

# TFT100 CAN I/O settings

[Main Page](#) > [E-Mobility Trackers](#) > [TFT100](#) > [TFT100 Configuration](#) > **TFT100 CAN I/O settings**



## Contents

- [1 CAN Protocol](#)
  - [1.1 Bosch CAN Powertrain](#)
    - [1.1.1 Bosch CAN Powertrain I/O](#)
  - [1.2 Askoll](#)
    - [1.2.1 Askoll I/O](#)
  - [1.3 Default J1939](#)
    - [1.3.1 Default J1939 I/O](#)
  - [1.4 Manual CAN](#)
    - [1.4.1 Manual CAN I/O](#)
  - [1.5 Manual J1939](#)
    - [1.5.1 Manual J1939 I/O](#)
  - [1.6 FLEX](#)
    - [1.6.1 FLEX I/O](#)
  - [1.7 Debug](#)
- [2 J1939 Settings](#)
- [3 CAN I/O reset](#)

## CAN Protocol



### ATTENTION

CAN interface **does not work** when device does not have external power available.



**CAN I/O** category offers users an ability to select one of CAN protocols to read specific data through TFT100 CAN interface.

TFT100 currently has 6 implemented CAN protocols and DEBUG function:

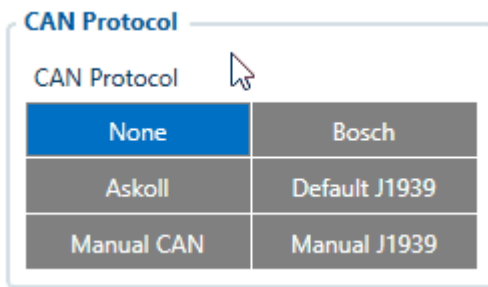
- Bosch CAN Powertrain
- Askoll
- Default J1939
- Manual CAN
- Manual J1939
- FLEX
- Debug feature

**Bosch**, **Askoll**, **Default J1939** and **FLEX** CAN protocols come with specific CAN I/O parameters that can be read, monitored, configured and their values sent to the configured server.

**Manual CAN** and **Manual J1939** protocols do not include dedicated specific I/O parameters however it welcomes user to configure what has to be read from the transport based on their transport CAN communication protocol.

## Bosch CAN Powertrain

---



**Bosch CAN Powertrain protocol** allows users to read parameters from their transport, if the transport uses *Bosch* motor. The protocol is enabled by selecting **Bosch** parameter in *CAN Protocol* field as shown in the image on the right hand side.

### Bosch CAN Powertrain I/O

Specific **Bosch CAN Powertrain** I/O parameters were implemented to our device's firmware that can be read, configured and sent to the server just like regular I/O parameters. The parameters that were added can be seen in the image below.

#### NOTE!

The parameters were added upon clients' request. *Bosch* CAN protocol contains way more parameters that can be added by us if necessary.

### CAN I/O

Input Name	Units	Priority				Low Level	High Level
		None	Low	High	Panic		
Park Brake		None	Low	High	Panic	0	
Selected Charge Mode		None	Low	High	Panic	0	
Charger State		None	Low	High	Panic	0	
Charger Voltage	mV	None	Low	High	Panic	0	
Charger Current	mA	None	Low	High	Panic	0	
Charger Control Mode		None	Low	High	Panic	0	
BMS COM Timeout		None	Low	High	Panic	0	
Charger CRC Violation		None	Low	High	Panic	0	
Charger MC Violation		None	Low	High	Panic	0	
Charger Status		None	Low	High	Panic	0	
Actual Voltage	mV	None	Low	High	Panic	0	
Charger Internal Fault		None	Low	High	Panic	0	
Charger Energy	Wh	None	Low	High	Panic	0	
Charger Current Actual	mA	None	Low	High	Panic	0	
Throttle Position	%	None	Low	High	Panic	0	
Brake Pressed		None	Low	High	Panic	0	
Charge Plugged		None	Low	High	Panic	0	
Kill Switch Active		None	Low	High	Panic	0	
Kickstand Balance Status		None	Low	High	Panic	0	

- The **AVL ID** list of **Bosch CAN Powertrain** parameters can be found in [here](#).
- **SMS/GPRS commands** that were developed for **Bosch CAN Powertrain** protocol can be found in [here](#).

