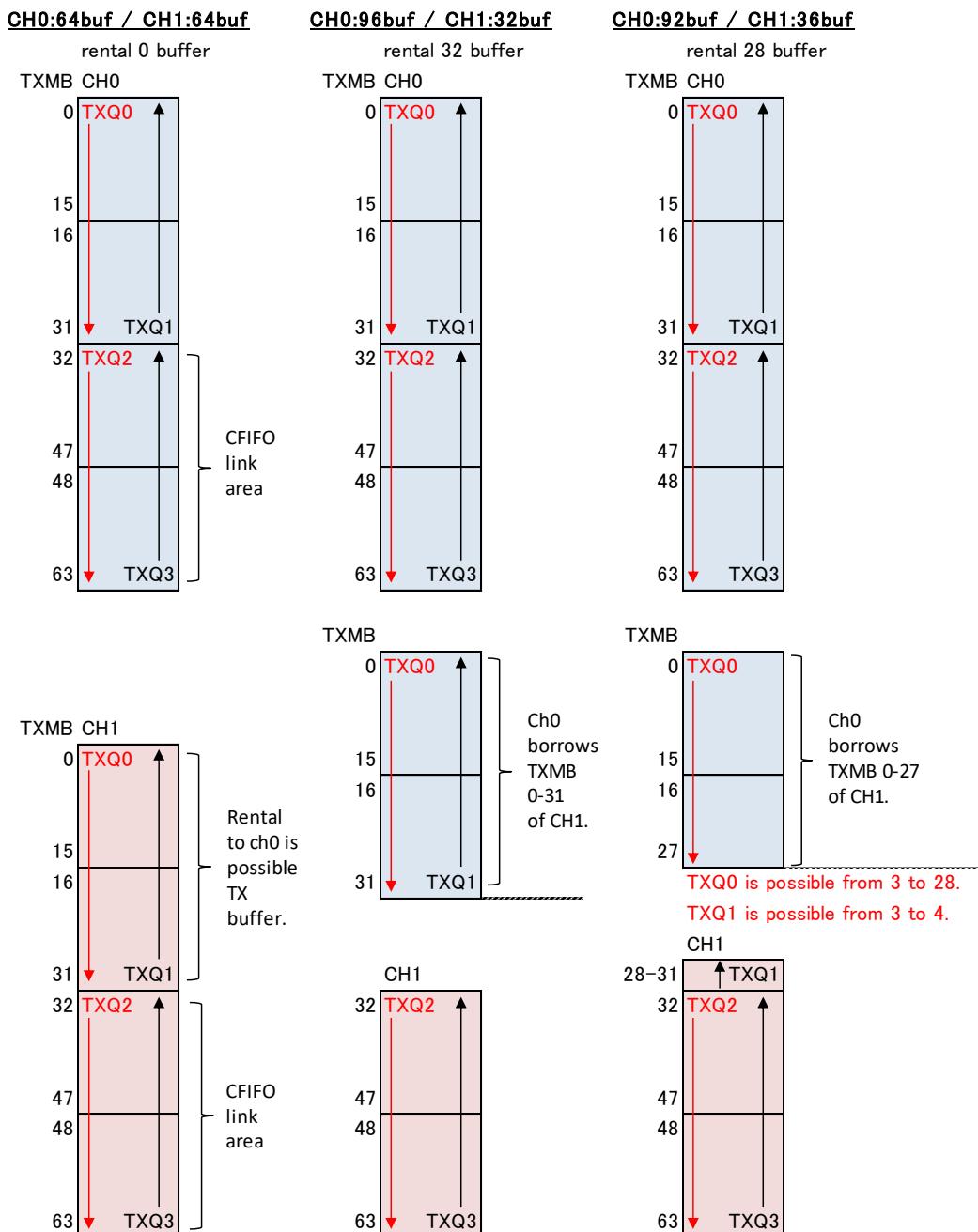
Moreover, rented TXMB is subject to the influence of the transmitting status, TX SCAN PROCESS, or transmitting abortion of the rented channel.

In the case of message buffer number priority mode, the priority of TXMB0 is high and the priority of TXMB95 (case which rented 32 buffers) becomes low. Priority of rented TXMB becomes low.



The priority of message buffer number priority mode (Example of CH0:96buf / CH1:32buf)

The example of a rental channel 0 and channel 1 is shown below.



The range of control of the channel 0.
channel_reset mode
Interrupt output

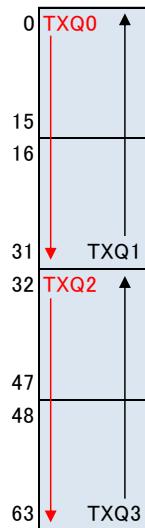
The range of control of the channel 1.
channel_reset mode
Interrupt output

Figure 16-1 Flexible transmission buffer assignment (0, 32, 28 TXMB rental)

CH0:88buf / CH1:40buf

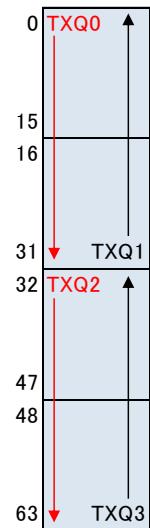
rental 24 buffer

TXMB CH0


CH0:84buf / CH1:44buf

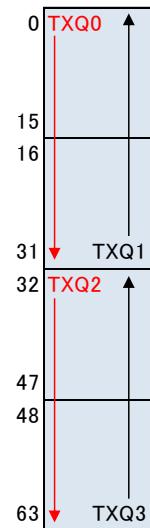
rental 20 buffer

TXMB CH0

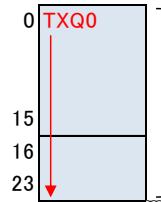

CH0:80buf / CH1:48buf

rental 16 buffer

TXMB CH0



TXMB

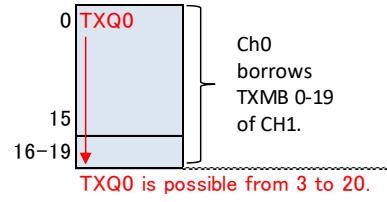


TXQ0 is possible from 3 to 24.

TXQ1 is possible from 3 to 8.

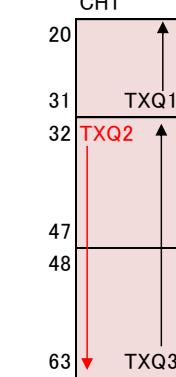


TXMB

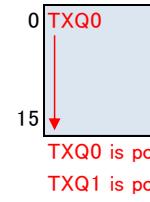


TXQ0 is possible from 3 to 20.

TXQ1 is possible from 3 to 12.



TXMB



TXQ0 is possible from 3 to 16.

TXQ1 is possible from 3 to 16.

