LANTRONIX®

Application Note

ContiPressureCheck™ System Integration into FOX3-2G/3G/4G Series

Part Number APP-0009
Revision A October 2019

Intellectual Property

Lantronix is a registered trademark of Lantronix, Inc. in the United States and other countries.

Patented: www.lantronix.com/legal/patents/; additional patents pending.

All trademarks and trade names are the property of their respective holders.

Contacts

Lantronix, Inc.
7535 Irvine Center Drive, Suite 100
Irvine, CA 92618, USA
Toll Free: 800-526-8766
Phone: 949-453-3990
Fax: 949-453-3995

Technical Support
Online: www.lantronix.com/support

Sales Offices
For a current list of our domestic and international sales offices, go to the Lantronix web site at www.lantronix.com/about/contact

Disclaimer

All information contained herein is provided “AS IS.” Lantronix undertakes no obligation to update the information in this publication. Lantronix does not make, and specifically disclaims, all warranties of any kind (express, implied or otherwise) regarding title, non-infringement, fitness, quality, accuracy, completeness, usefulness, suitability or performance of the information provided herein. Lantronix shall have no liability whatsoever to any user for any damages, losses and causes of action (whether in contract or in tort or otherwise) in connection with the user’s access or usage of any of the information or content contained herein. The information and specifications contained in this document are subject to change without notice.

Revision History

Date	Rev.	Comments
August 2017	1.0.0	Initial version.
December 2018	2.0.0	Implemented events related to the CPC TTM (Truck tire module “Sensor inside the tire”) data – See chapter 2.4 Implemented dynamic variables related to the CPC TTM (Truck tire module “Sensor inside the tire”) data – See chapter 2.5 Added Figure 3 and Table 5 in chapter 2.5 Added new configuration examples in Table 1, chapter 2.1
October 2019	A	Initial Lantronix document. Added Lantronix document part number, logo, contact information, and links.

For the latest revision of this product document, please check our online documentation at www.lantronix.com/support/documentation.

Table of Contents

1. GENERAL

1.1 Information about this installation guide

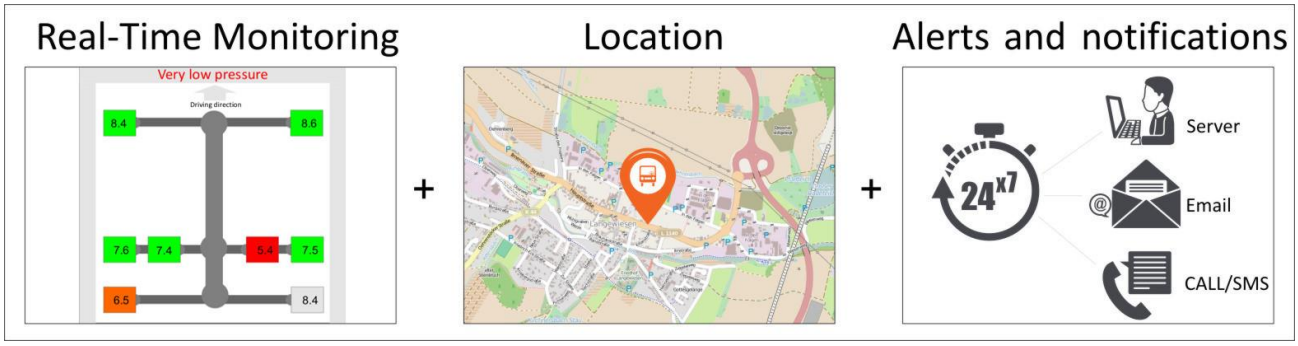
This guide provides information on how to configure a FOX3-2G/3G/4G series device, connect to an in-vehicle installed ContiPressureCheck™ System(CPC-System) via FMS Gateway, read the data from this system and transfer them to your platform server for further evaluation.

1.2 What is ContiPressureCheck™ System ?

The ContiPressureCheck™ system (CPC system) permits continuous monitoring of tire pressure and tire temperature. The status of the tire is shown on the display. In the event of a pressure drops in a tire, the driver immediately receives a corresponding warning. The basic system consists of the central control unit (CCU) and the tire sensors. Each tire sensor on the inside of the tire transmits the measured data via a radio-frequency transmitter to the CCU. The analyzed data are then transmitted via the bus system to the display in the driver's cab. In the event of a deviation from the programmed value for the tires, a warning is immediately sent to the display. The same data, transmitted via the bus system, is also output to the built-in CAN interface of the CPC system. The system can manage up to 24 tire sensors per control unit.

1.3 Why telematics for monitoring the CPC system?

The CPC system installed in a vehicle provides permanent data related to the tire pressure and tire temperature. Integrated with FOX3-2G/3G/4G series telematics device, the system gives you the ability to monitor online the tire pressure and temperature data, as well as receive real-time alerts. When a tire alerts to a problem, the FOX3-2G/3G/4G series telematics device receives such information via the CAN-Bus interface of the CPC system and immediately transmits this data together with the location and speed from the vehicle to your server or via SMS to a phone number. In this way, fleet managers are automatically informed about their on-road vehicles and in case of problems, find suitable repair facilities on the way.



The connection of the FOX3-2G/3G/4G series with the ContiPressureCheck™ system is very simple. It takes place by connecting the CAN-Bus interface of the FOX3-2G/3G/4G series to the FMS-Gateway available in a truck. For more details, refer to the chapter 2.6.

