

# Overview

The Controller Area Network (CAN) is a highly reliable serial bus protocol defined in the Bosch CAN specifications for standard CAN 2.0B and CAN FD, as well as ISO 11898-1:2024.

The TES CAN Flexible Data-Rate Controller IP core is a Hardware IP core written in VHDL. The core is intended for use in a System-on-Chip (SoC) environment. It can be integrated into a wide range of applications and target technologies. The code is synthesizable for Application Specific Integrated Circuits (ASIC) and Field Programmable Gate Arrays (FPGA).

## Features

- CAN 2.0B protocol compatible (ISO 11898-1).
- CAN Flexible Data-Rate (FD) protocol compatible (ISO 11898-1:2024).
- Supported Frame Formats:
  - CAN Base: 11-bit Identifier and constant bit rate.
  - o CAN Extended: 29-bit Identifier and constant bit rate.
  - CAN FD Base: CAN Base format with dual bit rate.
  - CAN FD Extended: CAN Extend with dual bit rate.
- Data rates:
  - Up to 1 Mbit/s for CAN 2.0B.
  - Up to 8 Mbit/s for CAN FD.
- Payload sizes:
  - CAN Base/Extended: 0...8 bytes.
  - CAN FD Base/Extended: 0...64 bytes.
- Clock Prescaler provides a wide frequency range.
- Flexible interface configurations.
- Receive frame filters.
- Bosch CAN Reference model certified for CAN 2.0B functionality.
- CAN 2.0B functionality certified for aviation-related applications.
- CAN FD functionality working in FPGA application.
- CAN FD in Silicon Q1'25.

# Function

#### MAC Device

The TES CAN FD Controller IP core is a device positioned in the OSI model in the Medium Access Control (MAC) sublayer section of the Data Link Layer (Layer 2).

The main tasks of an MAC layer device are:

- Message framing.
- Transmission and reception of frames.
- Arbitration.
- Error detection and signaling.
- Fault confinement.
- Bit timing and synchronization.

CAN Flexible Data-Rate Controller IP Core – Datasheet



V1.0, February 20<sup>th</sup> 2025

### Logical Link Control (LLC) Support

To achieve a high application flexibility for this core, the acceptance filtering and buffering of received messages is provided for Logical Link Control (LLC, part of layer 2 of the OSI model) sublayer application support, which may be implemented using hardware, software, or a combination of both.

#### **RX Filter**

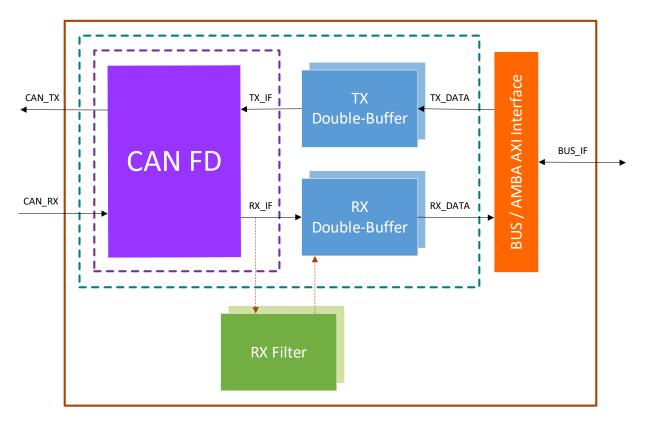
Multiple RX frame filter can be added. The filter can be masked for standard and extended identifier and the frame format. Each filter can be enabled and disabled separately. Each filter has a separate interrupt enable and interrupt flag.

### **Interface Configurations**

The TES CAN FD Controller IP core can be integrated into SoC device or other applications using one of the following configurations:

- Core interface with a memory interface.
- Core interface double buffers for transmit and receive. Data interface to the double buffers.
- BUS / AMBA AXI interface for access via internal SoC bus.

The RX Filter can be added in each variant.



### Core Interface with a memory interface

The CAN FD IP core interface is defined in following table. There is full flexibility to configure and control the CAN FD to transmit and receive frames.

Signal name	Туре	Description		
clk	In	System clock		
reset_n	In	Reset input		
Synchronization Nominal				
tseg1_i	In[5:0]	Time segment 1 nominal		
tseg2_i	In[4:0]	Time segment 2 nominal		