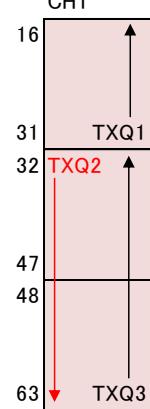
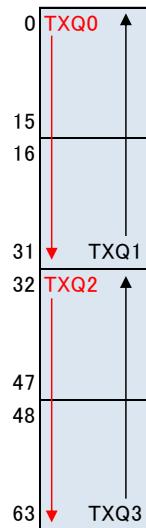


Figure 16-2 Flexible transmission buffer assignment (24, 20, 16 TXMB rental)

CH0:76buf / CH1:52buf

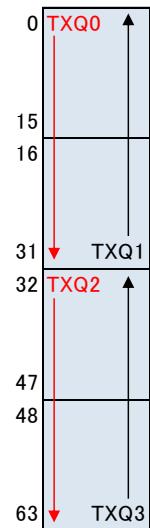
rental 12 buffer

TXMB CH0


CH0:72buf / CH1:56buf

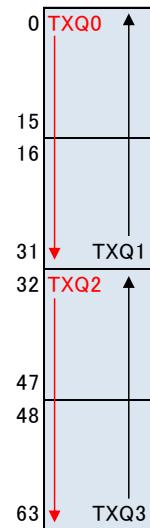
rental 8 buffer

TXMB CH0

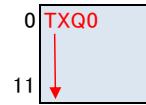

CH0:68buf / CH1:60buf

rental 4 buffer

TXMB CH0



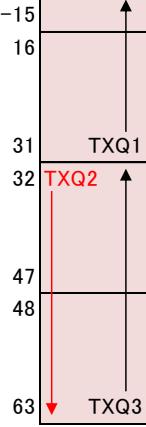
TXMB


Ch0
borrows
TXMB 0-11
of CH1.

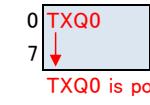
TXQ0 is possible from 3 to 12.

TXQ1 is possible from 3 to 20.

CH1

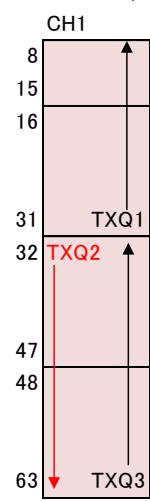


TXMB

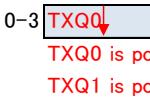

Ch0
borrows
TXMB 0-7
of CH1.

TXQ0 is possible from 3 to 8.

TXQ1 is possible from 3 to 24.



TXMB



TXQ dir

TXQ0 is possible from 3 to 4.

TXQ1 is possible from 3 to 28.

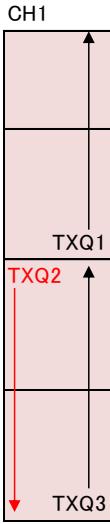


Figure 16–3 Flexible transmission buffer assignment (12, 8, 4 TXMB rental)

17 FFI mode

It becomes possible to control a channel register at each channel unit in order to prevent competition of register access with a multi-core by the FFI mode (Freedom From Inference).

Furthermore, it is controllable for each buffer unit.

Similarly, interrupt signal can also be outputted to each multi-cores.

Using this mode can correspond to FFI (Freedom From Interference) between channels and between buffers.

(The access guard of a register is not performed in RS-CANFD. The products design part side has an access guard of a register performed.)

Figure17.1 shows the image in each FFI mode

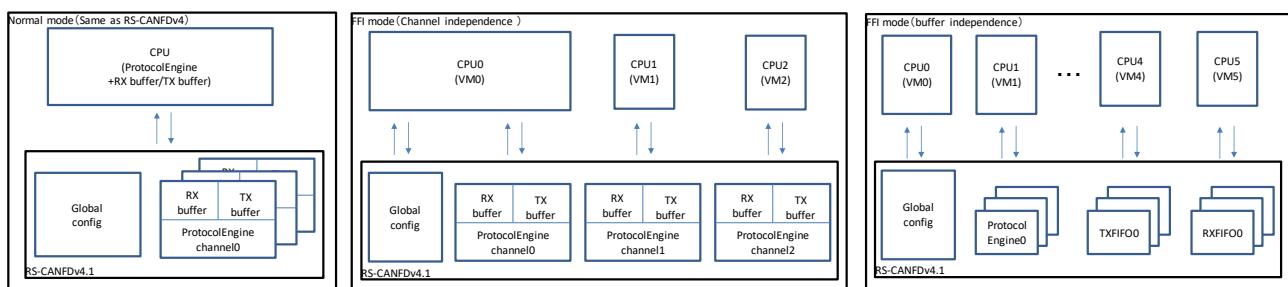


Figure 17.1 Image in each FFI mode

※VM=Virtual Machine

17.1 Outline in FFI mode (channel independence)

- 1 Global config is set by virtual machine for exclusive use. (It prevents accessing this area from other virtual machine.)
- 2 The register of each channel and the register of Reception FIFO, COMFIFO, and TXMB and TXQ buffer relation, assign to exclusive virtual machine respectively. channel 0 is assigned to VM0 and channel 1 is assigned to VM1 and ... the channel 7 is assigned to VM7. This VM number is a corresponding number to output of interrupt.
- 3 Each virtual machine cannot access the channel area which other virtual machine manage.
(It is based on the register access guard which is implemented by the product side.)
- 4 Virtual machine which manages each channel needs to manage two interruption, interruption in the existing channel unit, and interruption outputted per virtual machine.
- 5 That can assign to error interruption of each buffer from Global interruption the receiving mode of COMFIFO, Message lost of TXQ, THL, and message overwrite.

17.2 Outline in FFI mode (buffer independence)

- 1 Global config is set by exclusive virtual machine. (It prevents accessing this area from other virtual machine.)
- 2 The register of each channel is assigned to exclusive virtual machine respectively.
- 3 The register of Reception FIFO and COMFIFO and TXQ buffer relation is assigned to exclusive virtual machine respectively.
- 4 Each virtual machine cannot access the channel area which other virtual machine manage.
(It is based on the register access guard which is implemented by the product side.)
- 5 Buffer independence prohibits using the TXMB since a management of VM unit cannot control the TXMB.
- 6 That can assign to error interruption of each buffer from Global interruption the receiving mode of COMFIFO, Message lost of TXQ, THL, and message overwrite.

17.3 Prohibition function in FFI mode

- 1 CH independence becomes impossible when TXMB of other CH is used by flexible TXMB function.
Therefor FFI mode prohibit use of flexible TXMB function.

17.4 Restrictions function in FFI mode

- 1 Can't assign each channel a receiving buffer (RXMB).
Therefore, exclusive virtual machine manages a receiving buffer.
- 2 The number of VM is the same as the number of channels.
In the case of 8 channels, VM number become up to 8.
8 sets of interrupt are outputted.
- 3 THL is changed into virtual machine management from channel management.
(THL0 is assigned to VM0 ... THL7 is assigned to VM7)
In the case of FFI mode, THL is initialized by Global reset.
In the case of FFI mode, THL is not initialized by Channel reset.
(Store to THL is successful when an applicable channel reset occurs after channel fix by THL CH arbiter. Store to THL is unsuccessful when an applicable channel reset occurs before channel fix by THL CH arbiter.)
If the transmitting history of many buffers is stored in one THL, it will overflow.
It manages by each virtual machine so that the entry of THL may not overflow.
Moreover, since it is stored in THL from two or more buffers,
Timestamp value may be reversed with storing order.
- 4 The DMA control register moved to the control register of each buffer in FFI mode.
Each virtual machine can control DMA by this.
Moreover, the channel of the DMA controller of a system is managed for each virtual machine.
Therefore, virtual machine which manages the channel of a DMA controller and the buffer of RS-CANFD needs to be the same.

17.5 Interruption output in FFI mode

- In FFI mode, interrupt can be outputted to virtual machine unit.
The interrupt output can select a virtual machine for each buffer.
In FFI mode, transmitted and reception interrupt of each buffer are outputted only to interrupt of virtual machine unit.
In Normal mode, transmitted and reception interrupt of each buffer are outputted only to the existing interrupt.
Moreover, in FFI mode, the Global error interrupt of each buffer can select whether the existing interrupt output or the interrupt output of a virtual machine unit is used.
Figure17.2 shows the interruption output to VM in FFI mode.