## Askoll

**CAN Protocol**

CAN Protocol	
None	Bosch
Askoll	Default J1939
Manual CAN	Manual J1939

**Askoll** CAN protocol allows users to read parameters from their e-scooters manufactured by *Askoll*. The protocol is enabled by selecting **Askoll** parameter in *CAN Protocol* field as shown in the image on the right hand side.

## Askoll I/O


Specific **Askoll** I/O parameters were implemented to our device's firmware that can be read, configured and sent to the server just like regular I/O parameters. The parameters that were added can be seen in the image below.

## CAN I/O

Input Name	Units	Priority				Low Level	High Level
Charger Voltage	mV	None	Low	High	Panic	0	
Charger Current	mA	None	Low	High	Panic	0	
Malfunction Indication		None	Low	High	Panic	0	
Estimated Range	km	None	Low	High	Panic	0	
SoC Battery	%	None	Low	High	Panic	0	
Remaining Capacity	Ah	None	Low	High	Panic	0	
Total Distance	km	None	Low	High	Panic	0	
Trip Distance	m	None	Low	High	Panic	0	
Vehicle Speed	km/h	None	Low	High	Panic	0	
Ignition Fast Status		None	Low	High	Panic	0	
Helmet Status		None	Low	High	Panic	0	
Top Case sensor		None	Low	High	Panic	0	
Central Stand Up		None	Low	High	Panic	0	
Emergency		None	Low	High	Panic	0	
Over-Under Temperature		None	Low	High	Panic	0	
Regeneration Disabled		None	Low	High	Panic	0	
Battery On/Off		None	Low	High	Panic	0	
Warning UnderVoltage		None	Low	High	Panic	0	
Warning OverVoltage		None	Low	High	Panic	0	

- The **AVL ID** list of **Askoll** parameters can be found in [here](#).
- **SMS/GPRS commands** that were developed for **Askoll** protocol can be found in [here](#).

## Default J1939

 **Default J1939** protocol support was added to TFT100 device's firmware since it is one of the most popular CAN protocols used in heavy duty (i.e. *forklifts*) machinery industry. If the machinery communicates via J1939 CAN protocol, the device will be able to read provided parameter's data and transmit readings to your configured server. To use **Default J1939** CAN protocol user should select **Default J1939** parameter in *CAN Protocol* field as shown in the image on the right hand side.

*Default J1939*  
can be tested  
**NOTE!** on FW  
55.00.09.rev.04  
or newer

To read data with this functionality, the user must have:

- **TFT100 device with CAN interface;**
- **55.00.09.Rev.04 or newer firmware;**
- **Machinery with CAN interface, which communicates via J1939 protocol.**

## Default J1939 I/O

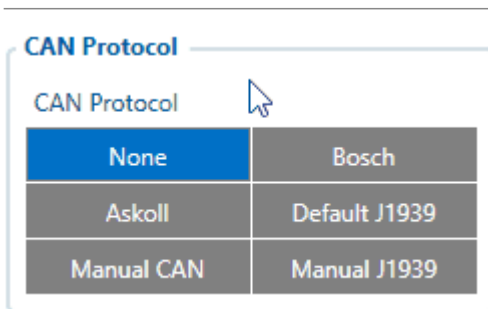
All **Default J1939** CAN I/O parameters can be configured like standard I/O parameters. User can

set priority, high level, low level parameter limits, eventual record, operand, SMS text and number to which SMS notification would be sent.