prescale_i In[4:0] Prescaling factor nominal Synchronization Data Useg1_dat_i In[5:0] Time segment 1 data Useg1_dat_i In[4:0] Time segment 2 data Sigw_dat_i In[4:0] Time segment 2 data Sigw_dat_i In[4:0] Prescaling factor data Delay Compensation delay_comp_en_i In Delay compensation offset delay_comp_of_i In[6:0] Delay compensation offset delay_comp_of_i In[6:0] Delay compensation offset delay_comp_on_d Out Pelay compensation enable delay_comp_on_of_i Un[6:0] Delay compensation enable valid (internal proofe) RX Buffer I Tam_rx_ena_0 Out RX Buffer enable ram_rx_addr_0 Out[9:0] RX Buffer data input ram_rx_data_0 Out[9:0] RX Buffer data output TX Buffer IF Tam_tx_ena_0 Out TX Buffer data output TX Buffer IF Tam_tx_ena_0 Out TX Buffer anable ram_tx_data_0 Out[3:0] TX Buffer adatess ram_tx_data_0 Out[3:0] TX Buffer adate output ram_tx_data_0 Out[3:0] TX Buffer adate output ram_tx_data_0 Out[3:0] TX Buffer adate output ram_tx_data_0 Out[3:0] TX Buffer adat output ram_tx_data_0 Out[3:0] TX Buffer adate output ram_tx_revini In TX Buffer data output ram_tx_data_0 Out[3:0] TX Buffer adate output ram_tx_data_0 Out[3:0] TX Buffer adate output ram_tx_data_0 Out[3:0] TX Buffer adat output ram_tx_data_0 Out[3:0] TX Buffer adat output ram_tx_data_0 Out[3:0] TX Buffer adat output ram_tx_revini In TX Suffer ready input Control Interface TX_received message (without any error) rx_receive_don_0 Out RX received message (without any error) rx_receive_don_0 Out RX pending (during receiving a frame) tx_start_i In TX stor request to send a frame tx_start_i In TX stor request to send a frame tx_stor_0 Out CAN TX (of serial interface) tx_retransmi_max_i In[2:0] TX retransitions (7-> infinite) CAN Interface TX erevine Out CAN TX (of serial interface) tx_events_0 Out CAN TX outper Interface) tx_retransmi_max_i In[2:0] TX error counter re	sjw_i	In[3:0]	Sample jump width nominal			
Synchronization Data $tseg1_dat_i$ In[5:0]Time segment 1 data $tseg2_dat_i$ In[4:0]Time segment 2 data $sjw_dat_i$ In[3:0]Sample jump width data $prescale_dat_i$ In[4:0]Prescaling factor dataDelay CompensationDelay compensation offsetdelay_comp_en_iInDelay compensation offsetdelay_comp_en_yos_oOutDelay compensation offsetdelay_comp_en_vol_oOutDelay compensation positiondelay_comp_en_vol_oOutDelay compensation positiondelay_comp_en_vol_oOutRX Buffer enableram_rx_ena_oOutRX Buffer addressram_rx_addr_oOut[9:0]RX Buffer addressram_rx_data_iIn[3:10]RX Buffer ready inputram_rx_data_oOutTX Buffer ready inputram_tx_data_oOutTX Buffer ready inputram_tx_data_oOutTX Buffer ready inputram_tx_data_iIn[3:10]TX Buffer ready inputram_tx_data_oOut(9:10]TX Buffer ready inputram_tx_data_iIn[3:10]TX Buffer data outputram_tx_data_iIn[3:10]TX Buffer data outputram_tx_data_iIn[3:10]TX Buffer data outputram_tx_data_iIn[3:10]TX Buffer ready inputControl InterfaceTXram_tx_data_iIn[3:10]TX Buffer ready inputControl InterfaceTXrx_pending_oOutRX received message (without any error)rx_pending_oOut		In[4:0]				
tseg1_dat_iIn[5:0]Time segment 1 datatseg2_dat_iIn[4:0]Time segment 2 datasjw_dat_iIn[6:0]Sample jump width datapelsale_dat_iIn[4:0]Prescaling factor dataDelay Compensationdelay_comp_on_iInDelay compensation enabledelay_comp_on_idelay_comp_on_oOut[7:0]Delay compensation onfsetdelay_comp_on_oOut[7:0]Delay compensation positiondelay_comp_on_oOut[7:0]Delay compensation positiondelay_comp_on_oOut [7:0]Delay compensation enable valid (internal proofe)RX Buffer IFTram_rx_ena_0OutRX Buffer enableram_rx_data_iIn[3:10]RX Buffer data inputram_rx_data_iIn[3:10]RX Buffer data outputram_rx_data_0Out[3:10]RX Buffer data outputram_rx_data_0Out TX Buffer data outputram_rx_data_0OutTX Buffer ready inputTX Buffer IFTram_tx_data_0Out [3:10]ram_rx_data_0Out [3:10]ram_rx_data_0Out [3:10]ram_rx_data_0Out [3:10]ram_tx_data_0Out [3:10]ram_tx_data_0Out [3:10]ram_tx_data_0Out [3:10]ram_tx_data_0Out [3:10]ram_tx_data_0Out RX received message (without any error)ram_tx_data_0Outram_tx_data_0Outram_tx_data_0Outram_tx_data_0Outram_tx_data_0Outram_tx_data	·					
tseg2dat_iIn[4:0]Time segment 2 datasjw_dat_iIn[4:0]Sample jump width dataprescale_dat_iIn[4:0]Prescaling factor dataDelay Compensationdelay_comp_en_iIndelay_comp_off_iIn[6:0]Delay compensation onffsetdelay_comp_os_oOut[7:0]Delay compensation positiondelay_comp_en_vld_oOutDelay compensation enable valid (internal proofe)RX Buffer IFTram_rx_ena_oOutRX Buffer addressram_rx_addf_oOut[3:0]RX Buffer addressram_rx_data_oOut[3:10]RX Buffer addressram_x_rdy_iInRX Buffer ready inputTX Buffer IFTTX Buffer IFTTx data_oOut[3:10]RX Buffer ready inputTX stry_rdy_iInRX Buffer ready inputTX buffer IFTTam_tx_ada_oOut [3:10]TX Buffer ready inputram_tx_rdy_iInRX Buffer ready inputTX buffer ifTTam_tx_data_oOut[3:0]TX Buffer addressram_tx_adat_iIn[3:0]TX Buffer addressram_tx_data_oOut[3:0]TX Buffer addressram_tx_data_oOutRX received message (without any error)rx_receive_done_oOutRX pending (during receiving a frame)tx_stop_iInTX stop requesttx_stop_iInTX stop requesttx_stop_iInTX stop requesttx_stop_iInTX stop requesttx_stop_i		In[5:0]	Time segment 1 data			
sjw_dat_i In[3:0] Sample jump width data prescale_dat_i In[4:0] Prescaling factor data Delay Compensation delay_comp_en_i In Delay compensation enable delay_comp_en_i O Dut Delay compensation offset delay_comp_en_vId_o Out Delay compensation onfset delay_comp_en_vId_o Out Delay compensation enable valid (internal proofe) RX Buffer IF ram_rx_ena_o Out RX Buffer enable ram_rx_data_i In[31:0] RX Buffer data input ram_rx_data_i Out[31:0] RX Buffer address ram_rx_data_i Out[31:0] RX Buffer readble ram_rx_data_o Out[31:0] RX Buffer readble ram_rx_rdy_i In RX Buffer readble ram_rx_data_o Out[31:0] RX Buffer readble ram_rx_data_o Out[31:0] RX Buffer readble ram_rx_data_o Out[31:0] RX Buffer readble ram_rx_data_o Out[31:0] RX Buffer address ram_rx_data_i In[31:0] TX Buffer readble ram_rx_data_o Out[31:0] TX Buffer readble ram_rx_data_i In[31:0] TX Buffer address ram_rx_data_i In[31:0] TX Buffer address ram_rx_data_i In[31:0] TX Buffer address ram_tx_data_0 Out[31:0] TX Buffer address ram_tx_data_0 Out[31:0] TX Buffer readble ram_tx_data_0 Out[31:0] TX Buffer readble ram_tx_data_0 Out[31:0] TX Buffer readble ram_tx_data_0 Out[31:0] TX Buffer readpress ram_tx_data_0 Out[31:0] TX Buffer readpress ram_tx_data_0 Out[31:0] TX Buffer readpress ram_tx_rdy_i In TX Buffer readpinput rx received one_0 Out RX received message (without any error) rx_pending_0 Out RX pending (during receiving a frame) tx_stor_i In TX stor request tx_stor_i In TX stor request tx_stor_i In TX stor request tx_pending_0 Out RX pending (during sending a frame) tx_retransmit_max_i In[2:0] TX retransitions (7 -> infinite) CAN Interface Fror Counters and States Error Counters and States Error Counters and States Erco_0 Out[8:0] TX error counter tec_warning_0 Out RX warning flag bus_off_0 Out CAN bus off error_passive_0 Out CAN bus off ervor_passive_0 Out CAN bus off ervor_passive_0 Out CAN bus off ervor_passive_0 Out CAN bus off ervor_maning event flags Testmode						