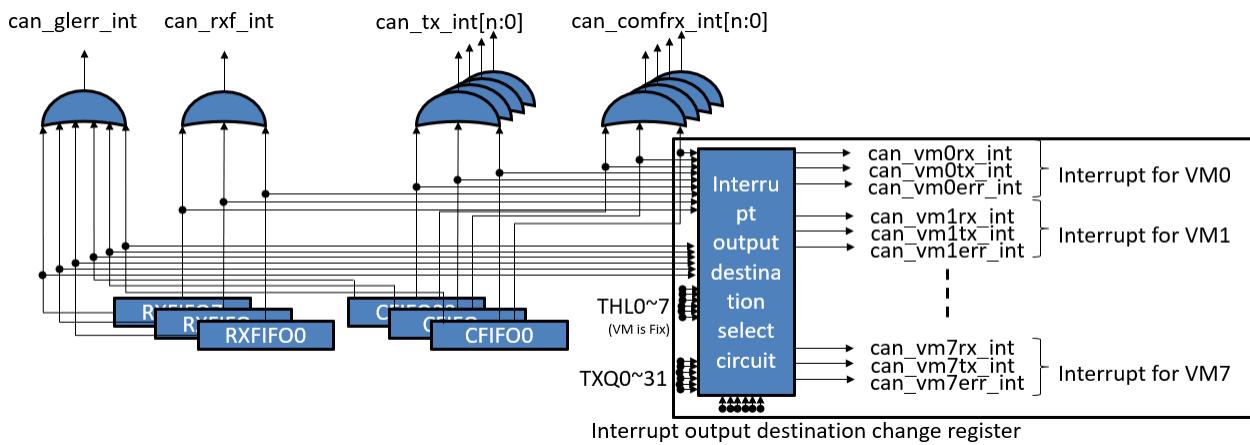


Figure 17.2 interruption output to VM in FFI mode

The interrupt output by the side of a virtual machine and the relation of a register setup are shown below.

CFDGFFIMC.FFIEN bit	CFDGVMEIS.FMLT bit	CFDVMRFCFG.VMN* bit	Interrupt output destination
0	x	x	can_glerr_int
x	0	x	can_glerr_int
		4'h0~4'h7	no output
		4'h8	can_vmrerr_int[0]
		4'h9	can_vmrerr_int[1]
	
		4'hF	can_vmrerr_int[7]

Table 17.1 RXFIFO message lost

CFDGFFIMC.FFIEN bit	CFDVMRFCFG.VMN* bit	Interrupt output destination
0	x	can_rxf_int
	4'h0~4'h7	no output
	4'h8	can_vmrx_int[0]
	4'h9	can_vmrx_int[1]

	4'hF	can_vmrx_int[7]

Table 17.2 RXFIFO reception success

CFDGFFIMC.FFIEN bit	CFDGVMEIS.FMLT bit	CFDVMCFGn.CF*VMN bit	Interrupt output destination
0	x	x	can_glerr_int
x	0	x	can_glerr_int
		4'h0~4'h7	no output
		4'h8	can_vmrerr_int[0]
		4'h9	can_vmrerr_int[1]
	
		4'hF	can_vmrerr_int[7]

Table 17.3 COMFIFO message lost

CFDGFFIMC.FFIEN bit	CFDGVMEMIS.CFMOW bit	CFDVMCFGn.CF*VMN bit	Interrupt output destination
0	x	x	can_glerr_int
x	0	x	can_glerr_int
		4'h0~4'h7	no output
		4'h8	can_vmerr_int[0]
		4'h9	can_vmerr_int[1]
	
		4'hF	can_vmerr_int[7]

Table17.4 COMFIFO message overwrite

CFDGFFIMC.FFIEN bit	CFDVMCFGn.CF*VMN bit	Interrupt output destination
0	x	can_comfrx_int[7:0]
	4'h0~4'h7	no output
	4'h8	can_vmrx_int[0]
	4'h9	can_vmrx_int[1]

	4'hF	can_vmrx_int[7]

Table17.5 COMFIFO reception success

CFDGFFIMC.FFIEN bit	CFDVMCFGn.CF*VMN bit	Interrupt output destination	THL FIFO entry destination
0	x	can_tx_int[7:0]	THL FIFO of related channel
	4'h0~4'h7	no output	no output
	4'h8	can_vmtx_int[0]	THL FIFO[0]
	4'h9	can_vmtx_int[1]	THL FIFO[1]

	4'hF	can_vmtx_int[7]	THL FIFO[7]

Table17.6 COMFIFO transmitting success

CFDGFFIMC.FFIEN bit	CFDGVMEMIS.TXQMLT bit	CFDVMCFGn.TXQ*VMN bit	Interrupt output destination
0	x	x	can_glerr_int
x	0	x	can_glerr_int
		4'h0~4'h7	no output
		4'h8	can_vmerr_int[0]
		4'h9	can_vmerr_int[1]
	
		4'hF	can_vmerr_int[7]

Table17.7 TXQ message lost

CFDGFFIMC.FFIEN bit	CFDGVMEIS.TXQOW bit	CFDVMCFGn.TXQ*VMN bit	Interrupt output destination
0	x	x	can_glerr_int
x	0	x	can_glerr_int
		4'h0~4'h7	no output
		4'h8	can_vmerr_int[0]
	1	4'h9	can_vmerr_int[1]
	
		4'hF	can_vmerr_int[7]

Table17.8 TXQ message overwrite

CFDGFFIMC.FFIEN bit	CFDVMCFGn.TXQ*VMN bit	Interrupt output destination
0	x	can_comfrx_int[7:0]
	4'h0~4'h7	no output
	4'h8	can_vmrx_int[0]
	4'h9	can_vmrx_int[1]

	4'hF	can_vmrx_int[7]

Table17.9 TXQ reception success

CFDGFFIMC.FFIEN bit	CFDVMCFGn.TXQ*VMN bit	Interrupt output destination	THL FIFO entry destination
0	x	can_tx_int[7:0]	THL FIFO of related channel
	4'h0~4'h7	no output	no output
	4'h8	can_vmtx_int[0]	THL FIFO[0]
	4'h9	can_vmtx_int[1]	THL FIFO[1]

	4'hF	can_vmtx_int[7]	THL FIFO[7]

Table17.10 TXQ transmitting success

CFDGFFIMC.FFIEN bit	CFDGVMEIS.THLELT bit	Interrupt output destination
0	x	can_glerr_int
	0	can_glerr_int
	1	can_vmerr_int[7:0]

Table17.11 THL entry lost

CFDGFFIMC.FFIEN bit	Interrupt output destination
0	can_tx_int[7:0]
1	can_vmtx_int[7:0]

Table17.12 THL entry success

17.6 Unavailable registers in FFI mode

Users should not access the following registers in FFI mode.

- | | |
|----------------------------------|-------------------------------------------------------------------------------------|
| 1 CFDGFTBAC | Flexible transmission buffer assignment function is prohibition of use in FFI mode. |
| 2 CFDCDTCT | moved to CFDCFCCEd and CFDRFCCa . |
| 3 CFDCDTTCT | moved to CFDCFCCEd , CFDTXQCC0n , and CFDTXQCC3n . |
| 4 CFDCDTSTS | moved to CFDCFSTSd and CFDRFSTSa . |
| 5 CFDCDTTSTS | moved to CFDCFSTSd , CFDTXQSTS0n , and CFDTXQSTS3n . |
| 6 CFDGRINTSTS_n | moved to CFDVMISTS_n . |

Users should not access the bit of the following registers in FFI mode.

