# CAN-CTRL

## CAN 2.0 & CAN FD Bus Controller Core

Implements a CAN bus controller that performs serial communication according to CAN 2.0, and CAN FD specifications. It supports the original Bosch protocol and ISO specifications as defined in ISO 1989—including time-triggered operation (TTCAN) as specified in ISO 19898-4—and is also optimized to support the popular AUTOSAR and SAE J1939 specifications.

The CAN protocol uses a multi-master bus configuration for the transfer of frames between nodes of the network and manages error handling with no burden on the host processor. The core enables the user to set up economic and reliable links between various components. It appears as a memory-mapped I/O device to the host processor, which accesses the CAN core to control the transmission or reception of frames.

The CAN core is easy to use and integrate, featuring programmable interrupts, data and baud rates; a configurable number of independently programmable acceptance filters; and a generic processor interface or optionally an AMBA APB, or AHB-Lite interface. It implements a flexible buffering scheme, allowing fine-tuning of the core size to satisfy the requirements of each specific application.

The number of receive buffers is synthesis-time configurable. Two types of transmit buffers are implemented: a high-priority primary transmit buffer (PTB) and a lower-priority secondary transmit buffer (STB). The PTB can store one message, while the number of included buffer slots for the STB is synthesis-time configurable. The transmit buffer can operate in FIFO or priority mode. An optional wrapper instantiating multiple CAN controller cores easies integration in cases where multiple bus nodes need to be controlled by the same host processor.

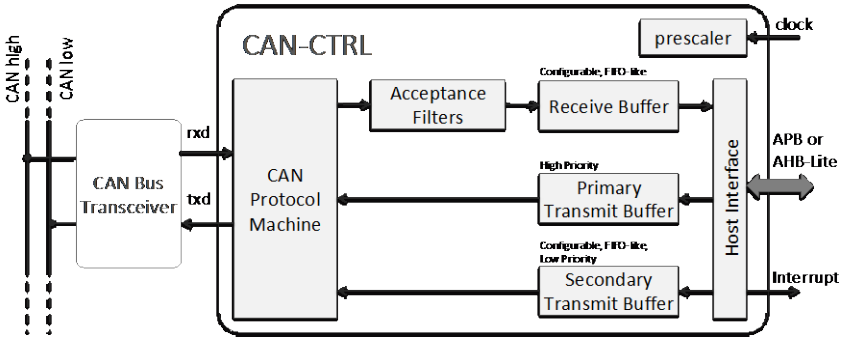
The core implements functionality similar to the Philips SJA1000 working with its PeliCAN mode extensions, providing error analysis, diagnosis, system maintenance, and optimization features.

The CAN core is extensively verified and proven in multiple production designs.

## Applications

The CAN-CTRL core can be integrated in devices that use CAN or higher-layer, CAN-based communication protocols. In addition to traditional automotive applications, such devices are used in industrial (e.g. the CANopen and DeviceNet protocols), aviation (e,g. the ARINC-825 and CANaerospace protocols), marine (e.g. the NMEA 2000 protocol) and other applications.

## Block Diagram




## Features

### CAN Specifications Support

- CAN 2.0 & CAN-FD (ISO 11898-1.2015, plus earlier ISO and Bosch specifications)CAN FD
- TTCAN (ISO 11898-4 level 1)
- Optimized for AUTOSAR and SAE J1939

### Enhanced Functionality

- Error Analysis features enabling diagnostics, system maintenance, and system optimization:
  - Last error type
  - Arbitration lost position
  - Error Warning Limit
- Listen-Only Mode enables CAN bus traffic analysis and automatic bit-rate detection
- Loop back mode for self-testing
- Time-stamping support, compliant to CiA's 603 specification
- Optional ECC memories support

### Flexible Message Buffering and Filtering

- Configurable number of receive buffers
- One high-priority transmit buffer
- Configurable number of lower-priority transmit buffers
- FIFO or priority mode for transmit buffers
- Configurable number of independently programmable 29-bit acceptance filters, 1 to 16

### Easy to Use and Integrate

- Programmable data rate up to 1 Mbit/s with CAN 2.0 and several Mbit/s with CAN FD option
- Programmable baud rate prescaler: 1 up to 1/256
- Single Shot Transmission Mode for lower software overhead and fast reloading of transmit buffer
- Programmable interrupt sources
- Generic 8-bit host-controller interface and optional 32-bit AMBA-APB or 32-bit AHB-Lite
- A single host can control multiple CAN bus nodes via an optional Multi-CAN wrapper

### Zero Risk

- Link to commercial bus drivers (e.g. PCA82C250T by Philips)
- Multiple times production proven

### Efficient and Portable Design

- Available in RTL, and portable to ASIC and FPGA technologies
- Size of approximately 12,000 gates

December 2017

## Functional Description

The CAN bus core is founded on the basic CAN protocol principle and meets all constraints of the CAN 2.0B and CAN FD specifications.

Several message buffers are used for buffering received or transmitted messages. The number of buffers can be selected before synthesis. Selecting a large number of buffers disables the need for real-time reaction to CAN messages for the host processor, which significantly eases software development of the system application.

The included high-priority primary transmit buffer (PTB) can be used to transmit an important message as fast as possible, even if several lower-priority messages are pending. The secondary transmit buffer (STB) operates either in FIFO mode or in priority mode where the order is changed depending on the message priority.

The host interface contains all necessary registers for controlling and configuring the core. The host is able to read and write all registers as conventional RAM in memory mapped mode.

The interface to the host is software configurable. All events on the CAN data bus or in the CAN controller core are signaled using an interrupt. Every interrupt source may be individually enabled or disabled. The CAN controller core contains up to sixteen software-programmable 29-bit acceptance filters that can be used to block unwanted CAN messages, which reduces the load to the host controller.

The host controller interface operates in a different clock domain, which can be synchronous or asynchronous to the core clock. The host-interface type is user selectable at synthesis time, and can be either an 8-bit-wide generic processor interface or a 32-bit-wide APB or AHB-lite interface. This enables easy interfacing to many host controller types, facilitating quick integration with a microcontroller. Furthermore, the width of the internal data path and buffers is adjusted depending on the host interface to ensure efficient utilization of the host bus bandwidth

## Support

The core as delivered is warranted against defects for ninety days from purchase. Thirty days of phone and email technical support are included, starting with the first interaction. Additional maintenance and support options are available.

## Verification

The core has been rigorously verified and has been production proven multiple times.

## Deliverables

The core includes everything required for successful implementation:

- VHDL or Verilog RTL source code
- Post-synthesis EDIF (netlist licenses)
- Testbenches
  - Behavioral tests
  - Post-synthesis verification
- Simulation scripts
- Synthesis scripts
- Linux driver
- Documentation

CAST
info@cast-inc.com
www.cast-inc.com

CAST, Inc. 50 Tice Blvd., Suite 340
Woodcliff Lake, NJ 07677 USA
tel 201-391-8300 fax 201-391-8694



Fraunhofer IPMS

This core is sourced from
Technology Partner Fraunhofer IPMS