prescale_dat_iIn[4:0]Prescaling factor dataDelay Compensationdelay_comp_en_iInDelay compensation enabledelay_comp_pos_oOut[7:0]Delay compensation positiondelay_comp_pos_oOut[7:0]Delay compensation positiondelay_comp_nvld_oOutDelay compensation positiondelay_comp_nvld_oOutRX Buffer inram_rx_ena_oOutRX Buffer enableram_rx_addr_oOut[9:0]RX Buffer addressram_rx_data_iIn[31:0]RX Buffer data inputram_rx_data_oOut[9:0]RX Buffer data outputram_rx_data_oOut[31:0]RX Buffer rade saram_tx_ena_oOutTX Buffer rade saram_x_rdy_iInRX Buffer data outputram_rx_data_oOut[9:0]TX Buffer rade saram_tx_ena_oOutTX Buffer rade saram_tx_data_oOut[9:0]TX Buffer rade saram_tx_adar_oOut[9:0]TX Buffer data outputram_tx_adat_oOut[9:0]TX Buffer data outputram_tx_data_iIn[31:0]TX Buffer data outputram_tx_data_oOut[9:0]TX Buffer data outputram_tx_data_oOut[9:0]TX Buffer rade sage (without any error)rx_receive_done_oOutRX received message (without any error)rx_receive_done_oOutRX satr request to send a frame)tx_start_iInTX start request to send a frame)tx_start_iInTX start request to send a frame)tx_start_iInCAN						
Delay Compensationdelay_comp_en_iInDelay compensation offsetdelay_comp_pos_oOut[7:0]Delay compensation positiondelay_comp_en_vld_oOutDelay compensation enable valid (internal proofe)RX Buffer IFram_rx_ena_oOutRX Buffer Pram_rx_data_iIn[31:0]RX Buffer addressram_rx_data_oOut[31:0]RX Buffer ready inputram_rx_data_oOut[31:0]RX Buffer ready inputram_rx_data_oOut[31:0]RX Buffer ready inputrx data_oOut[31:0]RX Buffer ready inputrx data_oOut[31:0]rx Buffer ready inputrx data_oOut[9:0]rx Buffer ready inputram_tx_data_oOut[9:0]ram_tx_data_oOut[9:0]ram_tx_data_oOut[9:0]ram_tx_data_oOut[9:0]ram_tx_data_oOut[9:0]ram_tx_data_oOut[9:0]ram_tx_data_oOut[9:0]ram_tx_data_oOut[9:0]ram_tx_data_oOut[9:0]ram_tx_data_oOut[9:0]ram_tx_data_oOut[9:0]ram_tx_data_oOut[9:0]ram_tx_data_oOut[9:0]ram_tx_data_oOut[9:0]ram_tx_data_oOut[9:0]ram_tx_data_oOutram_tx_data_oOutram_tx_data_oOutram_tx_data_oOutram_tx_data_oOutram_tx_data_oOutram_tx_data_o						
delay_comp_en_iInDelay compensation offsetdelay_comp_off_iIn[6:0]Delay compensation offsetdelay_comp_en_vld_oOut[7:0]Delay compensation positiondelay_comp_en_vld_oOutDelay compensation enable valid (internal proofe)RX Buffer IFram_rx_ena_oOutRX Buffer enableram_rx_data_iIn[3:10]RX Buffer write enableram_rx_data_iIn[3:10]RX Buffer data inputram_rx_data_oOut[9:0]RX Buffer data outputram_rx_data_iIn[3:10]RX Buffer eady inputTX Buffer IFInRX Buffer ready inputTX Buffer IFTTram_tx_ena_oOutTX Buffer ready inputTX Buffer IFInRX Buffer data outputram_tx_ena_oOutTX Buffer ready inputTX Buffer IFInRX Buffer data outputram_tx_ena_oOutTX Buffer data inputram_tx_ena_oOutTX Buffer ready inputTX Buffer IFInTX Buffer adata inputram_tx_ena_oOutTX Buffer data outputram_tx_ena_oOutTX Buffer ready inputTX buffer of ada outputTX Buffer data outputram_tx_data_iIn[3:0]TX Buffer ready inputcontrol InterfaceTrx_receive_done_oOutRX pending (during receiving a frame)tx_start_iInTX start request to send a frametx_start_iInTX start request to send a frametx_stop_iInTX stop request <td>·</td> <td></td> <td></td>	·					
delay_comp_off_iIn[6:0]Delay compensation offsetdelay_comp_pos_oOut[7:0]Delay compensation positiondelay_comp_en_vld_oOutDelay compensation enable valid (internal proofe)RX Buffer IFram_rx_ena_oOutRX Buffer write enableram_rx_add_oOut[9:0]RX Buffer data inputram_rx_data_iIn[31:0]RX Buffer data outputram_rx_data_oOut[31:0]RX Buffer data outputram_rx_rdy_iInRX Buffer data outputram_tx_ena_oOutTX Buffer data outputram_tx_ena_oOutTX Buffer ready inputTX Buffer JTauffer enableram_tx_ena_oOutTX Buffer addressram_tx_addr_oOut[9:0]TX Buffer data inputram_tx_data_iIn[31:0]TX Buffer data outputram_tx_data_oOut[31:0]TX Buffer data outputram_tx_data_oOut[31:0]TX Buffer data outputram_tx_data_oOut[31:0]TX Buffer data outputram_tx_rdy_iInTX Buffer data outputram_tx_rdy_iInTX Suffer data outputram_tx_rdy_iInTX Buffer inceoking a frame)tx_start_iInTX suffer ready inputControl InterfaceOutRX pending (during receiving a frame)tx_start_iInTX start request to send a frametx_stop_iInTX stop requesttx_pending_oOutTX pending (during sending a frame)tx_ertansmit_max_iIn[2:0]TX retransition (7 -> infinite)		In	Delay compensation enable			
delay_comp_pos_oOut[7:0]Delay compensation positiondelay_comp_en_vld_oOutDelay compensation enable valid (internal proofe)RX Buffer IFram_rx_ena_oOutRX Buffer enableram_rx_addr_oOut[9:0]RX Buffer addressram_rx_data_iIn[31:0]RX Buffer data inputram_rx_data_oOut[31:0]RX Buffer data outputram_rx_data_oOut[31:0]RX Buffer ready inputTX Buffer IFram_tx_ena_oOutTX Buffer ready inputTX buffer IFram_tx_addr_oOut[9:0]TX Buffer addressram_tx_addr_oOut[1:0]TX Buffer addressram_tx_addr_oOut[1:0]TX Buffer addressram_tx_addr_oOut[31:0]TX Buffer addressram_tx_rdy_iInTX Buffer addressram_tx_rdy_iInTX Buffer addressram_tx_rdy_iInTX Buffer ready inputControl Interfacerx_receive_done_oOutRX received message (without any error)rx_pending_oOutRX pending (during receiving a frame)tx_start_iInTX start request to send a frametx_stop_iInTX pending (during sending a frame)tx_stop_iInTX pending (during sending a frame)tx_stop_iInTX start request to send a frametx_stop_iInTX stopre equesttx_pending_oOutTX retransitions (7 -> infinite)CAN Interfacerx		In[6:0]				
delay_comp_en_vld_oOutDelay compensation enable valid (internal proofe)RX Buffer IFram_rx_ena_oOutRX Buffer enableram_rx_wr_oOut (9:0)RX Buffer write enableram_rx_data_iIn[31:0)RX Buffer addressram_rx_data_oOut[9:0]RX Buffer data inputram_rx_data_oOut[3:0]RX Buffer data outputram_rx_data_oOut[3:0]RX Buffer ready inputTX Buffer IFTTX Buffer ready inputram_tx_ena_oOutTX Buffer enableram_tx_wr_oOutTX Buffer data inputram_tx_data_iIn[31:0]TX Buffer data outputram_tx_data_oOut[31:0]TX Buffer data outputram_tx_rdy_iInTX Buffer data outputram_tx_rdy_iInTX Buffer data outputram_tx_rdy_iInTX Buffer data outputram_tx_rdy_iInTX Buffer outputcontrol InterfaceInrx_receive_done_oOutrx_receive_done_oOutRX pending (during receiving a frame)tx_stopiInTX stop requesttx_retransmit_max_iIn[2:0]TX retransitions (7 -> infinite)CAN Interfacerx </td <td></td> <td></td> <td></td>						