- 1 **CFDGTINTSTS_v.CFOTIF_n**
- 2 **CFDGTINTSTS_v.TQOFIF_n**
- 3 **CFDGTINTSTS_v.THIF_n**
- 4 **CFDGTINTSTS_v.CFTIF_n**
- 5 **CFDGTINTSTS_v.TQIF_n**

In FFI mode, these bits become invalid and move to **CFDVMISTS_n**.

These bits are read as 0.

Users should not access the bit of the following registers when the VM error interrupt is selected in FFI mode.

These bits become valid when the Global error interrupt is selected in FFI mode. (as usual)

- 1 **CFDGERFL.MOWES**
- 2 **CFDGERFL.QMES**
- 3 **CFDGERFL.QOWES**
- 4 **CFDGERFL.THLES**
- 5 **CFDGERFL.MES**

These bits are read as 0.

Moved to **CFDVMISTS_n**.

- 6 **CFDGCTR.MOWEIE**
- 7 **CFDGCTR.QMEIE**
- 8 **CFDGCTR.QOWEIE**
- 9 **CFDGCTR.THLEIE**
- 10 **CFDGCTR.MEIE**

Moved to **CFDCFCCEd**, **CFDTXQCC0n**, **CFDTXQCC1n**, **CFDTXQCC2n**, **CFDTXQCC3n** and **CFDTHLCCn**.

The write value should be always 0. Read value of these bits is ignored.

17.7 Transition to FFI mode

Figure17.3 shows the FFI mode transition

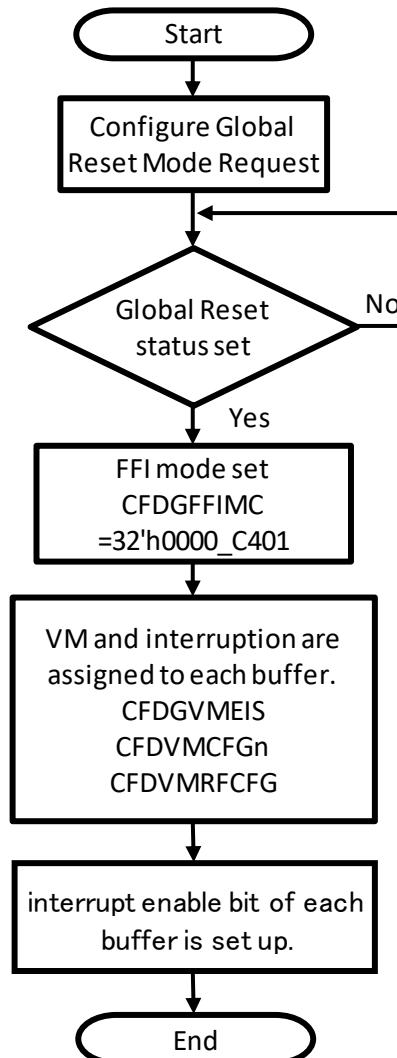


Figure17.3 FFI mode transition

The interrupt check flow in FFI mode is shown in Figure17.4.

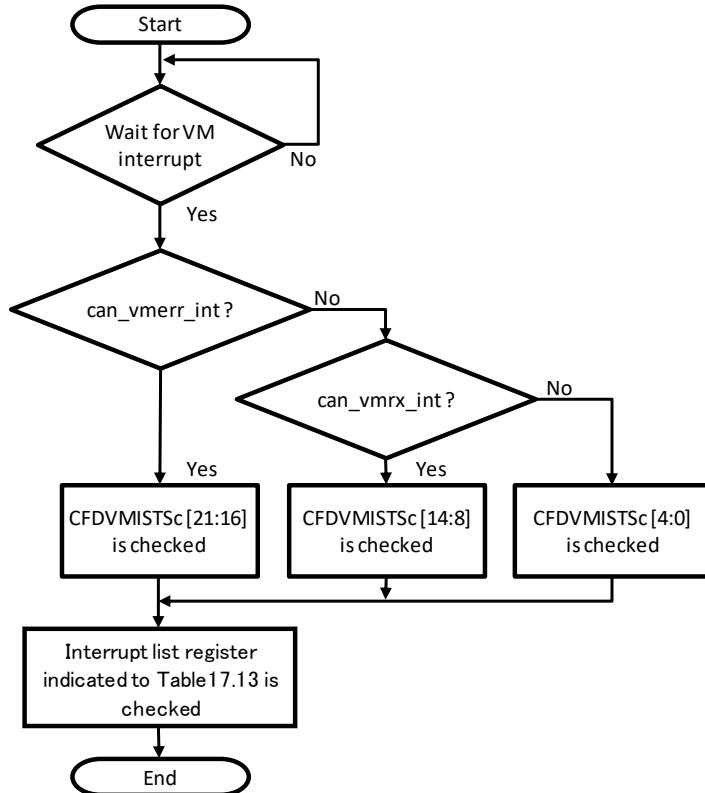


Figure17.4 Check flow of interrupt

Interrupt signal	CFDVMISTS bit No.	bit name	register to check
can_vmerr_int	b21	THLELT	CFDTHLSTS_n
	b20	TXQMOW	CFDTXQOWSTS
	b19	TXQMLT	CFDTXQMSTS
	b18	CFMOW	CFDCFМОWSTS
	b17	CFMLT	CFDFMSTS.CFxMLT
	b16	RFMLT	CFDFMSTS.RFxMLT
can_vmrx_int	b14	TXQFIF	CFDTXQFISTS
	b13	TXQOFRXIF	CFDTXQOFRISTS
	b12	CFFIF	CFDFFFSTS
	b11	CFOFRXINT	CFDCFOFRISTS
	b10	CFRXINT	CFDCFISTS
	b9	RFFIF	CFDRFISTS.RFxFFLL
	b8	RFIF	CFDRFISTS.RFxIF
can_vmtx_int	b4	THLIF	CFDTHLSTS_n
	b3	TXQOFTXIF	CFDTXQOFTISTS
	b2	TXQTXIF	CFDTXQISTS
	b1	CFOFTXINT	CFDCFOFTISTS
	b0	CFTXINT	CFDCFTISTS

Table17.13 Interrupt check register list

The transition to Normal mode from FFI mode is shown in Figure17.5.

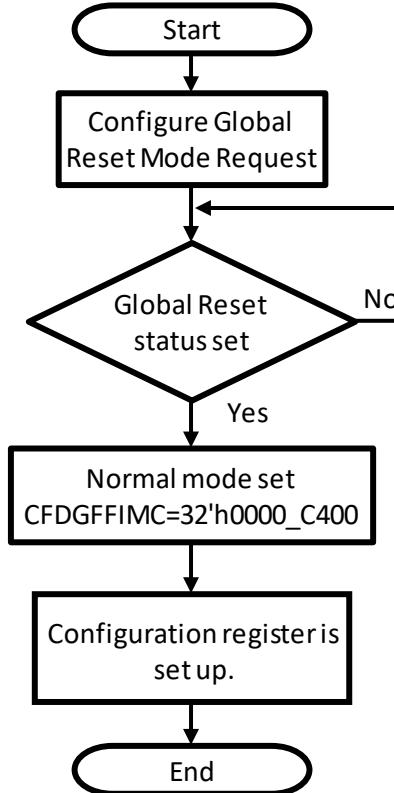


Figure17.5 Transition to Normal mode from FFI mode

17.8 Setup of an external terminal (rs_vnum_fix)

If an external terminal (rs_vnum_fix) is set to 1 in FFI mode, as for a **CFDV_nMCFG_n** register, the number of each channel will be set up.