1.4 Hardware Requirements

- A vehicle with an integrated ContiPressureCheck™ system for trucks/tractors
- FOX3-2G series devices with CAN-INTERFACE*, CPC* and firmware 2.16.0 or higher
- FOX3-3G/4G series with CAN-INTERFACE*, CPC* and firmware 3.1.0 or higher
- A SIM card with data service
- Device configuration profile to read the CPC data
- Remote server platform
- Installation and power supply cable for FOX3-2G/3G/4G series devices (CA68)

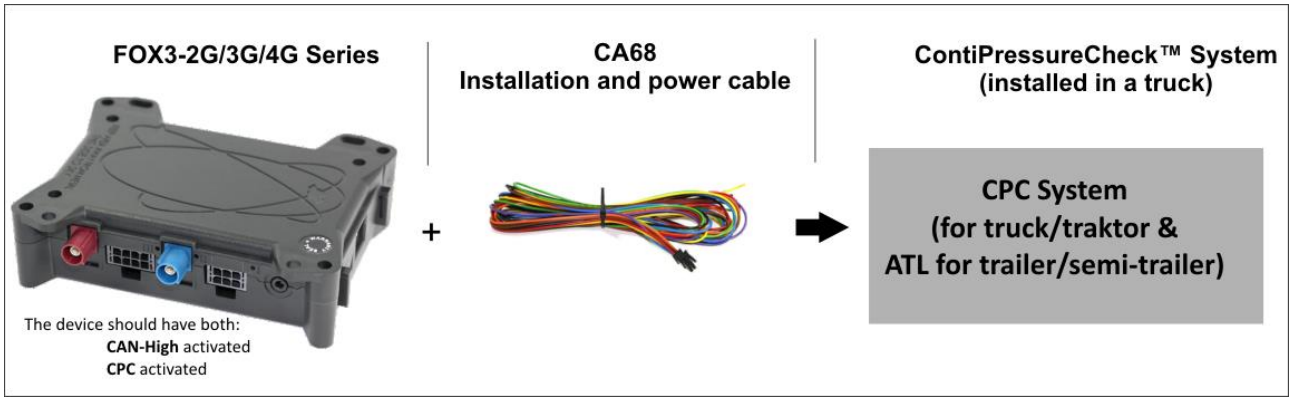


Fig. 1: Requirements for connection of FOX3-2G/3G/4G with ContiPressureCheck™ system

1.5 Main benefits of CPC monitoring data in real time

The main features and benefits of CPC data are:

- ✓ Monitoring tire pressure and temperature data of the entire fleet
- ✓ Managing information collected in real time
- ✓ 24/7 communication between vehicle, driver, and fleet manager
- ✓ Managing the GNSS location of every asset
- ✓ Reducing tire repairs and maintenance costs
- ✓ Decreasing operational costs and increasing on-time deliveries

** Premium features (it needs to be activated before trying to use it: refer to the AppNotes in chapter 0)*

2. READING DATA FROM THE CPC VIA THE FMS GATEWAYFOX3-2G/3G/4G device configuration settings

To read the data from a CPC system via the FMS gateway, you have to first configure your FOX3-2G/3G/4G device by sending/saving the following PFAL script into the FOX3 series device with Workbench software. Refer to the chapter 2.6 and find out how to install the FOX3 device in a truck.

The Workbench software can be downloaded from the Lantronix website (see chapter **Error! Reference source not found.**). After installation on your PC, load the PFAL script below into your device. More information on how to download the Workbench software, install and load the PFAL script can be found in the hardware manual of the device you are using (see chapter **Error! Reference source not found.**):

Enable CAN Interface and FSM to be able to read data from the CAN Bus interface of the ContiPressureCheck™ system:
$PFAL,Sys.Can.Enable,250K,RW $PFAL,Sys.Can.FMS.Enable
Set the device name to better identify your device on your server side.
$PFAL,CNF.Set,DEVICE.NAME=ContiPressureCheck
To read automatically, every 15 seconds, the complete data from the ContiPressureCheck™ system and send them at once together with GNSS data to your server, use the alarm configuration below:
$PFAL,CNF.Set,AL0=SYS.Device.eStart:SYS.TIMER0.start=cyclic,15000 $PFAL,CNF.Set,AL1=SYS.TIMER.e0:TCP.Client.Send,8,"**SPEED km/h:** &(Speed); **Temp/°C:** &(J1939.TIRE_TEMP); **Nominative Tire PRESS/kPa:** &(J1939.TIRE2_NOMPRESS); **TirePressure/kPa:** &(J1939.TIRE_PRESSURE); **Sensor Fault:** &(J1939.TIRE_FAULT); **Tire Pressure Threshold Detection:** &(J1939.TIRE_PTD); **Pressure Loss:** &(J1939.TIRE_PLOSS); **Sensor Enable:** &(J1939.TIRE_SEN); **Tire State:** &(J1939.TIRE_STAT); **ATL:** &(J1939.TIRE_CPC_STAT_LEARN) "
Each time the timer0 with a 15 second timeout expires, the FOX3/-3G/-4G device sends the data to your server in the format as shown below. The text shown in red depends on the settings of the parameter "$PFAL,Cnf.Set,DEVICE.PFAL.SEND.FORMAT". Each line send to the server is terminated by carriage return (CR) and line feed (LF). The conversation of the temperature from Celsius (°C) to Fahrenheit (°F) and the pressure from (k)Pascal to PSI (pound per square inch) has to be done on the server side.
$**SPEED km/h:** 80; **Temp/°C:** 00:20,01:21,10:21,11:21,12:21,13:21; **Nominative Tire PRESS/kPa:**00:183,01:183,10:183,11:183,12:183,13:183; **TirePressure/kPa:** 00:4,01:4,10:4,11:4,12:4,13:4; **Sensor Fault:** 00:0,01:0,10:0,11:0,12:0,13:0; **Tire Pressure Threshold Detection:** 00:4,01:4,10:4,11:4,12:4,13:4; **Pressure Loss:** 00:0,01:0,10:0,11:0,12:0,13:0; **Sensor Enable:** 00:1,01:1,10:1,11:1,12:1,13:1; **Tire State:** 00:0,01:0,10:0,11:0,12:0,13:0; **ATL:** 00:0*D6 $GPRMC,151725.000,A,5040.4088,N,01058.8556,E,0.01,0.00,210817,,*0D $<end>0
When a tire alerts to a problem, use the alarm configuration below to send the tire state together with GNSS data to your server:
$PFAL,CNF.Set,AL2=Sys.eJ1939.TIRE_STAT=0:TCP.Client.Send,8,"TIRE_STAT: OK > Tire PRESS: &(J1939.TIRE_PRESSURE)" $PFAL,CNF.Set,AL3=Sys.eJ1939.TIRE_STAT>0:TCP.Client.Send,8,"J1939.TIRE_STAT: leak > TirePressureThresholdDetection: &(J1939.TIRE_PTD); Tire State: &(J1939.TIRE_STAT)" $PFAL,CNF.Set,AL4=Sys.eJ1939.TIRE_FAULT=0:TCP.Client.Send,8,"TIRE_FAULT: OK > Tire PRESS: &(J1939.TIRE_PRESSURE); Tire State: &(J1939.TIRE_STAT)" $PFAL,CNF.Set,AL5=Sys.eJ1939.TIRE_FAULT>0:TCP.Client.Send,8," J1939.TIRE_FAULT: fault > TirePressureThresholdDetection: &(J1939.TIRE_PTD); Tire State: &(J1939.TIRE_STAT)"
To check from your server if the detected tire sensors are enabled, send the following command to the device
$PFAL,TCP.Client.Send,08,"Sensor Enable: &(J1939.TIRE_SEN)"