RX Buffer IFram_rx_ena_oOutRX Buffer enableram_rx_wr_oOutRX Buffer write enableram_rx_addr_oOut[9:0]RX Buffer data sinputram_rx_data_iIn[31:0]RX Buffer data outputram_rx_data_oOut[31:0]RX Buffer data outputram_rx_rdy_iInRX Buffer ready inputTX Buffer IFInRX Buffer enableram_tx_ena_oOutTX Buffer enableram_tx_addr_oOut[9:0]TX Buffer addressram_tx_data_iIn[31:0]TX Buffer data inputram_tx_data_oOut[31:0]TX Buffer data outputram_tx_data_oOut[31:0]TX Buffer data outputram_tx_data_oOut[31:0]TX Buffer ready inputControl InterfaceTr x Eceived message (without any error)rx_pending_oOutRX received message (without any error)rx_pending_oOutTX supending (during sending a frame)tx_start_iInTX stop requesttx_pending_oOutTX retransitions (7 -> infinite)CAN InterfaceTrxInCAN RX (of serial interface)tx_wordOutCAN RX (of serial interface)txUCAN TX (of serial interface)txOutCAN RX (of serial interface)trec_oOut[8:0]TX error counterrec_oOutRX reror counterrec_oOutCAN bus offeevents_oOutCAN bus offeror_passive_oOutCAN bus off <td></td> <td></td> <td></td>						
ram_rx_wr_oOutRX Buffer write enableram_rx_addr_oOut[9:0]RX Buffer addressram_rx_data_iIn[31:0]RX Buffer data inputram_rx_data_oOut[31:0]RX Buffer data outputram_rx_dry_iInRX Buffer ready inputTX Buffer IFram_tx_wr_oOutTX Buffer enableram_tx_wr_oOutTX Buffer enableram_tx_wr_oOutTX Buffer data inputram_tx_data_iIn[31:0]TX Buffer data inputram_tx_data_iIn[31:0]TX Buffer data inputram_tx_data_iIn[31:0]TX Buffer data inputram_tx_data_iIn[31:0]TX Buffer data outputram_tx_data_oOut[31:0]TX Buffer data inputram_tx_data_iInTX Buffer data inputram_tx_data_oOut[31:0]TX Buffer data inputram_tx_data_oOut[31:0]TX Buffer data outputram_tx_data_oOutRX received message (without any error)rx_receive_done_oOutRX received message (without any error)rx_tart_iInTX start request to send a frame)tx_storp_iInTX stop requesttx_stop_iInTX received for senal interface)tx_retransmit_max_iIn[2:0]TX retransitions (7 -> infinite)CAN InterfaceInCAN RX (of serial interface)txOutCAN RX (of serial interface)txOutCAN RX (of serial interface)txOutCAN RX (of serial interface)tc<_o						
ram_rx_wr_oOutRX Buffer write enableram_rx_addr_oOut(9:0)RX Buffer addressram_rx_data_iIn[31:0]RX Buffer data inputram_rx_data_oOut[31:0]RX Buffer data outputram_rx_dry_iInRX Buffer ready inputTX Buffer IFram_tx_wr_oOutTX Buffer enableram_tx_data_iIn[31:0]TX Buffer enableram_tx_data_iIn[31:0]TX Buffer addressram_tx_data_iIn[31:0]TX Buffer data inputram_tx_data_iIn[31:0]TX Buffer data outputram_tx_data_oOut[31:0]TX Buffer data outputram_tx_data_oOut[31:0]TX Buffer data outputram_tx_data_oOut[31:0]TX Buffer data outputram_tx_rdy_iInTX Buffer ready inputControl Interfacerx_receive_done_oOutRX received message (without any error)rx_spending_oOutRX pending (during receiving a frame)tx_start_iInTX start request to send a frametx_stop_iInTX pending outputtx_retransmit_max_iIn[2:0]TX retransmitos (7 -> infinite)CAN InterfacerxInCAN RX (of serial interface)txOutCAN RX (of serial interface)txOutCAN TX (of serial interface)txOutCAN RX (of serial interface)rxInCAN RX (of serial interface)txOutCAN RX (of serial interface)txOut<	ram rx ena o	Out	RX Buffer enable			
ram_rx_addr_oOut[9:0]RX Buffer addressram_rx_data_iIn[31:0]RX Buffer data inputram_rx_data_oOut[31:0]RX Buffer data outputram_rx_rdy_iInRX Buffer data outputram_rx_rdy_iInRX Buffer ready inputTX Buffer IFram_tx_ena_oOutTX Buffer enableram_tx_addr_oOut[9:0]TX Buffer addressram_tx_data_iIn[31:0]TX Buffer data outputram_tx_data_oOut[31:0]TX Buffer data outputram_tx_data_oOut[31:0]TX Buffer ready inputcontrol Interfacerx_receive_done_oOutRX received message (without any error)rx_start_iInTX start request to send a frametx_stop_iInTX stop requesttx_stop_iInTX stop requesttx_nection_max_iIn[2:0]TX retransitions (7 -> infinite)CAN InterfacerxInCAN RX (of serial interface)txOut (2.0)TX error countertec_oOut (8:0)TX error countertec_oOutTX warning flaginc_oOutRX warning flaginc_oOutCAN bus opficieevents_oOutCAN bus opficieevents_oOutCAN bus opficieram_tx_string_oOutTX error counterrec_oOut(8:0)TX error countertec_oOut(8:0)RX error counterrec_oOut(8:0)RX error counterrec_			RX Buffer write enable			
ram_rx_data_iIn[31:0]RX Buffer data inputram_rx_data_oOut[31:0]RX Buffer data outputram_rx_rdy_iInRX Buffer ready inputTX Buffer IFram_tx_ena_oOutTX Buffer enableram_tx_wr_oOutTX Buffer write enableram_tx_addr_oOut[9:0]TX Buffer data inputram_tx_data_iIn[31:0]TX Buffer data inputram_tx_data_oOut[10]TX Buffer data outputram_tx_data_oOut[31:0]TX Buffer data outputram_tx_rdy_iInTX Buffer ready inputControl Interfacerx_receive_done_oOutRX received message (without any error)rx_start_iInTX start request to send a frametx_stop_iInTX stop requesttx_stop_iInTX stop requesttx_retransmit_max_iIn[2:0]TX retransitions (7 -> infinite)CAN InterfacerxInCAN RX (of serial interface)txOutCAN RX (of serial interface)txOutTX warning flagrec_oOut[8:0]TX error countertec_oOutCAN bus offrec_oOutCAN bus offervor_passive_oOutCAN bus passiveevents_oOutCAN bus passiveevents_oOutCAN bus passiverecoOutError counterrecoOutCAN bus passiverecoOutCAN bus passiverecoOutCAN bus passive </td <td></td> <td></td> <td></td>						
ram_rx_data_oOut[31:0]RX Buffer data outputram_rx_rdy_iInRX Buffer data outputTX Buffer IFram_tx_ena_oOutTX Buffer enableram_tx_wr_oOutTX Buffer write enableram_tx_data_iIn[31:0]TX Buffer addressram_tx_data_oOut[31:0]TX Buffer data inputram_tx_data_oOut[31:0]TX Buffer ready inputcontrol InterfaceTX Buffer ready inputrx_receive_done_oOutRX received message (without any error)rx_pending_oOutRX start request to send a frame)tx_start_iInTX stop requesttx_start_iInTX received for serial interface)tx_start_iInTX stop requesttx_start_iInTX stop requesttx_start_iInTX retransitions (7 -> infinite)CAN InterfaceInCAN RX (of serial interface)txOutCAN RX (of serial interface)txOutCAN RX (of serial interface)txOutCAN RX (of serial interface)txOutCAN RX (of serial interface)tcc_oOut[8:0]TX error countertec_oOutRX error counterrec_oOutCAN bus passiveerror_passive_oOutCAN bus passiveevents_oOutCAN bus passiveevents_oOutCAN bus passive						