In this case, the write of data becomes invalid at a **CFDV_nMCFG_n** register.

FFI mode, a setup of an external terminal (rs_vnum_fix), and the relation of a **CFDV_nMCFG_n** register are shown below.

Register name	Read value (When rs_vnum_fix of an external terminal is 1'b0 and CFDGFFIMC.FFIEN is 1'b1)	Read value (When rs_vnum_fix of an external terminal is 1'b1 and CFDGFFIMC.FFIEN is 1'b1)	Read value (When CFDGFFIMC.FFIEN is 1'b0)
CFDV ₀ MCFG ₀	CFDV ₀ MCFG ₀ register set value	32h0888_8888	32h0000_0000
CFDV ₀ MCFG ₁	CFDV ₀ MCFG ₁ register set value	32h0999_9999	32h0000_0000
CFDV ₀ MCFG ₂	CFDV ₀ MCFG ₂ register set value	32h0AAA_AAAA	32h0000_0000
CFDV ₀ MCFG ₃	CFDV ₀ MCFG ₃ register set value	32h0BBB_BBBB	32h0000_0000
CFDV ₀ MCFG ₄	CFDV ₀ MCFG ₄ register set value	32h0CCC_CCCC	32h0000_0000
CFDV ₀ MCFG ₅	CFDV ₀ MCFG ₅ register set value	32h0DDD_DDDD	32h0000_0000
CFDV ₀ MCFG ₆	CFDV ₀ MCFG ₆ register set value	32h0EEE_EEEE	32h0000_0000
CFDV ₀ MCFG ₇	CFDV ₀ MCFG ₇ register set value	32h0FFF_FFFF	32h0000_0000

CFDVMRCFG register does not control from an external terminal (rs_vnum_fix).
Users should set **CFDVMRCFG** register.

17.9 Measures of fail safe

When a channel is in an abnormal condition, CAN communication stops a corresponding channel by changing to a channel reset mode.

Furthermore, the buffer of a transmitting system related by this mode transition is cleared.

However, the buffer (RX mode of RXFIFO and COMFIFO) of a receiving system does not clear.

In the case of FIFO which the applicable channel is using by exclusive use, FIFO is cleared by clearing the enabling bit of each FIFO.

After performing a channel configuration again in the state of a channel reset mode, it changes to a channel operation mode and performs CAN communication.

Moreover, when performing fail-safe after completing CAN communication, channel mode changes in HALT mode.

CAN communication will be completed if transition is completed in Halt mode. Channel mode makes a reset mode change.

18 PNF (Pretended Network Filter)

The purpose of this function is to reduce the consumption current.

If without the PN mode, CPU check all frames which passed ID filter and decides application processing. Therefore, CPU is always active.

In PN mode, RS-CAN FD generate wake-up request for CPU by result of payload filter.

In this case, the "RS-CAN FD" process a part of CPU processing.

Therefore, CPU can sleep and reduce the consumption current.

RSCAN-FD controller compare a payload with the payload filter, and if successful, CPU wakeup by CAN reception Interrupt.

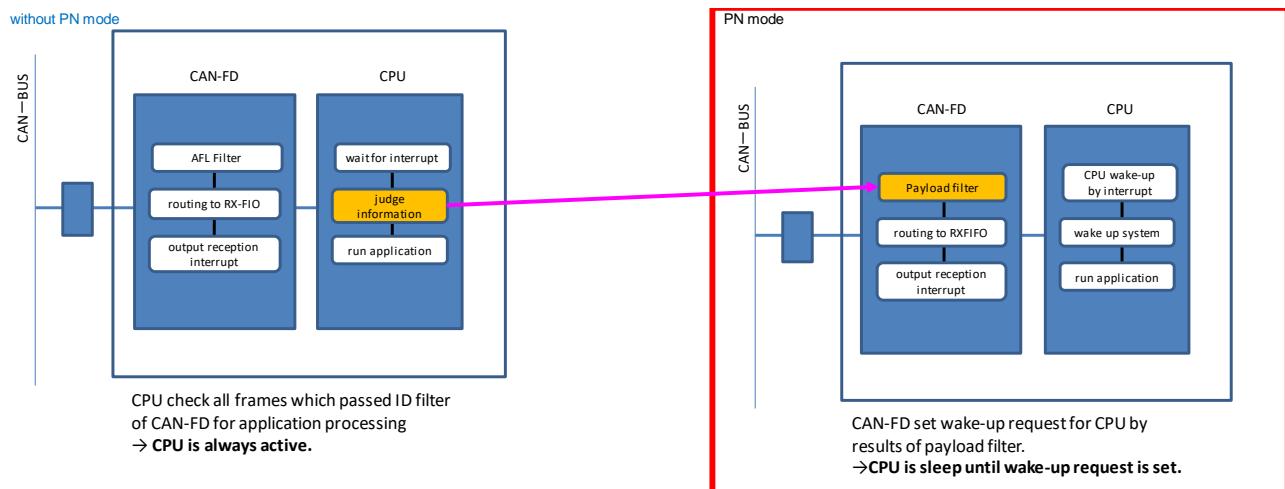


Figure 18.1 contents of processing in PN mode

18.1 Explanation in each mode

PNF compares ID and payload value of a receiving frame with a filter value.

The PNF operates independently from the AFL.

There are four kinds of combination of filtering of PNF and AFL.

User can select the mode by **CFDCnFDCTR.PNMDC** bit.

User can confirm the current mode by **CFDCnFDSTS.PNSTS** bit.

Mode Filter	Pretended Network Filter Mode (PN mode)	Normal Mode		
		Pretended Network Filter ID only and Acceptance Filter Mode	Acceptance Filter Mode	Pretended Network Filter and Acceptance Filter Mode
AFL Filter	OFF	ON	ON	ON
PN ID Filter	ON	ON	OFF	ON
PN Payload Filter	ON	OFF	OFF	ON
CFDCnFDCTR.PNMDC				
CFDCnFDSTS.PNSTS	11b	01b	00b	10b

Table 18.1 PN mode & Normal mode

18.1.1 Pretended Network Filter Mode

This mode is for CPU wake up from MCU STOP mode.

In PN mode, received ID and received payload are compared with PN filter.

In this mode AFL is not performed.

If ID and payload matches PN filter, this mode changes to the mode automatically set up in the **CFDCnFDCFG.RPNMD** bit.

Priority is given to mode transition of IP when the mode transition from IP and the mode transition from CPU compete.