More device configuration examples:

$PFAL,cnf.set,AL6=SYS.eJ1939.TIRE_CPC_STAT_HEALTH>=0: TCP.Client.Send,8,"CONTI_Helth (J1939.TIRE_CPC_STAT_HEALTH)"
$PFAL,cnf.set,AL7=SYS.eJ1939.TIRE_CPC_STAT_WEX>=0:TCP.Client.Send,8,"CONTI_WEX &(J1939.TIRE_CPC_STAT_WEX)"
$PFAL,cnf.set,AL8=SYS.eJ1939.TIRE_CPC_STAT_LEARN>=0:TCP.Client.Send,8,"CONTI_Learn &(J1939.TIRE_CPC_STAT_LEARN)"
$PFAL,cnf.set,AL9=SYS.eJ1939.TIRE_CPC_TTM_STATE>=0:TCP.Client.Send,8,"CONTI_TTM_State &(J1939.TIRE_CPC_TTM_STATE)"
$PFAL,cnf.set,AL10=SYS.eJ1939.TIRE_CPC_TTM_ALARM>=0:TCP.Client.Send,8,"CONTI_TTM ALarm &(J1939.TIRE_CPC_TTM_ALARM)"
$PFAL,cnf.set,AL11=SYS.eJ1939.TIRE_CPC_TTM_BAT>=0:TCP.Client.Send,8,"CONTI_BAT &(J1939.TIRE_CPC_TTM_BAT)"
$PFAL,cnf.set,AL12=SYS.eJ1939.TIRE_CPC_TTM_DEFECT>=0:TCP.Client.Send,8,"CONTI_TTM DEFECT &(J1939.TIRE_CPC_TTM_DEFECT)"
$PFAL,cnf.set,AL13=SYS.eJ1939.TIRE_CPC_TTM_LOSE>=0:TCP.Client.Send,8,"CONTI_TTM LOSE &(J1939.TIRE_CPC_TTM_LOSE)"
$PFAL,CNF.Set,AL14=Sys.eJ1939.TIRE_STAT=0:TCP.Client.Send,8," J1939.TIRE_STAT=OK &(J1939.TIRE_STAT) "
$PFAL,CNF.Set,AL15=Sys.eJ1939.TIRE_STAT>0:TCP.Client.Send,8,"J1939.TIRE_STAT=leak &(J1939.TIRE_STAT) "
$PFAL,CNF.Set,AL16=Sys.eJ1939.TIRE_FAULT=0:TCP.Client.Send,8,"J1939.TIRE_FAULT=OK &(J1939.TIRE_FAULT)"
$PFAL,CNF.Set,AL17=Sys.eJ1939.TIRE_FAULT>0:TCP.Client.Send,8,"J1939.TIRE_FAULT=fault &(J1939.TIRE_FAULT)"

Table 1: Device settings for reading and sending real time data from the ContiPressureCheck™ system

2.2 CPC system CAN bus details

The CAN bus of the CPC system follows the standard SAE J1939.

Subject Area	J1939	Sub document
Physical Layer		
Baud rate	250.000 bits/sec	
Maximum nodes	10	J1939-15
Maximum bus length	40 m	
Maximum stub length	3 m (up to 5 m for OBD)	J1939-15
Termination resistor	2 required (each 120 Ω)	
Cable	UN-Shielded Twisted Pair (UTP)	J1939-15
Data Link Layer		
Protocol	CAN 2.0b (29-bit ID)	

Table 2: The CAN bus of the ContiPressureCheck™ system

2.3 Tire identification diagram

This tire identification diagram shows the wheel position of a 3-axle truck/tractor and a 3-axle trailer. The wheel position is numbered as follow:

00 – Front left
01 – Front right
10 – 2nd axle left outer
11 – 2nd axle left inner
12 – 2nd axle right inner
13 – 2nd axle right outer
20 – 3rd axle left
21 – 3rd axle right
30 – Trailer 1st axle left
31 – Trailer 1st axle right
40 – Trailer 2nd axle left
41 – Trailer 2nd axle right
50 – Trailer 3rd axle left
51 – Trailer 3rd axle right