Input Name	Units	Priority	Low Level	High Level	Event Only	Operand	Send SMS To	SMS Text
Accelerator Pedal 1 Low Idle Switch		None Low High Panic	0	0	Crash Yes No	Monitoring		Accelerator Pedal 1 Low Idle
Accelerator Pedal Kickdown Switch		None Low High Panic	0	0	Crash Yes No	Monitoring		Accelerator Pedal Kickdown
Road Speed Limit Status		None Low High Panic	0	0	Crash Yes No	Monitoring		Road Speed Limit Status
Road Speed Limit Status		None Low High Panic	0	0	Crash Yes No	Monitoring		Accelerator Pedal 2 Low Idle
Accelerator Pedal Position 1	%	None Low High Panic	0	0	Crash Yes No	Monitoring		Accelerator Pedal Position 1
Engine Percent Load At Current Speed	%	None Low High Panic	0	0	Crash Yes No	Monitoring		Engine Percent Load At Curr
Remote Accelerator Pedal Position		None Low High Panic	0	0	Crash Yes No	Monitoring		Remote Accelerator Pedal P
Accelerator Pedal 2 Position	%	None Low High Panic	0	0	Crash Yes No	Monitoring		Accelerator Pedal 2 Position
Vehicle Acceleration Rate Limit Status		None Low High Panic	0	0	Crash Yes No	Monitoring		Vehicle Acceleration Rate Li
Momentary Engine Maximum Power Enable Feedback		None Low High Panic	0	0	Crash Yes No	Monitoring		Momentary Engine Maximu
DPF Thermal Management Active		None Low High Panic	0	0	Crash Yes No	Monitoring		DPF Thermal Management
SCR Thermal Management Active		None Low High Panic	0	0	Crash Yes No	Monitoring		SCR Thermal Management
Actual Maximum Available Engine - Percent Torque	%	None Low High Panic	0	0	Crash Yes No	Monitoring		Actual Maximum Available f
Estimated Pumping - Percent Torque	%	None Low High Panic	0	0	Crash Yes No	Monitoring		Estimated Pumping - Percer
Engine Torque Mode		None Low High Panic	0	0	Crash Yes No	Monitoring		Engine Torque Mode
Actual Engine - Percent Torque (Fractional)	%	None Low High Panic	0	0	Crash Yes No	Monitoring		Actual Engine - Percent Torc
Driver's Demand Engine - Percent Torque	%	None Low High Panic	0	0	Crash Yes No	Monitoring		Driver's Demand Engine - P
Actual Engine - Percent Torque	%	None Low High Panic	0	0	Crash Yes No	Monitoring		Actual Engine - Percent Torc
Engine Speed	rpm	None Low High Panic	0	0	Crash Yes No	Monitoring		Engine Speed

- The AVL ID list of Default J1939 parameters can be found in [here](#).

## Manual CAN



The **Manual CAN** protocol is enabled by selecting **Manual CAN** parameter in *CAN Protocol* field as shown in the image on the right hand side. Then user can configure CAN parameters in [Manual CAN Settings](#) tab.

*Manual CAN* can be tested  
**NOTE!** on FW  
 55.00.09.rev.04  
 or newer

The main benefit, of using **Manual CAN** functionality is that user is able to read data via CAN BUS without requiring additional CAN protocol development from the device's firmware side. To read data with this functionality, the user must have:

- **TFT100 device with CAN interface;**
- **55.00.09.Rev.04 or newer firmware;**
- **Transport with CAN interface;**
- **Transport's CAN communication protocol (with information about frames, parameters, ID's, baud rate).**

## Manual CAN I/O

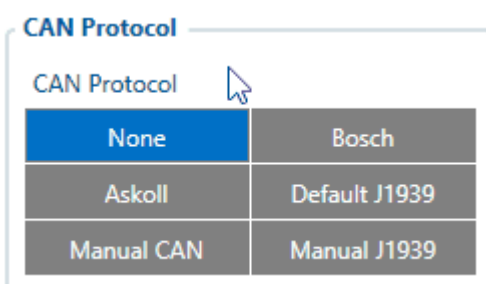
Up to 30 I/O parameters (Priority, Event Only, Operand, Send SMS To, SMS text) for **Manual CAN** functionality can be configured in CAN I/O tab.