ram_rx_rdy_iInRX Buffer ready inputTX Buffer IFram_tx_ena_oOutTX Buffer enableram_tx_wr_oOutTX Buffer write enableram_tx_addr_oOut[9:0]TX Buffer addressram_tx_data_iIn[31:0]TX Buffer data inputram_tx_data_oOut[31:0]TX Buffer data outputram_tx_data_oOut[31:0]TX Buffer ready inputcontrol Interfacerx_receive_done_oOutrx_start_iInTX start request to send a frame)tx_start_iInTX stop requesttx_stop_iInTX retransitions (7 -> infinite)CAN InterfacerxInrxInCAN RX (of serial interface)tx_stop_iInCAN RX (of serial interface)tx_stop_iInCAN RX (of serial interface)tx_retransmit_max_iInCAN RX (of serial interface)txOutCAN RX (of serial interface)txOutCAN RX (of serial interface)tx_oOutTX warning flagrec_oOut[8:0]TX error countertec_warning_oOutRX error counterrec_oOut[7:0]RX error counterrec_oOut[6:0]Error/warning flagrec_oOutCAN bus offerror_passive_oOutCAN bus passiveevents_oOutCAN bus passiveevents_oOut[6:0]Error/warning event flags						
TX Buffer IFram_tx_ena_0OutTX Buffer enableram_tx_wr_0Out[9:0]TX Buffer write enableram_tx_addr_0Out[9:0]TX Buffer addressram_tx_data_iIn[31:0]TX Buffer data inputram_tx_data_oOut[31:0]TX Buffer data outputram_tx_rdy_iInTX Buffer ready inputControl Interfacerx_receive_done_0OutRX received message (without any error)rx_start_iInTX start request to send a frame)tx_start_iInTX stop requesttx_start_iInTX pending (during sending a frame)tx_retransmit_max_iIn[2:0]TX retransitions (7 -> infinite)CAN InterfacerxInCAN RX (of serial interface)txOutCAN RX (of serial interface)txOutTX error countertec_oOut[8:0]TX error countertec_oOut[7:0]RX error counterrec_warning_oOutCAN bus offerror_passive_oOutCAN bus offeuror_sasive_oOutCAN bus offeuror_sasive_oOutCAN bus offeuror_oOutCAN bus offeuror_sasive_oOutCAN bus offeuror_sasive_oOutCAN bus offeuror_assive_oOutCAN bus offeuror_bassiveError/warning event flags						
ram_tx_ena_oOutTX Buffer enableram_tx_wr_oOutTX Buffer write enableram_tx_addr_oOut[9:0]TX Buffer addressram_tx_data_iIn[31:0]TX Buffer data inputram_tx_data_oOut[31:0]TX Buffer data outputram_tx_rdy_iInTX Buffer ready inputControl InterfaceTXrx_receive_done_oOutRX received message (without any error)rx_start_iInTX start request to send a frame)tx_start_iInTX stop requesttx_stop_iInTX retransitions (7 -> infinite)CAN InterfaceInCAN RX (of serial interface)rxOutCAN RX (of serial interface)txOutCAN TX (of serial interface)txOutCAN RX (of serial interface)tcc_oOut[8:0]TX error countertec_oOut[7:0]RX error counterrec_warning_oOutRX warning flagbus_off_oOutCAN bus offerror_passive_oOutCAN bus offerror_passive_oOutCAN bus offerror_warning_oOutCAN bus offerror_passive_oOutCAN bus offerror_passive_oOut<						
ram_tx_wr_oOutTX Buffer write enableram_tx_addr_oOut[9:0]TX Buffer addressram_tx_data_iIn[31:0]TX Buffer data inputram_tx_data_oOut[31:0]TX Buffer data outputram_tx_rdy_iInTX Buffer ready inputControl InterfaceTxrx_receive_done_oOutRX received message (without any error)rx_start_iInTX start request to send a frametx_start_iInTX storp requesttx_start_iInTX storp requesttx_retransmit_max_iIn[2:0]TX retransitions (7 -> infinite)CAN InterfaceTXOutrxOutCAN RX (of serial interface)txUtCAN TX (of serial interface)txOutTX warning flagtec_oOutTX error countertec_warning_oOutTX warning flagoutTX warning flagtec_oOutRX error counterrec_warning_oOutRX error counterecwarning_oOutCAN bus offerror_passive_oOutCAN bus offerror_passive_oOutCAN bus offerror_passive_oOutCAN bus offerror_bassiveOutCAN bus offerror_passive_oOutCAN bus offerror_passive_oOutCAN bus offerror_bassiveError/warning event flagsTestmodeError/warning event flags		Out	TX Buffer enable			
$ram_tx_addr_o$ Out[9:0]TX Buffer address $ram_tx_adata_i$ In[31:0]TX Buffer data input $ram_tx_data_o$ Out[31:0]TX Buffer data output $ram_tx_rdy_i$ InTX Buffer ready inputControl InterfaceInTX Buffer ready input $rx_receive_done_o$ OutRX received message (without any error) $rx_start_i$ InTX start request to send a frame) $tx_start_i$ InTX stop request $tx_start_i$ InTX stop request $tx_pending_o$ OutTX pending (during sending a frame) $tx_start_i$ InTX stop request $tx_retransmit_max_i$ In[2:0]TX retransitions (7 -> infinite)CAN InterfaceInCAN RX (of serial interface) $tx$ OutCAN RX (of serial interface) $tx_co_o$ Out[8:0]TX error counter $tec_o$ OutTX warning flag $tec_o$ OutRX warning flag $tec_warning_o$ OutCAN bus off $error_passive_o$ OutCAN bus passive $events_o$ OutCAN bus passive						
ram_tx_data_iIn[31:0]TX Buffer data inputram_tx_data_oOut[31:0]TX Buffer data outputram_tx_rdy_iInTX Buffer ready inputControl Interfacerx_receive_done_oOutRX received message (without any error)rx_spending_oOutRX pending (during receiving a frame)tx_start_iInTX start request to send a frametx_start_iInTX stop requesttx_stop_iInTX pending (during sending a frame)tx_retransmit_max_iIn[2:0]TX retransitions (7 -> infinite)CAN InterfaceInCAN RX (of serial interface)txOutCAN RX (of serial interface)txOutCAN RX (of serial interface)tcc_oOut[8:0]TX error countertec_oOutTX warning flagrec_oOutRX warning flagrec_oOutCAN bus offerror_passive_oOutCAN bus passiveevents_oOut[6:0]Error/warning event flags						
ram_tx_data_oOut[31:0]TX Buffer data outputram_tx_rdy_iInTX Buffer ready inputControl InterfaceTX Buffer ready inputrx_receive_done_oOutRX received message (without any error)rx_pending_oOutRX pending (during receiving a frame)tx_start_iInTX start request to send a frametx_stop_iInTX stop requesttx_pending_oOutTX pending (during sending a frame)tx_stop_iInTX stop requesttx_retransmit_max_iIn[2:0]TX retransitions (7 -> infinite)CAN InterfaceTXOutrxInCAN RX (of serial interface)txOutCAN RX (of serial interface)txOutCAN TX (of serial interface)tec_oOut[8:0]TX error countertec_oOut[7:0]RX error counterrec_oOut[7:0]RX error counterrec_oOut[7:0]RX error counterrec_warning_oOutCAN bus offevents_oOut[6:0]Error/warning event flagsTestmodeFrestmodeFrestmode						
ram_tx_rdy_iInTX Buffer ready inputControl Interfacerx_receive_done_oOutRX received message (without any error)rx_pending_oOutRX pending (during receiving a frame)tx_start_iInTX start request to send a frametx_stop_iInTX stop requesttx_pending_oOutTX pending (during sending a frame)tx_stop_iInTX stop requesttx_pending_oOutTX pending (during sending a frame)tx_retransmit_max_iIn[2:0]TX retransitions (7 -> infinite)CAN InterfaceTXrxInCAN RX (of serial interface)txOutCAN TX (of serial interface)txOutCAN TX (of serial interface)trec_oOut[8:0]TX error countertec_oOut[7:0]RX error counterrec_oOut[7:0]RX error counterrec_oOut[7:0]RX error counterrec_oOutCAN bus offeuror_passive_oOutCAN bus passiveevents_oOut[6:0]Error/warning event flags						