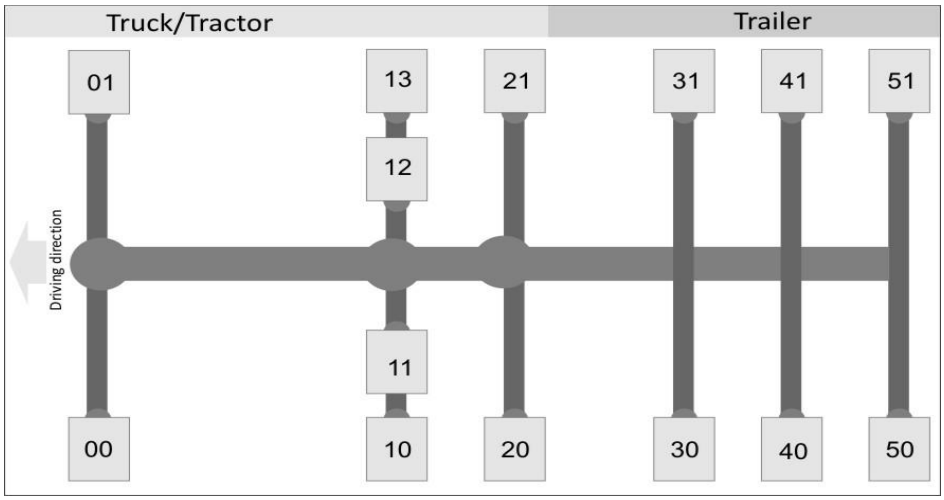


Fig. 2: Tire identification diagram

If there are more than these axles, this same counting continues. Up to 32 sensors are supported. The numbering of wheels, the graphical position and the tire location refer to the data, the telematics device sends to the server. In this way you can easily identify which wheel alerts a problem. Please refer also to the datasheet of the ContiPressureCheck™ system.

2.4 Supported Events/states for ContiPressureCheck™

The following table shows the implemented events for reading real time data from ContiPressureCheck™ system.

Events	Event description
SYS.eJ1939.TIRE_FAULT<comp><value>	This event is generated whenever the contents of the J1939.TIRE_FAULT variable changes. <comp> Compares the contents of this variable with the used specified one and return a Boolean (True/False) representing the result of the comparison. It can be set to =, !=, <,>, <= or>=. <value> Specifies the value to be compared. *Example: SYS.eJ1939.TIRE_FAULT=0 or SYS.eJ1939.TIRE_FAULT>0*
SYS.eJ1939.TIRE_STAT<comp><value>	This event is generated whenever the contents of the SYS.eJ1939.TIRE_STAT changes. <comp> Compares the contents of this variable with the used specified one and return a Boolean (True/False) representing the result of the comparison. It can be set to =, !=, <,>, <= or>=. <value> Specifies the value to be compared. *Example: SYS.eJ1939.TIRE_STAT=0 or SYS.eJ1939.TIRE_STAT>0*
SYS.eJ1939.TIRE_CPC_STAT_HEALTH<comp><value>	This event is generated whenever the contents of the SYS.eJ1939.TIRE_CPC_STAT_HEALTH variable changes. <comp> Compares the contents of this variable with the used specified one and return a Boolean (True/False) representing the result of the comparison. It can be set to =, !=, <,>, <= or>=. <value> Specifies the value to be compared.

	Example: SYS.eJ1939.TIRE_HEALTH>=0
SYS.eJ1939.TIRE_CPC_STAT_WEX<comp><value>	This event is generated whenever the contents of the SYS.eJ1939.TIRE_CPC_STAT_WEX variable changes. <comp> Compares the contents of this variable with the used specified one and return a Boolean (True/False) representing the result of the comparison. It can be set to =, !=, <,>, <= or>=. <value> Specifies the value to be compared. *Example: SYS.eJ1939.TIRE_CPC_STAT_WEX>=0*
SYS.eJ1939.TIRE_CPC_STAT_LEARN<comp><value>	This event is generated whenever the contents of the SYS.eJ1939.TIRE_CPC_STAT_LEARN variable changes. <comp> Compares the contents of this variable with the used specified one and return a Boolean (True/False) representing the result of the comparison. It can be set to =, !=, <,>, <= or>=. <value> Specifies the value to be compared. *Example: SYS.eJ1939.TIRE_CPC_STAT_LEARN>=0*
SYS.eJ1939.TIRE_CPC_TTM_STATE<comp><value>	This event is generated whenever the contents of the SYS.eJ1939.TIRE_CPC_TTM_STATE variable changes. <comp> Compares the contents of this variable with the used specified one and return a Boolean (True/False) representing the result of the comparison. It can be set to =, !=, <,>, <= or>=. <value> Specifies the value to be compared. *Example: SYS.eJ1939.TIRE_CPC_TTM_STATE>=0*
SYS.eJ1939.TIRE_CPC_TTM_ALARM<comp><value>	This event is generated whenever the contents of the SYS.eJ1939.TIRE_CPC_TTM_ALARM variable changes. <comp> Compares the contents of this variable with the used specified one and return a Boolean (True/False) representing the result of the comparison. It can be set to =, !=, <,>, <= or>=. <value> Specifies the value to be compared. *Example: SYS.eJ1939.TIRE_CPC_TTM_ALARM>=0*
SYS.eJ1939.TIRE_CPC_TTM_BAT<comp><value>	This event is generated whenever the contents of the SYS.eJ1939.TIRE_CPC_TTM_BAT variable changes. <comp> Compares the contents of this variable with the used specified one and return a Boolean (True/False) representing the result of the comparison. It can be set to =, !=, <,>, <= or>=. <value> Specifies the value to be compared. *Example: SYS.eJ1939.TIRE_CPC_TTM_BAT>=0*
SYS.eJ1939.TIRE_CPC_TTM_DEFECT<comp><value>	This event is generated whenever the contents of the SYS.eJ1939.TIRE_CPC_TTM_DEFECT variable changes. <comp> Compares the contents of this variable with the used specified one and return a Boolean (True/False) representing the result of the comparison. It can be set to =, !=, <,>, <= or>=. <value> Specifies the value to be compared. *Example: SYS.eJ1939.TIRE_CPC_TTM_DEFECT>=0*
SYS.eJ1939.TIRE_CPC_TTM_LOSE<comp><value>	This event is generated whenever the contents of the SYS.eJ1939.TIRE_CPC_TTM_LOSE variable changes. <comp> Compares the contents of this variable with the used specified one and return a Boolean (True/False) representing the result of the comparison. It can be set to =, !=, <,>, <= or>=. <value> Specifies the value to be compared. *Example: SYS.eJ1939.TIRE_CPC_TTM_LOSE>=0*