It is important to not forget that [Manual CAN Settings](#) tab **must be configured** as well.

Input Name	Units	Priority	Low Level	High Level	Event Only	Operand	Send SMS To	SMS Text
Manual CAN0		None Low High Panic			Crash Yes No	Monitoring		Manual CAN0
Manual CAN1		None Low High Panic			Crash Yes No	Monitoring		Manual CAN1
Manual CAN2		None Low High Panic			Crash Yes No	Monitoring		Manual CAN2
Manual CAN3		None Low High Panic			Crash Yes No	Monitoring		Manual CAN3
Manual CAN4		None Low High Panic			Crash Yes No	Monitoring		Manual CAN4
Manual CAN5		None Low High Panic			Crash Yes No	Monitoring		Manual CAN5
Manual CAN6		None Low High Panic			Crash Yes No	Monitoring		Manual CAN6
Manual CAN7		None Low High Panic			Crash Yes No	Monitoring		Manual CAN7
Manual CAN8		None Low High Panic			Crash Yes No	Monitoring		Manual CAN8
Manual CAN9		None Low High Panic			Crash Yes No	Monitoring		Manual CAN9
Manual CAN10		None Low High Panic			Crash Yes No	Monitoring		Manual CAN10
Manual CAN11		None Low High Panic			Crash Yes No	Monitoring		Manual CAN11
Manual CAN12		None Low High Panic			Crash Yes No	Monitoring		Manual CAN12
Manual CAN13		None Low High Panic			Crash Yes No	Monitoring		Manual CAN13
Manual CAN14		None Low High Panic			Crash Yes No	Monitoring		Manual CAN14
Manual CAN15		None Low High Panic			Crash Yes No	Monitoring		Manual CAN15
Manual CAN16		None Low High Panic			Crash Yes No	Monitoring		Manual CAN16
Manual CAN17		None Low High Panic			Crash Yes No	Monitoring		Manual CAN17
Manual CAN18		None Low High Panic			Crash Yes No	Monitoring		Manual CAN18
Manual CAN19		None Low High Panic			Crash Yes No	Monitoring		Manual CAN19
Manual CAN20		None Low High Panic			Crash Yes No	Monitoring		Manual CAN20
Manual CAN21		None Low High Panic			Crash Yes No	Monitoring		Manual CAN21

- The **AVL ID** list of **Manual CAN** parameters can be found in [here](#).
- Full **Manual CAN** and **Manual CAN Settings** description can be found in [here](#).
- Example on how to configure **Manual CAN** functionality can be found in [here](#).

## Manual J1939



**Manual J1939** protocol support was added to TFT100 device's firmware as an addition to **Default**

**J1939** protocol in case the given **Default J1939** I/O elements are not enough. With this functionality, the user is able to configure which **J1939** parameters he would like to read from the transport that uses **J1939** protocol. **Manual J1939** protocol is enabled by selecting **Manual J1939** parameter in *CAN Protocol* field as shown in the image on the right hand side. Then user can configure CAN parameters in [Manual CAN Settings](#) tab.

**NOTE!** *Manual J1939* can be tested on FW 55.00.09.rev.08 or newer

To read data with this functionality, the user must have:

- **TFT100 device with CAN interface;**
- **55.00.09.Rev.08 or newer firmware;**
- **Machinery with CAN interface, which communicates via J1939 protocol;**
- **Machinery's J1939 protocol documentation (with required SPN's and PGN's).**

### Manual J1939 I/O

Up to 29 I/O parameters (Priority, Event Only, Operand, Send SMS To, SMS text) for **Manual J1939** functionality can be configured in CAN I/O tab the same way as for **Manual CAN** functionality.