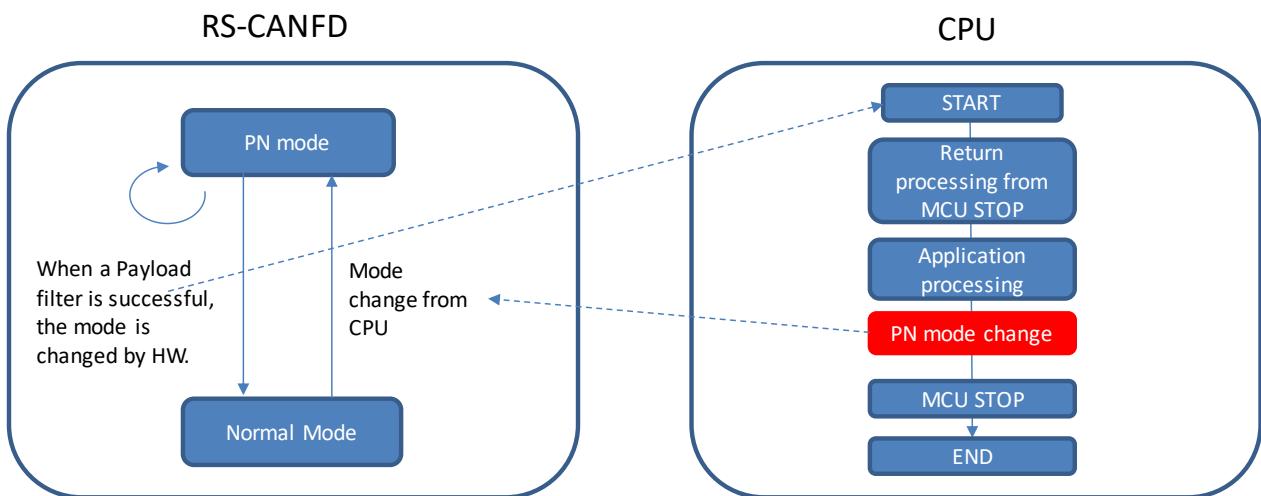


Figure 18.2 processing flow in PN mode

18.1.2 Pretended Network Filter ID only and Acceptance Filter Mode

In this mode, ID of a receiving frame is filtered using AFL and PNF.

Comparison of a payload is not performed.

When a filter matches by both AFL and PNF, routing is carried out based on the information on PNF, and routing of AFL is not performed.

AFL	PNF		Routing
ID Filter	ID Filter	Payload Filter	
unmatch	unmatch	not perform	received message is rejected
not perform	match		use PNF information
match	unmatch		use AFL information

Table 18.2 match/unmatch action of filter

18.1.3 Acceptance Filter Mode

This mode is the existing mode.

In this mode, ID of a receiving frame is filtered only by AFL.

The PNF is not performed.

AFL	PNF		Routing
ID Filter	ID Filter	Payload Filter	
unmatch	not perform	not perform	received message is rejected
match			use AFL information

Table 18.3 match/unmatch action of filter

18.1.4 Pretended Network Filter and Acceptance Filter Mode

In this mode, ID of a receiving frame is filtered using AFL and PNF.

PNF compares the payload of a receiving frame.

When a filter matches by both AFL and PNF, priority is given to PNF and then a payload is compared.

Routing will be performed if a payload filter is matches.

Received messages is rejected when a filter is unmatching.

AFL	PNF		Routing
ID Filter	ID Filter	Payload Filter	
unmatch	unmatch	not perform	received message is rejected
not perform	match	unmatch	received message is rejected
not perform	match	match	use PNF information
match	unmatch	not perform	use AFL information

Table 18.4 match/unmatch action of filter

18.2 Details of a filtering function

18.2.1 PN ID filter

PN ID filter is the same operation as ID filter of AFL.

However, PFL does not correspond to an AFL rewriting function.

Table 18.5 shows the behaviour of the Pretended network filter unit depending on the setting of the related input signals.

Mirror mode enable (MME configuration bit)	Loopback in test mode (Selftest Mode 0 or Selftest Mode 1)	Channel Mode is	Loopback configuration bit in PNF Entry	PNF entry is
0	0	Receiver	0	valid
			1	invalid
		Transmitter	0	invalid
			1	invalid
	1	Receiver	0	valid
			1	invalid
		Transmitter	0	valid
			1	valid
1	0	Receiver	0	valid
			1	invalid
		Transmitter	0	invalid
			1	valid
	1	Receiver	0	valid
			1	invalid
		Transmitter	0	valid
			1	valid

Table 18.5 Pretended network filter behaviour based on Loopback Configuration setting in PNF entry

18.2.2 PN payload filter

Payload filter compares 32-bit data at two places.

A comparison position is chosen by the preset value of offset. (Offset position is 4 byte boundaries)

Receiving frame is rejected when DLC is less than four.

Moreover, which also rejects a remote frame.

When the range specified by offset exceeds payload length, payload filter rejects a frame.

There are two filter in order to compare to two 32 bit data.

There are the two comparison methods in each filter.

A match filter compares a payload per bit.

An upper and lower limit filter confirms that the value of a payload is within range of upper and lower limit.
Or an upper and lower limit filter confirms that the value of a payload is out of range of upper and lower limit.

The filter 0 and the filter 1 can select the different comparison method.

The case considered as a pass by match of both the filter0 and the filter1.

The case considered as a pass by match of either one of the filter0 or the filter1.

A filter can be selected from these two cases.

The conditions within the range of an upper and lower limit filter are shown below.

Lower limit <= payload value <= upper limit

The conditions out the range of an upper and lower limit filter are shown below.

Payload value < lower limit or upper limit < payload value

18.2.3 Setup of the number of rules of PNF

PNF mounts RAM for exclusive use aside from AFL and stores a rule.

The number of rules becomes number of channels x 20 rule in all.

The number of rules per channel can be assigned in the 0 to 32 ranges.

9 registers per one rule are needed.

1entry=9Longword=36byte

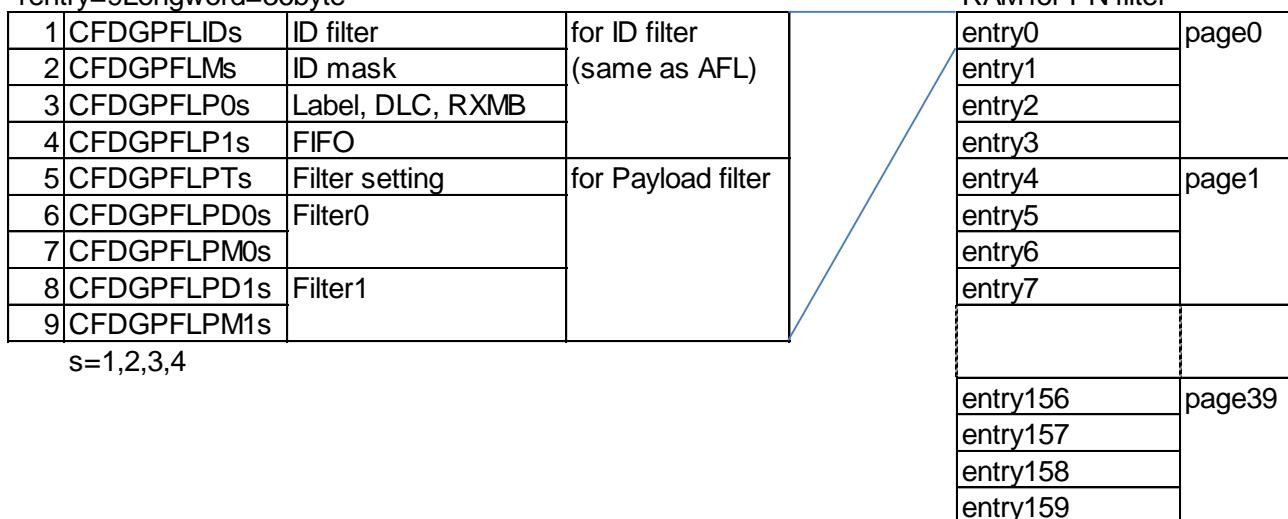


Figure 18.3 Configuration of PNF for each channel

18.2.4 Software flow of a payload filter

Application SW should not write numbers higher than 6'h27 for the PFL Page number.