TTM=Truck tire module "Sensor inside the tire"

Table 3: Supported events for alarm configuration for for ContiPressureCheck™ system

2.5 Supported parameters for ContiPressureCheck™

The following table shows the implemented dynamic entries/variables for reading real time data from ContiPressureCheck™ system.

Dynamic entries/Variables	Description of data received from the ContiPressureCheck™ system
Tire Condition	
J1939.TIRE_TEMP ≙ PNG 0xFEF4	It reports, into the comma separated value format, the temperature of each wheel as follows. e.g. **00**:21,**01**:22,**10**:20,**11**:20,**12**:20,**13**:21,**20**:21,**21**:22 00:21 – 00 = wheel position, 21 = temperature in °C.
J1939.TIRE2_NOMPRESS ≙ PNG 0xFEF4	It reports, into the comma separated value format, the nominative pressure in kPa of each tire as follows. e.g.: **00**:790,**01**:795,**10**:790,**11**:795,**12**:790,**13**:795,**13**:790,**13**:795 00:790 – 00 = wheel position, 790 = nominative tire pressure in kPa.
J1939.TIRE2_PRESSURE ≙ PNG 0xFEF4	It reports, into the comma separated value format, the pressure of each tire as follows. e.g.: **00**:4,**01**:4,**10**:249,**11**:4,**12**:4,**13**:4,**20**:249,**21**:4 10:790 – 00 = wheel position, 249 = tire pressure in kPa.
J1939.TIRE_PRESSURE ≙ PNG 0xFEF4	It reports, into the comma separated value format, the pressure of each tire as follows. e.g.: **00**:4,**01**:4,**10**:248,**11**:4,**12**:4,**13**:4,**20**:248,**21**:4 10:790 – 11 = wheel position, 248 = tire pressure in kPa.
J1939.TIRE_FAULT ≙ PNG 0xFEF4	It reports, into the comma separated value format, the fault state of each tire as follows. e.g.: **00**:0,**01**:0,**10**:0,**11**:1, **12**:0,**13**:0,**20**:0,**121**:0 11:1 – 11 = wheel position, 0 = no fault; 1 = fault.
J1939.TIRE_PTD ≙ PNG 0xFEF4	It reports, into the comma separated value format, the pressure threshold detection of each tire as follows. e.g.: **00**:4,**01**:2,**10**:2,**11**:2,**12**:4,**13**:2,**20**:2,**21**:2 00:1 – 00 = wheel position; 1 = Over pressure, 2 = No warning pressure, 3 = Under pressure, 4 = Extreme under pressure.
J1939.TIRE_PLOSS ≙ PNG 0xFEF4	It reports, into the comma separated value format, the pressure loss of each tire as follows. e.g.: **00**:0,**01**:0,**10**:0,**11**:0,**12**:0,**13**:0,**20**:0,**21**:0 00:0 – 00 = wheel position; 1 = tire pressure loss in Pa/s (0.1 Pa/s per bit).
J1939.TIRE_SEN ≙ PNG 0xFEF4	It reports, into the comma separated value format, if the sensors are enabled as follows. e.g.: **00**:1,**01**:1,**10**:1,**11**:1,**12**:1,**13**:1,**20**:1,**21**:1 00:1 – 00 = wheel position; 0 = Sensor disabled, 1 = Sensor enabled.
J1939.TIRE_STAT ≙ PNG 0xFEF4	It reports, into the comma separated value format, the status of each tire as follows. e.g.: **00**:0,**01**:0,**10**:0,**11**:0,**00**:0,**01**:0,**10**:0,**11**:0 00:0 – 00 = wheel position; 0 = tire OK, 1 = tire leak.
J1939.TIRE_ETP ≙ PNG 0xFEF4	It reports, into the comma separated value format, the while position and if the extended tire pressure is supported as follows. e.g.: **00**:1,**01**:1,**10**:1,**11**:1,**20**:1,**21**:1,**30**:1,**31**:1 00 = wheel position; 0 = Not using Extended Tire Pressure, 1 = Using Extended Tire Pressure, 10: Error, 11: Not available/Not supported
CPC System Configuration	
J1939.TIRE_CPC_CNF_NAXLE ≙ PNG 0xFF00	It reports the number of axles for the tractor or trailer as follows. e.g.: **00**:4 00 = tractor, 01 = trailer; 4 = Number of axles.
J1939.TIRE_CPC_CNF_NCTTM	It reports the number of TTMs for the tractor and trailer as follows. e.g.:

≙ PNG 0xFF00	**00:**8 00 = tractor, 01 = trailer; 08 = Number of TTMs.
CPC System Status	
J1939.TIRE_CPC_STAT_HEALTH ≙ PNG 0xFF01	It reports the status of CPC system for the tractor and trailer as follows. e.g.: **00:**1 00 = tractor, 01 = trailer; 0 = OK; “No TTM mounted” NOT detected, 0 = “No TTM mounted” detected.
J1939.TIRE_CPC_NOTTM ≙ PNG 0xFF01	It reports if no TTMs mounted for the tractor and trailer as follows. e.g.: **00:**0 00 = tractor, 01 = trailer; 0 = OK “No TTM mounted” NOT detected, 0 = “No TTM mounted” detected..
J1939.TIRE_CPC_STAT_WEX ≙ PNG 0xFF01	It reports, into the comma separated value format, the single wheel exchange as follows. e.g.: **00:**n/a 00 = tractor, 01 = trailer; 0 = “Single wheel exchanged” not detected, 0 = “Single wheel exchanged” detected.
J1939.TIRE_CPC_STAT_LEARN ≙ PNG 0xFF01	It reports, into the comma separated value format, the automatic trailer learning as follows. e.g.: **00:**0 00 = tractor; 0: Automatic trailer learning ongoing; 1: Automatic trailer learning finished, known trailer found, 2: Automatic trailer learning finished, new trailer found, 3: Automatic trailer learning finished, no trailer found, 4: Feature not active.
CPC TTM Data	
J1939.TIRE_CPC_TTM_PRESSURE ≙ PNG 0xFF02	It reports, into the comma separated value format, the TTM pressure 4.706 kPa/bit as follows. e.g.: **00:**aa,**01:**0,**02:**0,**03:**0,**04:0,05:0,06:**0,**07:0,08:**0,**09:**0 00…17 = index range in hex used to identify a sensor id; 0 = Sensor defective or data not available; 01…FF = (aa -1) * 4.706 kPa/bit = 800 kPa, FF=Overflow.
J1939.TIRE_CPC_TTM_TEMP ≙ PNG 0xFF02	It reports, into the comma separated value format, the TTM temperature 1 °C/bit -50 K offset as follows. e.g.: **00:**22,**01:**22,**02:**22,**03:**22,**04:**22,**05:**22,**06:**22,**07:**22,**08:**22,**09:**22 00…17 = index range in hex used to identify a sensor id; 0 = Sensor defective or data not available; 01…FF = (4F -50K) = 45 °C, FF=Overflow.
J1939.TIRE_CPC_TTM_STATE ≙ PNG 0xFF02	It reports, into the comma separated value format, the TTM state as follows. e.g.: **00:**8,**01:**8,**02:**8,**03:**8,**04:**8,**05:**8,**06:**8,**07:**8,**08:**8,**09:**8 00…17 = index range in hex used to identify a sensor id; 8 = TTM state.
J1939.TIRE_CPC_TTM_ALARM ≙ PNG 0xFF02	It reports, into the comma separated value format, the alarm and warning as follows. e.g.: **00:**2,**01:**2,**02:**2,**03:**2,**04:**2,**05:**2,**06:**2,**07:**2,**08:**2,**09:**2 00…17 = index range in hex used to identify a sensor id; 2 = Alarm and warning.
J1939.TIRE_CPC_TTM_BAT ≙ PNG 0xFF02	It reports the battery flag as follows. e.g.: **n/a** n/a = Battery flag.
J1939.TIRE_CPC_TTM_DEFECT ≙ PNG 0xFF02	It reports the TTM defective as follows. e.g.: **n/a** 00 = wheel position; n/a = TTM defective.
J1939.TIRE_CPC_TTM_LOSE ≙ PNG 0xFF02	It reports, into the comma separated value format, the loose TTM detection as follows. e.g.: **00:**0,**01:**0,**02:**0,**03:**0,**04:**0,**05:**0,**06:**0,**07:**0,**08:**0,**09:**0 00…17 = index range in hex used to identify a sensor id; 0 = Loose TTM detection
CPC Graphical Position Configuration	
J1939.TIRE_CPC_POS:H<n>	It reports, into the comma separated value format, the value according to the matrix