**NOTE!** It is important to not forget that [Manual CAN Settings](#) tab **must be configured** as well.

CAN I/O													
Input Name	Units	Priority				Low Level	High Level	Event Only			Operand	Send SMS To	SMS Text
Manual CAN0		None	Low	High	Panic			Crash	Yes	No	Monitoring		Manual CAN0
Manual CAN1		None	Low	High	Panic			Crash	Yes	No	Monitoring		Manual CAN1
Manual CAN2		None	Low	High	Panic			Crash	Yes	No	Monitoring		Manual CAN2
Manual CAN3		None	Low	High	Panic			Crash	Yes	No	Monitoring		Manual CAN3
Manual CAN4		None	Low	High	Panic			Crash	Yes	No	Monitoring		Manual CAN4
Manual CAN5		None	Low	High	Panic			Crash	Yes	No	Monitoring		Manual CAN5
Manual CAN6		None	Low	High	Panic			Crash	Yes	No	Monitoring		Manual CAN6
Manual CAN7		None	Low	High	Panic			Crash	Yes	No	Monitoring		Manual CAN7
Manual CAN8		None	Low	High	Panic			Crash	Yes	No	Monitoring		Manual CAN8
Manual CAN9		None	Low	High	Panic			Crash	Yes	No	Monitoring		Manual CAN9
Manual CAN10		None	Low	High	Panic			Crash	Yes	No	Monitoring		Manual CAN10
Manual CAN11		None	Low	High	Panic			Crash	Yes	No	Monitoring		Manual CAN11
Manual CAN12		None	Low	High	Panic			Crash	Yes	No	Monitoring		Manual CAN12
Manual CAN13		None	Low	High	Panic			Crash	Yes	No	Monitoring		Manual CAN13
Manual CAN14		None	Low	High	Panic			Crash	Yes	No	Monitoring		Manual CAN14
Manual CAN15		None	Low	High	Panic			Crash	Yes	No	Monitoring		Manual CAN15
Manual CAN16		None	Low	High	Panic			Crash	Yes	No	Monitoring		Manual CAN16
Manual CAN17		None	Low	High	Panic			Crash	Yes	No	Monitoring		Manual CAN17
Manual CAN18		None	Low	High	Panic			Crash	Yes	No	Monitoring		Manual CAN18

# FLEX



The **FLEX** protocol is enabled by selecting **FLEX** parameter in *CAN Protocol* field.

## FLEX I/O

Specific **FLEX I/O** parameters were implemented to our device's firmware that can be read, configured and sent to the server just like regular I/O parameters.

- The **AVL ID** list of **FLEX** parameters can be found in [here](#).

Input Name	Units	Priority	Low Level	High Level	Event Only	Operand	Send SMS To	SMS Text
Throttle Position	%	None Low High Panic	0	0	Crash Yes No	Monitoring		Throttle Position
Brake Pressed		None Low High Panic	0	0	Crash Yes No	Monitoring		Brake Pressed
Charge Plugged		None Low High Panic	0	0	Crash Yes No	Monitoring		Charger Plugged
Kill Switch Active		None Low High Panic	0	0	Crash Yes No	Monitoring		Kill Switch Active
Kickstand Release Status		None Low High Panic	0	0	Crash Yes No	Monitoring		Kickstand Release Status
Powertrain State		None Low High Panic	0	0	Crash Yes No	Monitoring		Powertrain State
Malfunction Indication		None Low High Panic	0	0	Crash Yes No	Monitoring		Malfunction Indication
Estimated Range	km	None Low High Panic	0	0	Crash Yes No	Monitoring		Estimated Range
SoC Battery	%	None Low High Panic	0	0	Crash Yes No	Monitoring		SoC Battery
Remaining Capacity	Ah	None Low High Panic	0	0	Crash Yes No	Monitoring		Remaining Capacity
Full Charge Capacity	Ah	None Low High Panic	0	0	Crash Yes No	Monitoring		Full Charge Capacity
Drive Mode		None Low High Panic	0	0	Crash Yes No	Monitoring		Drive Mode
Total Distance	km	None Low High Panic	0	0	Crash Yes No	Monitoring		Total Distance
Vehicle Speed	km/h	None Low High Panic	0	0	Crash Yes No	Monitoring		Vehicle Speed
Ignition Status		None Low High Panic	0	0	Crash Yes No	Monitoring		Ignition Status
Top Case sensor		None Low High Panic	0	0	Crash Yes No	Monitoring		Top Case Sensor
Max Available Power	W	None Low High Panic	0	0	Crash Yes No	Monitoring		Max Available Power
Handlebar Lock		None Low High Panic	0	0	Crash Yes No	Monitoring		Handlebar Lock