Control Interfacerx_receive_done_oOutRX received message (without any error)rx_pending_oOutRX pending (during receiving a frame)tx_start_iInTX start request to send a frametx_stop_iInTX stop requesttx_pending_oOutTX pending (during sending a frame)tx_retransmit_max_iIn[2:0]TX retransitions (7 -> infinite)CAN InterfacerxInCAN RX (of serial interface)tx_OutCAN RX (of serial interface)tx_counters and Statestec_oOut[8:0]TX error countertec_warning_oOutTX warning flagrec_oOut[7:0]RX error counterrec_warning_oOutCAN bus offerror_passive_oOutCAN bus passiveevents_oOut[6:0]Error/warning event flagsTestmode						
rx_receive_done_oOutRX received message (without any error)rx_pending_oOutRX pending (during receiving a frame)tx_start_iInTX start request to send a frametx_stop_iInTX stop requesttx_pending_oOutTX pending (during sending a frame)tx_retransmit_max_iIn[2:0]TX retransitions (7 -> infinite)CAN InterfaceTXrxInCAN RX (of serial interface)tx_vero Counters and StatesTX error countertec_oOutTX varning flagrec_oOutRX error counterrec_oOutRX error counterrec_oOutRX warning flagbus_off_oOutCAN bus offerror_passive_oOutCAN bus passiveevents_oOut[6:0]Error/warning event flagsTestmodeError/warning event flags		[				
rx_pending_oOutRX pending (during receiving a frame)tx_start_iInTX start request to send a frametx_stop_iInTX stop requesttx_pending_oOutTX pending (during sending a frame)tx_retransmit_max_iIn[2:0]TX retransitions (7 -> infinite)CAN InterfacerxOutrxInCAN RX (of serial interface)txOutCAN RX (of serial interface)txOutCAN TX (of serial interface)tror Counters and Statestec_otec_oOut[8:0]TX error countertec_varning_oOutTX warning flagrec_oOut[7:0]RX error counterrec_warning_oOutCAN bus offerror_passive_oOutCAN bus passiveevents_oOut[6:0]Error/warning event flagsTestmode		Out	RX received message (without any error)			
tx_start_iInTX start request to send a frametx_stop_iInTX stop requesttx_pending_oOutTX pending (during sending a frame)tx_retransmit_max_iIn[2:0]TX retransitions (7 -> infinite)CAN InterfacerxInCAN RX (of serial interface)txOutCAN RX (of serial interface)txOutCAN RX (of serial interface)tror Counters and Statestec_oOut[8:0]TX error countertec_oOut[7:0]RX error counterrec_oOut[7:0]RX error counterrec_oOutCAN bus offerror_passive_oOutCAN bus passiveevents_oOut[6:0]Error/warning event flagsTestmode		Out				
tx_stop_iInTX stop requesttx_pending_oOutTX pending (during sending a frame)tx_retransmit_max_iIn[2:0]TX retransitions (7 -> infinite)CAN InterfacerxInCAN RX (of serial interface)txOutCAN RX (of serial interface)txOut (CAN TX (of serial interface)tror Counters and Statestec_oOut[8:0]TX error countertec_oOut (TX warning flagtec_oOut (CAN bus offerror_passive_oOut (CAN bus offervents_oOut (CAN bus passiveevents_oOut (CAN bus passiveevents_oOut (CAN bus passiveevents_oOut (CAN bus passiveError/warning event flags						
tx_pending_oOutTX pending (during sending a frame)tx_retransmit_max_iIn[2:0]TX retransitions (7 -> infinite)CAN InterfacerxInCAN RX (of serial interface)txOutCAN TX (of serial interface)txOutCAN TX (of serial interface)Error Counters and Statestec_oOut[8:0]TX error countertec_warning_oOutTX warning flagrec_oOut[7:0]RX error counterrec_warning_oOutCAN bus offbus_off_oOutCAN bus passiveevents_oOut[6:0]Error/warning event flagsTestmode		In				
tx_retransmit_max_iIn[2:0]TX retransitions (7 -> infinite)CAN InterfacerxInCAN RX (of serial interface)txOutCAN TX (of serial interface)Error Counters and Statetec_oOut[8:0]TX error countertec_warning_oOutTX warning flagrec_oOut[7:0]RX error counterrec_warning_oOutCAN bus offbus_off_oOutCAN bus offerror_passive_oOutCAN bus passiveevents_oOut[6:0]Error/warning event flags						
CAN InterfacerxInCAN RX (of serial interface)txOutCAN TX (of serial interface)Error Counters and Statestec_oOut[8:0]TX error countertec_warning_oOutTX warning flagrec_oOut[7:0]RX error counterrec_warning_oOutRX warning flagbus_off_oOutCAN bus offerror_passive_oOutCAN bus passiveevents_oOut[6:0]Error/warning event flags		In[2:0]				
txOutCAN TX (of serial interface)Error Counters and Statestec_oOut[8:0]TX error countertec_warning_oOutTX warning flagrec_oOut[7:0]RX error counterrec_warning_oOutRX warning flagbus_off_oOutCAN bus offerror_passive_oOutCAN bus passiveevents_oOut[6:0]Error/warning event flagsTestmode						
txOutCAN TX (of serial interface)Error Counters and Statestec_oOut[8:0]TX error countertec_warning_oOutTX warning flagrec_oOut[7:0]RX error counterrec_warning_oOutRX warning flagbus_off_oOutCAN bus offerror_passive_oOutCAN bus passiveevents_oOut[6:0]Error/warning event flagsTestmode	rx	In	CAN RX (of serial interface)			
Error Counters and Statestec_oOut[8:0]TX error countertec_warning_oOutTX warning flagrec_oOut[7:0]RX error counterrec_warning_oOutRX warning flagbus_off_oOutCAN bus offerror_passive_oOutCAN bus passiveevents_oOut[6:0]Error/warning event flagsTestmodeImage: State		Out				
tec_warning_oOutTX warning flagrec_oOut[7:0]RX error counterrec_warning_oOutRX warning flagbus_off_oOutCAN bus offerror_passive_oOutCAN bus passiveevents_oOut[6:0]Error/warning event flagsTestmode	Error Counters and Stat					
rec_oOut[7:0]RX error counterrec_warning_oOutRX warning flagbus_off_oOutCAN bus offerror_passive_oOutCAN bus passiveevents_oOut[6:0]Error/warning event flagsTestmode	tec_o	Out[8:0]	TX error counter			
rec_warning_o       Out       RX warning flag         bus_off_o       Out       CAN bus off         error_passive_o       Out       CAN bus passive         events_o       Out[6:0]       Error/warning event flags         Testmode       Image: Comparison of the second se	tec_warning_o	Out				
rec_warning_o       Out       RX warning flag         bus_off_o       Out       CAN bus off         error_passive_o       Out       CAN bus passive         events_o       Out[6:0]       Error/warning event flags         Testmode       Image: Comparison of the second se		Out[7:0]				
bus_off_o       Out       CAN bus off         error_passive_o       Out       CAN bus passive         events_o       Out[6:0]       Error/warning event flags         Testmode       Vertice	rec_warning_o	Out	RX warning flag			
error_passive_o     Out     CAN bus passive       events_o     Out[6:0]     Error/warning event flags       Testmode     Image: Canadian Content flags		Out				
events_o     Out[6:0]     Error/warning event flags       Testmode		Out	CAN bus passive			
Testmode		Out[6:0]				
testmode_n In Testmode (loopback, internally tx->rx)						
	testmode_n	In	Testmode (loopback, internally tx->rx)			