≙ PNG 0xFF04	(graphical position and tire location) and the index used as a reference/conjunction for matching the tire location, graphical position and ID of the sensors in hexadecimal as follow s. e.g.: **00:**03,**01:**0b,**02:**43,**03:**4b,**04:**53,**05:**5b 00…1F = index range in hex used to identify a sensor id; 0b = Hexadecimal value to identify the graphical position of the sensors. Refer to the pic. 3 and Table 5 below.
J1939.TIRE_CPC_POS ≙ PNG 0xFF04	It reports, into the comma separated value format, the value according to the matrix (graphical position and tire location) and the index used as a reference/conjunction for matching the tire location, graphical position and ID of the sensors in decimal as follows. e.g.: **00:**3,**01:**11,**02:**67,**03:**75,**04:**83,**05:**91 00…1F = index range in hex used to identify a sensor id; 0b = Decimal value to identify the graphical position of the sensors. Refer to the pic. 3 and Table 5 below.
J1939.TIRE_CPC_LOC:H<n> ≙ PNG 0xFF04	It reports, into the comma separated value format, the value according to the matrix (graphical position and tire location) and the index used as a reference/conjunction for matching the tire location, graphical position and ID of the sensors in hexadecimal as follows. e.g.: **00:**00,**01:**01,**02:**10,**03:**11,**04:**20,**05:**21 00…1F = index range in hex used to identify a sensor id; 11 = Hexadecimal value to identify the tire location. Refer to the pic. 3 and Table 5 below.
J1939.TIRE_CPC_LOC ≙ PNG 0xFF04	It reports, into the comma separated value format, the value according to the matrix (graphical position and tire location) and the index used as a reference/conjunction for matching the tire location, graphical position and ID of the sensors in decimal as follows. e.g.: **00:**0,**01:**1,**02:**16,**03:**17,**04:**32,**05:**33 00…1F = index range in hex used to identify a sensor id; 17 = decimal value to identify the graphical position of the sensors. Refer to the pic. 3 and Table 5 below.
J1939.TIRE_CPC_TTM_ID ≙ PNG 0xFF04	It reports, into the comma separated value format, the while position and the TTM ID of the sensor in decimal as follows. e.g.: **00:**1835297152,**01:**1835297152,**02:**1821695488,**03:**1818171136,**04:**1825507328,**05:**1825000448,**06:**1821695360,**07:**1821695232 00…1F = index range in hex used to identify a sensor id; 1835297152=TTM ID in decimal value. Refer to the pic. 3 and Table 5 below.
J1939.TIRE_CPC_TTM_ID:H<n> ≙ PNG 0xFF04	It reports, into the comma separated value format, the while position and the TTM ID of the sensor in <n> decimal digits as follows. e.g.: J1939.TIRE_CPC_TTM_ID:H8 **00:**6d646950,**01:**6d646948,**02:**6c94de3c,**03:**6c5f16e6,**04:**6ccf0811,**05:**6cc74bf2,**06:**6c94dd4e,**07:**6c94dd08 00…1F = index range in hex used to identify a sensor id; 6d646950=TTM ID in hexadecimal value. Refer to the pic. 3 and Table 5 below.
J1939.TIRE_CPC_X ≙ PNG 0xFF04	It reports, into the carriage-return and line-feed, the collection of all CPC messages as follows. e.g.: J1939.TIRE_CPC_CNF_NAXLE:00:4 J1939.TIRE_CPC_CNF_NCTTM:00:8 J1939.TIRE_CPC_STAT_HEALTH:00:1 J1939.TIRE_CPC_STAT_WEX:n/a J1939.TIRE_CPC_STAT_LEARN:00:0 J1939.TIRE_CPC_TTM_PRESSURE:00:aa,01:0,02:0,03:0,04:0,05:0,06:0,07:0,08:0,09:0 J1939.TIRE_CPC_TTM_TEMP:00:22,01:22,02:22,03:22,04:22,05:22,06:22,07:22,08:22,09:22 J1939.TIRE_CPC_TTM_STATE:00:8,01:8,02:8,03:8,04:8,05:8,06:8,07:8,08:8,09:8 J1939.TIRE_CPC_TTM_ALARM:00:2,01:2,02:2,03:2,04:2,05:2,06:2,07:2,08:2,09:2 J1939.TIRE_CPC_TTM_BAT:n/a J1939.TIRE_CPC_TTM_DEFECT:n/a J1939.TIRE_CPC_TTM_LOSE:00:0,01:0,02:0,03:0,04:0,05:0,06:0,07:0,08:0,09:0 J1939.TIRE_CPC_POS:00:3,04:11,08:19,0c:27,10:67,14:75,18:83,1c:91 J1939.TIRE_CPC_LOC:00:0,04:1,08:16,0c:17,10:32,14:33,18:48,1c:49

	J1939.TIRE_CPC_TTM_ID:00:1835297152,01:1835297152,02:1821695488,03:1818171136,04:1825507328,05:1825000448,06:1821695360,07:1821695232

TTM=Truck tire module "Sensor inside the tire"

Table 4: Supported dynamic entries/variables for sending out values received from ContiPressureCheck™ system

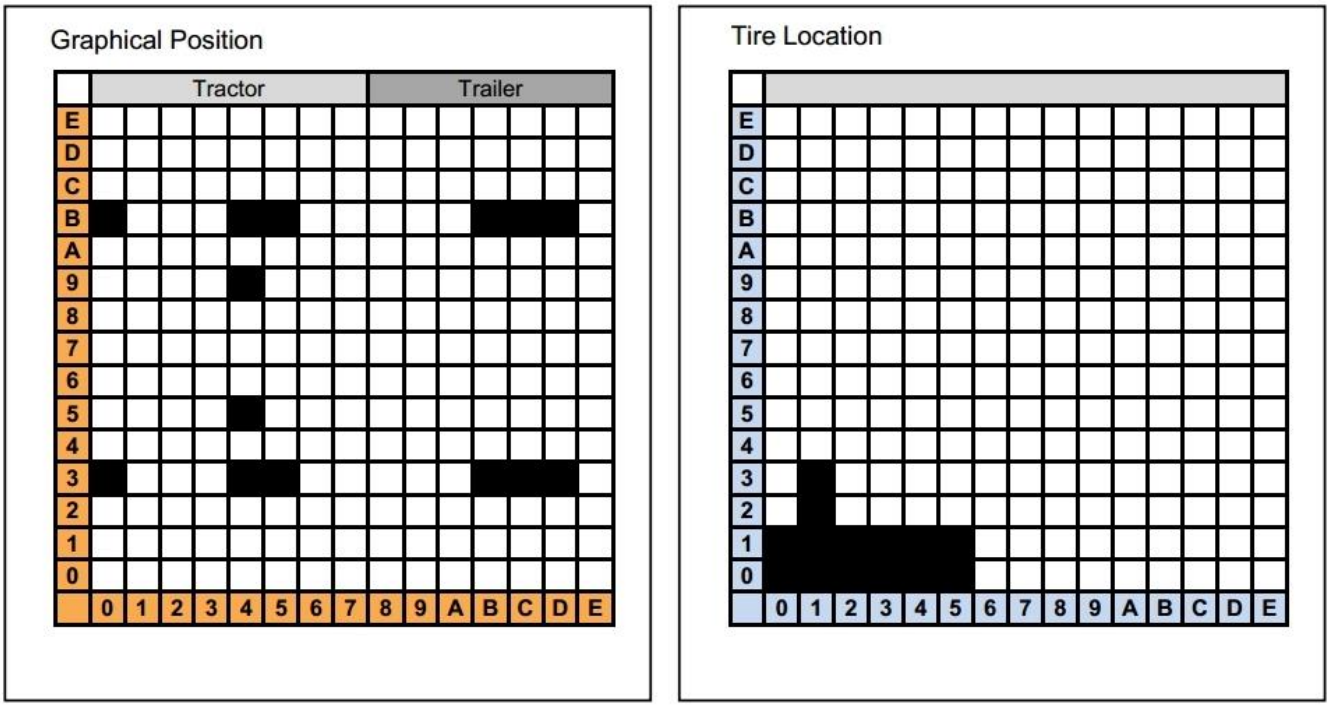


Fig 3: Example representing a 3-axle truck and a 3-axle trailer

Index (HEX)	Graphical Position (HEX)	Graphical Position (Dec)	Tire Location (HEX)	Tire Location (Dec)	Sensor ID
00	03	3	00	0	6d646948
01	0b	11	01	1	6c94de3c
02	43	67	10	16	1db1bd52
03	4b	75	11	17	6c94dd4e
04	53	83	20	32	1c649712
05	5b	91	21	33	6cc74bf2
…	…	…	…	…	…
1F	….	…	..	…	…