## Debug



Debug Protocol is used to read all incoming message through CAN, it works as a CAN reader. By default it will print incoming messages into Terminal, including CAN ID and Data.



When the **DEBUG** protocol is selected, **CAN Log** option will appear. By Enabling **CAN Log** parameter it will store CAN received data and CAN ID in a file.

GNSS Info
GSM Info
I/O Info
Maintenance

**Log / Dump**

Log

Dump

**DOUT**

DOUT 1

OFF

DOUT 2

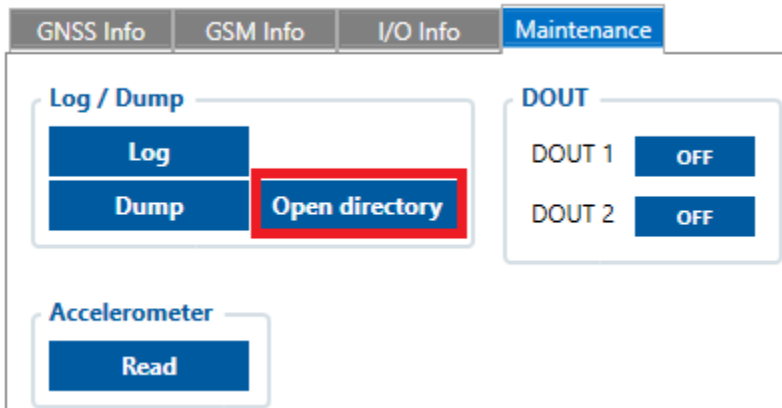
OFF

**Accelerometer**

Read

To download saved file go to Status Tab and Maintenance section, press **“Dump”** button.





After downloading, follow “**Open directory**” button. After unzipping file there will be CAN folder with “0.log” file. Maximum file size is 5MB. After exceeding limit it will rewrite data.

## J1939 Settings



Some J1939 PGN’s are not sent periodically, so they must be requested. In order to request PGN device must be registered to J1939 network. To register device user should configure **J1939 Source Address** and **J1939 Name** parameters:

- **J1939 Source Address** parameter is used to set TFT source address on J1939 network. If device with selected source address already exists on network, user should change it to another address which might be free, or PGN’s which must be requested will not be received.
- **J1939 Name** parameter is used to set TFT name on J1939 network.

## CAN I/O reset



**\* This parameter is available only with FW 55.01.10.Rev.00 or newer**

If CAN IO reset parameter configured as 0 it will keep captured CAN data, otherwise if data is not received anymore it will reset I/Os after configured timeout.

Possible values (in seconds):

**0** - Data will not be cleaned. If external power is removed it will still keep the value.

**1-3600** - Data will be cleaned within timeout period if no more data is received or external power removed.

CAN IO resets in any case after  
**NOTE!** device restart or CAN Protocol switching.



### I/O reset description

- **Disabled** - All I/O's will be reset, when no CAN data received in at least one of I/O's.
- **Enabled** - Only I/O's with Reset timeout \*\* parameter enabled (in Manual CAN Settings

section) and which received no data will be reset.