### Core interface double buffers for transmit and receive

The CAN FD IP interface with double buffers is defined in following table. There is already the double buffer mechanism implemented. The RAM Blocks are single port RAMs, this means that during the



transfer of a frame from the first buffer, only the second buffer can be written and read from the outside.

Signal name	Туре	Description			
clk	In	System clock			
reset n	In	Reset input			
Synchronization Nomina					
tseg1_i	In[5:0]	Time segment 1 nominal			
tseg2_i	In[4:0]	Time segment 2 nominal			
sjw_i	In[3:0]	Sample jump width nominal			
prescale_i	In[4:0]	Prescaling factor nominal			
Synchronization Data	•				
tseg1_dat_i	In[5:0]	Time segment 1 data			
tseg2_dat _i	In[4:0]	Time segment 2 data			
sjw_dat _i	In[3:0]	Sample jump width data			
prescale_dat _i	In[4:0]	Prescaling factor data			
Delay Compensation					
delay_comp_en_i	In	Delay compensation enable			
delay_comp_off_i	In[6:0]	Delay compensation offset			
delay_comp_pos_o	Out[7:0]	Delay compensation position			
delay_comp_en_vld_o	Out	Delay compensation enable valid (internal proofe)			
RX Buffer IF					
ram_rx_ena_i	In	RX Buffer enable			
 ram_rx_wr_i	In	RX Buffer write enable			
ramrx_addr_i	In[9:0]	RX Buffer address			
ram_rx_data_i	In[31:0]	RX Buffer data input			
ram_rx_data_o	Out[31:0]	RX Buffer data output			
ram_rx_rdy_o	Out	RX Buffer ready input			
TX Buffer IF					
ram_tx_ena_i	In	TX Buffer enable			
ram_tx_wr_i	In	TX Buffer write enable			
ram_tx_addr_i	In[9:0]	TX Buffer address			
ram_tx_data_i	In[31:0]	TX Buffer data input			
ram_tx_data_o	Out[31:0]	TX Buffer data output			
ram_tx_rdy_o	Out	TX Buffer ready input			
Control Interface					
rx_receive_done_o	Out	RX received message (without any error)			
rx_pending_o	Out	RX pending (during receiving a frame)			
tx_start_i	In	TX start request to send a frame			
tx_stop_i	In	TX stop request			
tx_pending_o	Out	TX pending (during sending a frame)			
tx_retransmit_max_i	In[2:0]	TX retransitions (7 -> infinite)			
CAN Interface					
rx	In	CAN RX (of serial interface)			
tx	Out	CAN TX (of serial interface)			
Error Counters and Stat	es				
tec_o	Out[8:0]	TX error counter			
tec_warning_o	Out	TX warning flag			
rec_o	Out[7:0]	RX error counter			
rec_warning_o	Out	RX warning flag			
bus_off_o	Out	CAN bus off			

CAN Flexible Data-Rate Controller IP Core – Datasheet

V1.0, February 20<sup>th</sup> 2025



error_passive_o	Out	CAN bus passive			
events_o	Out[6:0]	Error/warning event flags			
Testmode					
testmode_n	In	Testmode (loopback, internally tx->rx)			

#### BUS / AMBA AXI interface for access via internal SoC bus

The CAN FD IP bus interface is defined in following table. Each configuration and control are reachable through the bus interface. In the next table is shown a proprietary bus interface, but there is also an AXI bus interface available.