The configuration flow shown in Figure 18.4 should be followed to program the PFL.

After entering all entries in Configuration Mode, locking of the PNF access should be performed to protect unwanted write access to the PFL.

Write protection is active during all global modes if the lock bit is set (GL_RESET, GL_HALT, and GL_OPERATION).

Read access to PFL is still possible during all global modes even if PFL data access is disabled (consistency check of PFL contents possible during runtime).

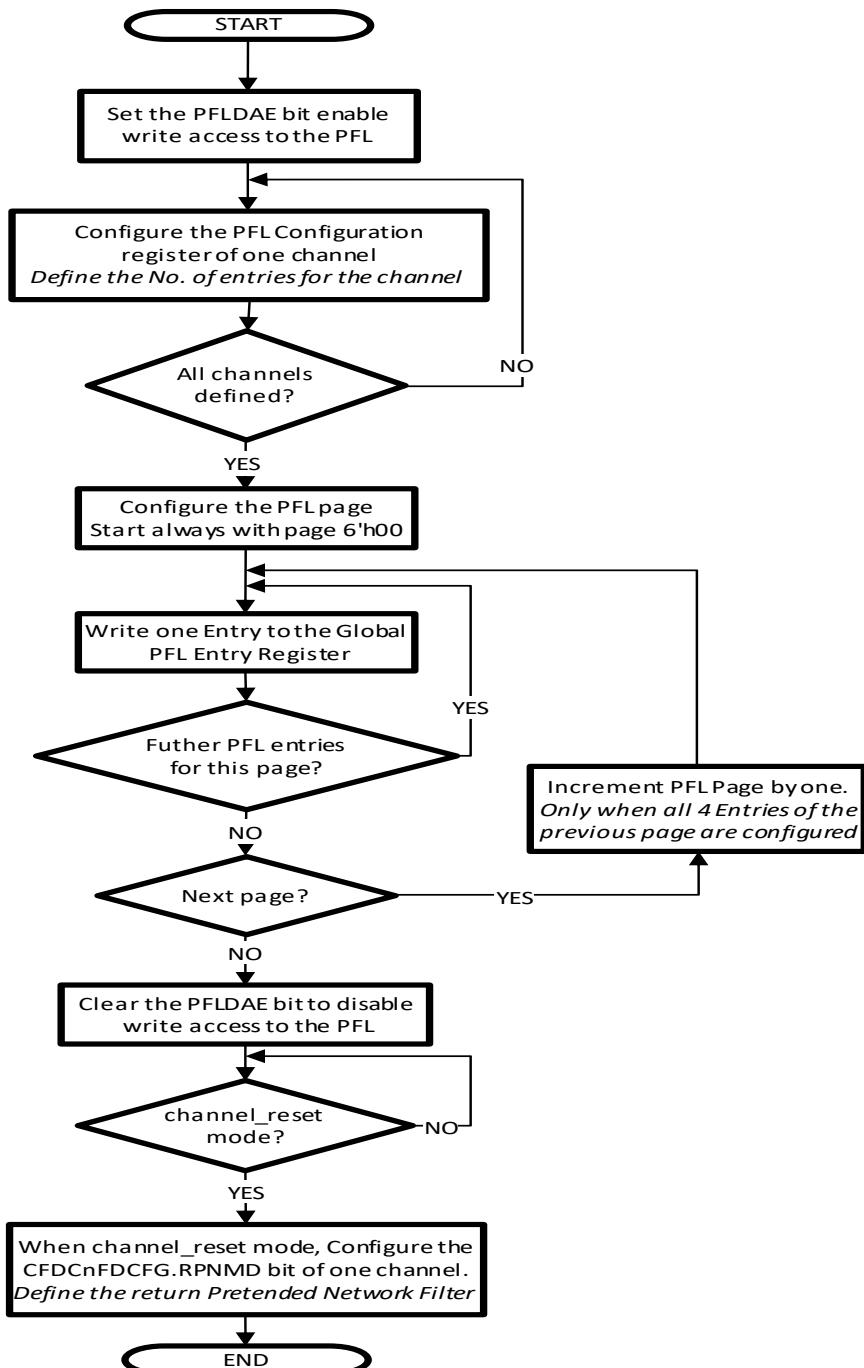


Figure 18.4 PNF configuration flow

Next, the software flow to operation in PN mode is shown below.

The interrupt return from PN mode becomes reception interrupt or Global error interrupt.

When the received payload filter frame matches ID filter and a payload filter, it is stored in RXFIFO etc. Reception interrupt will be generated if a receiving frame is stored in FIFO.

When the error of a DLC check or payload overflow occurs after matching ID filter and a payload filter, receiving frame is not stored in FIFO.

When these errors occur, Global error interrupt is generated.

CPU transitions to normal status from stop status by these two interrupts.

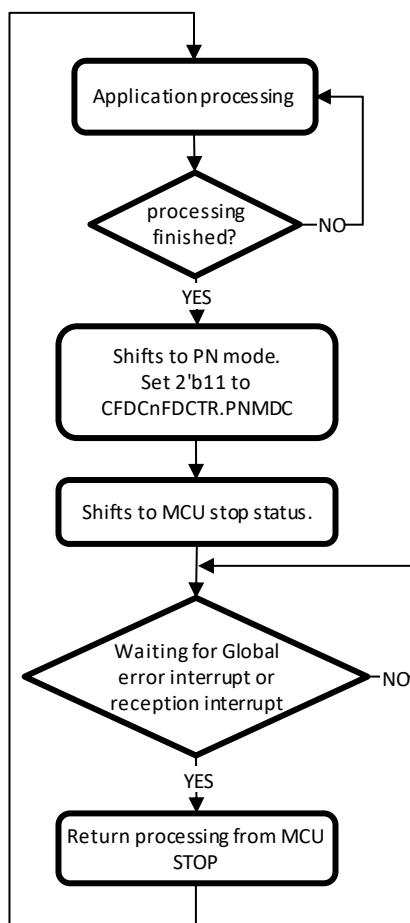


Figure 18.5 PNF operation flow

The transition to a normal mode from PN mode is shown below.

This is used when it cannot return from PN mode.

Mode transition will be performed if it writes in **CFDCnFDCTR.PNMDC**.

Users should write the same value as **CFDCnFDCFG.RPNMD** in **CFDCnFDCTR.PNMDC**.