Table 5: Combination of index with graphical position and tire location and sensor ID received from the corresponding variables listed in table 3

The feature ATL (Automatic Trailer Learning) of the CPC system enables the CCU (Central Control Unit) of CPC, which is normally mounted on a truck, to detect a trailer which is hooked to this truck. That means, if the trailer tires are equipped with TTM, too, the pressure, temperature and warning information is also received by the CCU on the truck and put on the CAN bus. Data of all learnt tires are available, when ATL function has finished. Data of some trailer tires might be on the CAN even before the ATL algorithm terminates. The TTM data of the trailer is sent in the same way on the proprietary CAN (schedule list 2) like the manually configured TTM. The only difference is the signal "Graphical Position" (PGN 0xFF04). For all TTM learned by ATL the value is always 0xFF.

On schedule List 1 there is no information about trailer tires detected by ATL, because in SAE J1939 trailers are not considered.

2.6 Installation steps

- ✓ Insert your SIM card (supporting data) into the SIM slot of the FOX3/-3G/-4G device (refer to the hardware manual of the FOX3-2G/3G/4G devices).
- ✓ Connect the CA68 cable (installation and power cable) to the main port (8pin connector).
- ✓ Power off your vehicle. A standard CAN Bus will have a 120 ohm resistor termination at each end of the bus. Before connecting a FOX3/-3G/4G device to the CAN-H and CAN-L of the FMS gateway in your truck, check with an ohmmeter between the CAN-H and CAN-L if they are terminated at both ends by a 120-ohm resistor. If you read approximately 60 ohms, this means that two 120 ohm resistors are in parallel. If you read 120 ohms, one of the terminations is missing.
- ✓ Connect the **CAN_ Low** pin of this cable to the **CAN-L** of the FMS gateway on your truck.
- ✓ Connect the **CAN_High** pin of this cable to the **CAN-H** of the FMS gateway on your truck.
- ✓ Apply power to the device by connecting the BROWN wire of this cable to KL 31 (GND) and the RED wire of this cable to KL 30 (+12/24VDC source in the vehicle's fuse box).
- ✓ Connect the IGN pin (Blue wire) of this cable to the KL 15 (ignition of the vehicle).
- ✓ More information on how to install properly Lantronix devices in the vehicles is available in a separate document, see chapter **Error! Reference source not found.**, "*AppNotes_AVL_Installation_Guide.pdf*".
- ✓ The FOX3 device is now ready to receive the FMS data from the CPC system.

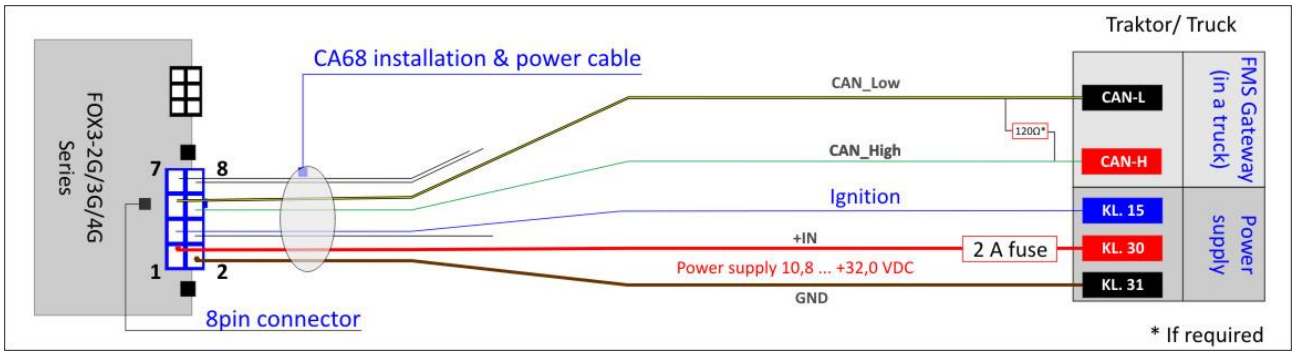


Fig. 4: Installation of FOX3-2G/3G/4G series to the FMS gateway.

3. DOCUMENTATION

3.1 Additional documentation & software tools

If this application note does not cover all the information you need to setup, refer to the additional documents listed below, which can be found on the *Product Index* or *Application Notes* pages.

Filename	Description
AVL_PFAL_Configuration_Command_Set.pdf	Lists and describes all PFAL commands supported by the AVL devices.
FOX3_3G_4G_HardwareManual.pdf	Contains instructions for safety and operation of the FOX3/-3G device.
AppNotes_AVL_Installation_Guide.pdf	Provides all the necessary information about installing the Lantronix products properly and safely.
AppNotes_HowToActivatePremiumFeatures.pdf	Provides required information on how to activate PREMIUM features in the ALV firmware 2.11.x and higher.
Workbench Software	**Description**
Lantronix Workbench	Lantronix Workbench configuration tool (Windows XP, Windows Vista, Windows 7)

Table 6: Additional documentation and software tools to this AppNotes