Signal name	Туре	Description		
clk	In	System clock		
reset_n	In	Reset input		
BUS Interface				
bus_ena_i	In	RX Buffer enable		
bus_wr_i	In	RX Buffer write enable		
bus_addr_i	In[9:0]	RX Buffer address		
bus_data_i	ln[31:0]	RX Buffer data input		
bus_data_o	Out[31:0]	RX Buffer data output		
bus_rdy_o	Out	RX Buffer ready input		
CAN Interface				
rx	In	CAN RX (of serial interface)		
tx	Out	CAN TX (of serial interface)		

The following register definition is implemented. The purple cells are the registers and the white cells are the fields of the register above.

Register name	Addr	Description
- Field name	Bits	Description
CAN_TX_SID	0	CAN TX Standard ID Register
- SID	[28:0]	Standard Identifier + Extended ID
CAN_TX_DLC	1	CAN TX DLC Register
- DLC	[3:0]	DLC (Data Length; 0 -> 0; 1 -> 1; 8 -> 8; 9 -> 12;
		10 -> 16; 11 -> 20; 12 -> 24; 13 -> 32; 14 -> 48; 15 -
		> 64)
- FORMAT	[10:8]	Frame format (000b CBFF; 100b CEFF; 010b
		FBFF; 110b FEFF; XLFF not supported)
- FTYP	[11]	RTR ('1' RF; '0' DF; remote frame selector)
- BRS	[12]	Bit rate switch (only FBFF and FEFF)
- ESI	[13]	Error state indicator (only FBFF and FEFF)
CAN_TX_BUFFER	[2:17]	Transmit Buffer
- DATA	[31:0]	Payload data
CAN_RX_SID	18	CAN RX Standard ID Register
- SID	[28:0]	Standard Identifier + Extended ID
CAN_TX_DLC	19	CAN TX DLC Register
- DLC	[3:0]	DLC (Data Length; 0 -> 0; 1 -> 1; 8 -> 8; 9 -> 12;
		10 -> 16; 11 -> 20; 12 -> 24; 13 -> 32; 14 -> 48; 15 -
		> 64)
- FORMAT	[10:8]	Frame format (000b CBFF; 100b CEFF; 010b
		FBFF; 110b FEFF; XLFF not supported)
- FTYP	[11]	RTR ('1' RF; '0' DF; remote frame selector)



- BRS	[12]	Bit rate switch (only FBFF and FEFF)
- ESI	[12]	Error state indicator (only FBFF and FEFF)
CAN_TX_BUFFER	[20:35]	Transmit Buffer
- DATA	[31:0]	Payload data
CAN_CONTROL	36	CAN Control Register
- TX_SEND_REQ	[0]	TX send request
- TX ABORD	[1]	TX abord
- TX BUSY	[2]	TX busy / pending
- TX_DELAY_COMPENSATION_EN	[3]	TX delay compensation (will do delay compensation
	[-]	only if CAN_CONFIG_DATA.BRP <= 1)
- RX_BUSY	[4]	RX busy
- RX_RECEIVED	[5]	RX received (should be cleared to receive next
		message! To clear, write '0'; if still pending '1' a
		seconded message already received)
- RX_RECEIVEDIE	[6]	RX received interrupt enable
- RX_RECEIVEDIF	[7]	RX received interrupt flag (clear by writing '0')
- RETRANSMIT_MAX	[10:8]	Max Number of Retransmission (0 no
		retransmission; 7 unlimited retransmission)
- RX_OVERRUN	[30]	RX received frame overrun (Buffer full), will be
		cleared together with RX_RECEIVED
- LOOPBACK	[31]	Enable loopback mode (RX = TX, without tranceiver)
	37	Nominal Bit Time Configuration Register
- SJW	[3:0]	Synchronization Jump Width bits
- TSEG1	[13:8]	Time Segment 1 bits (Propagation Segment + Phase
		Segment 1)
- TSEG2	[20:16]	Time Segment 2 bits (Phase Segment 2)
- BRP	[28:24]	Baud Rate Prescaler bits
CAN_CONFIG_DATA	38	Data Bit Time Configuration Register
- SJW	[3:0]	Synchronization Jump Width bits
- TSEG1	[13:8]	Time Segment 1 bits (Propagation Segment + Phase Segment 1)
- TSEG2	[20:16]	
- BRP	[28:24]	Baud Rate Prescaler bits
CAN_DELAY_COMP	39	CAN Delay Compensation (for FD Frames)
- OFFSET	[6:0]	Delay compensation offset (suggestion to use TSEG1
		of data rate)
- POSITION	[15:8]	Delay compensation position (= measured delay +
		OFFSET)
CAN_REC	40	CAN Transmit/Receive Error Count Register
- REC	[7:0]	Receive Error Counter bits
- TEC	[16:8]	Transmit Error Counter bits
- EWARN	[17]	Transmitter or Receiver is in Error Warning State bit
- RXWARN	[18]	Receiver in Error Warning State bit (128>REC>95)
- TXWARN	[19]	Transmitter in Error Warning State bit (128>REC>95)
- RXBP	[20]	Receiver in Error Passive State bit (REC>127)
- TXBP	[21]	Transmitter in Error Passive State bit (TEC>127)
- TXBO	[22]	Transmitter in Bus Off State bit (TEC>255)
- EVENTS	[31:24]	Indicating internal conditions and events (Bit 0:
		Acknowledgement Error; Bit 1: Bit Error; Bit 2: Stuff
		Error; Bit 3: Form Error; Bit 4: CRC Error; Bit 5:
	[44] Z	Overload Condition; Bit 6: Lost Arbitration)
CAN_FILTER	[41:x]	CAN Filter Register

CAN Flexible Data-Rate Controller IP Core – Datasheet



V1.0, February 20<sup>th</sup> 2025

- SID	[28:0]	Standard Identifier + Extended ID
- FORMAT	[31:29]	Frame format
CAN_FILTER_MASK	[x:y]	CAN Filter Mask Register
- SID	[28:0]	Standard Identifier + Extended ID
- FORMAT	[31:29]	Frame format
CAN_FILTER_CTRL	[y:z]	CAN Filter Control Register
- ENABLE	[0]	Filter enable
- CANFLT_IE	[1]	Filter interrupt enable
- CANFLT_IF	[2]	Interrupt flag (clear by writing '0')

## Testing

The testcases are defined to the ISO 11898-1 requirements.

# Deliveries

- VHDL RTL code or netlist for each technology
- Testbench

# Sales & Marketing Contact

TES Electronic Solutions GmbH Nikolaus-Otto-Straße 25 70771 Leinfelden-Echterdingen Germany sales@tes-dst.com www.tes-dst.com