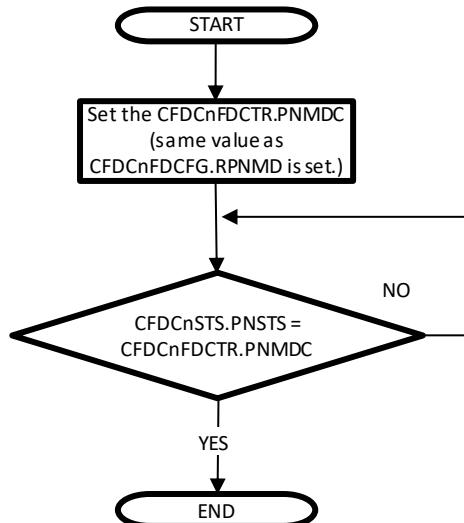


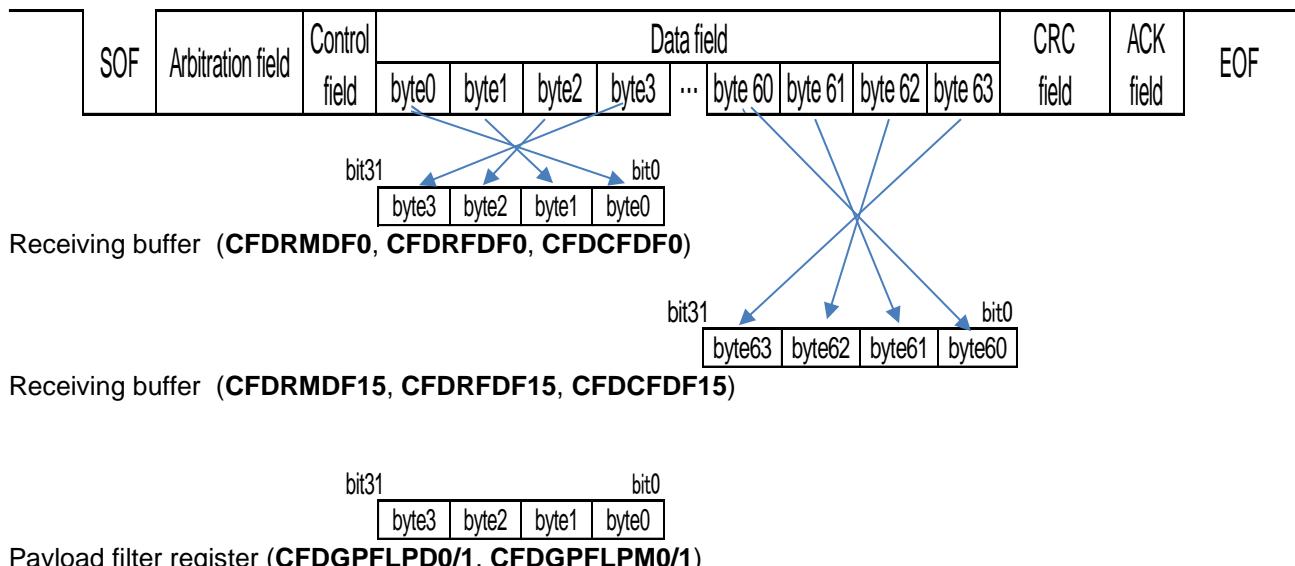
Figure 18.6 Normal mode transition

18.2.5 Byte alignment of payload filter

Byte alignment of the filter register (**CFDGPFLPD0/1**, **CFDGPFLPM0/1**) of an upper and lower limit filter is explained.

The CAN communications system consists of little endians. Moreover, RS-CANFD is little endian structure, endian is matching. Therefore, the endian of a register is as follows.

CAN/CANFD Frame



An upper and lower limit filter compares the data of this receiving buffer and the contents of the payload filter register.

New to issue	Sections	Change	Changed by	Date
1.00	All	New Release (IPS applies and creates "uciaprcn0140_IPSpec_v1.05.doc")	H.Nakamura	18-Mar-2021
	3.2.7 R-ACE Interface	can_race_ts[(n+1)*4-1:0] -> can_race_ts[(n+1)*16-1:0] is modified		
	3.2.7.1.14 can_race_ts	can_race_ts[(n+1)*4-1:0] -> can_race_ts[(n+1)*16-1:0] is modified		
	4.3.10.1 CFDGIPV.IPV	8'h43 = IP Release 4.1a is added		
	4.3.10.2 CFDGIPV.IPT	8'h43 is added		
	4.3.85.2 CFDGFDCFG.T SCCFG	Users should set 2'b01 to these bits when can_race_ts_en = 1 is added		
	4.3.76.5 CFDTHLACCOn. TMTS	The explanation in the case of can_race_ts_en=1 is added.		
	4.5.4.1 CFDRMPTR.RM TS			
	4.5.9.1 CFDRFPTR.RFT S			
	4.5.14.1 CFDCFPTC.CFT S			
	4.3.94 CFDVMCFGn	The register name controlled by each bit is added.		
	4.3.95 CFDVMRFCFG	The register name controlled by each bit is added.		
	3.2.5.1.19 can_race_ts[(n+1)*16-1:0]	Addition		
	3.2.5.1.20 can_race_ts_en[n:0]			
	3.2.5.1.21 can_rx_ts_captur e[n:0]			
	3.2.5.1.22 can_tx_ts_captur e[n:0]			
	4.3.99 CFDGPFLIDs	(no_of_CFDGAFLIDs=4) -> (no_of_CFDGPFLIDs=4) (s = [1...no_of_CFDGAFLIDs]) -> (s = [1...no_of_CFDGPFLIDs])		



New to issue	Sections	Change	Changed by	Date
	4.3.108 CFDGAFLIGNE NT	Changed from the red color(don't open to the customer) to the blue color(open to the customer). (CFDGAFLIGNENT,CFDGAFLIGNCTR)		
	4.3.109 CFDGAFLIGNC TR			
	5.4.3 Register behaviour in global/channel Modes			
	5.7 Influence of the bit by Parameterization			
	7.7 updating AFL entry during communication			
	2.1 Overview	Macro name uciaprcn0140 -> uciaprcn0141 Macro Version 4.1 -> 4.1a		
	2.4 Notes for the User Manual (UM) creation	A blue colour shows the Change part from RS-CANFDv4 to RS-CANFDv4.1kai. -> A blue colour shows the Change part from RS-CANFDv4.1 to RS-CANFDv4.1a.		
	3.2.1.2 Clock tree	Figure 3.2 Clock tree:uciaprcn0140 -> uciaprcn0141		
	4.2 Legend	CFDGIPV : 00000142h->00000143h		
	4.3.10 CFDGIP	CFDGIPV.IPV bit0 initial value : 0 -> 1		
	4.3.10.3 CFDGIPV.PSI	When IPV=8'h40, this bit becomes valid. ->These bits are valid for IPV=8'h40 or more.		
	5.4.3 Register behaviour in global/channel Modes	CFDGIPV.IPV:8'h42 -> 8'h43		
1.01	4.2 Legend	CFDPFLCFGw -> CFDPFLCFGu modified	H.Nakamura	22-May-2021
	4.3.98 CFDPFLCFGw	CFDPFLCFGw -> CFDPFLCFGu modified		
	5.6 Influence of the register by Parameterization	CFDPFLCFGw -> CFDPFLCFGu modified		
	4.3.30.8 CFDCFSTSd.CF OFRXIF	This bit is not influenced by the value of CFDCFCCd.CFIM .		
	4.3.30.9 CFDCFSTSd.CF OFTXIF	This bit is not influenced by the value of CFDCFCCd.CFIM .		
1.02	9.2 DMA Transfer	Corrected the following description <u>below is configuration flow for an initial set-up. →</u> <u>Figure 9.6 DMA enable flow is a configuration flow for an initial set-up modified</u>	H.Nakamura	2022/6/24



New to issue	Sections	Change	Changed by	Date
		<p>2. When transmit data can be used, CPU sets this data to Common FIFO or TXQ. →</p> <p>2. When transmit data can be used, CPU enables DMA to set this data to Common FIFO or TXQ.</p>		