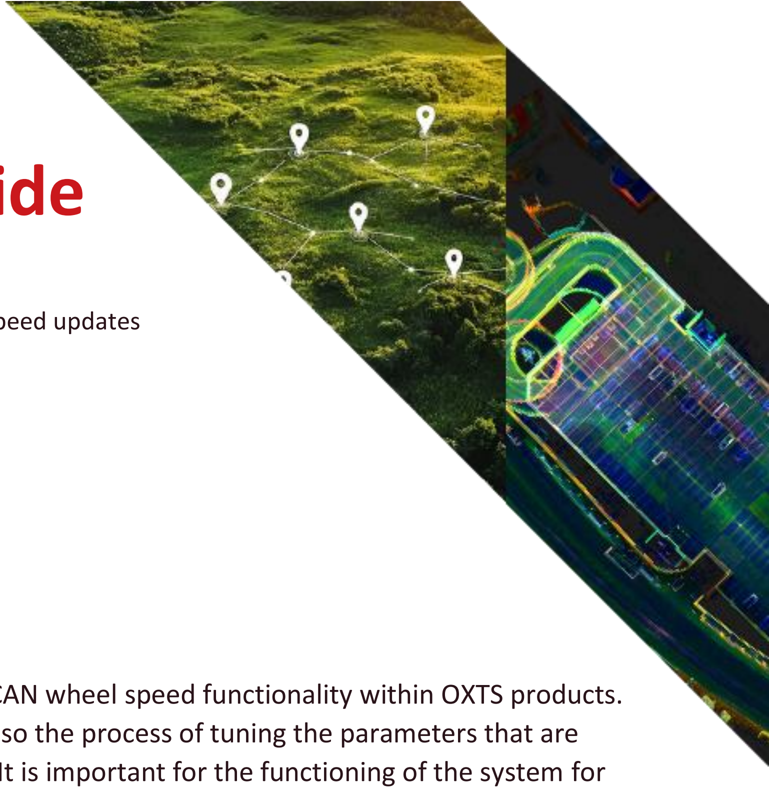

CAN Wheel Speed Guide

Description: Instruction on how to use and tune CAN wheel speed updates

Document owner: Navigation

Version: 250502

Keep informed:

1. Purpose

This document provides instructions on how to use the CAN wheel speed functionality within OXTS products. It covers how to configure the unit to allow its use and also the process of tuning the parameters that are required for the speed updates to function as expected. It is important for the functioning of the system for the CAN speeds to accurately represent the true speed, and this can be done with a comparison against speeds output by the OXTS unit in good GNSS conditions (NCOM format).

2. Prerequisites

The following are required in order to be able to configure and use the CAN wheel speed functionality with an OXTS unit:

- CAN acquisition and GAD feature codes enabled on unit

- NavSuite 3.13 (or higher) and associated firmware

- A device connected via CAN to record wheel speeds

3. User Guide

3.1 Configuring unit to use CAN wheel speed

Connect the unit to a computer with NavSuite via ethernet and open NavConfig and start a new configuration. There are two tabs to watch out for when configuring the unit to use CAN wheel speed updates.

The first is the CAN acquisition tab. Here the option to enable the interface should be selected, which will allow the option of specifying the CAN dbc file to be used. This is a file that defines the CAN messages for the connected device. For a general introduction to dbc files and how they are structured please see [1].

Once the file has been loaded, the message of interest (the one giving the desired speed measurements) should be selected. This should automatically read in the required information from the selected dbc file. A screen shot of how the CAN acquisition tab should look after the dbc file is imported is shown in Figure 1.

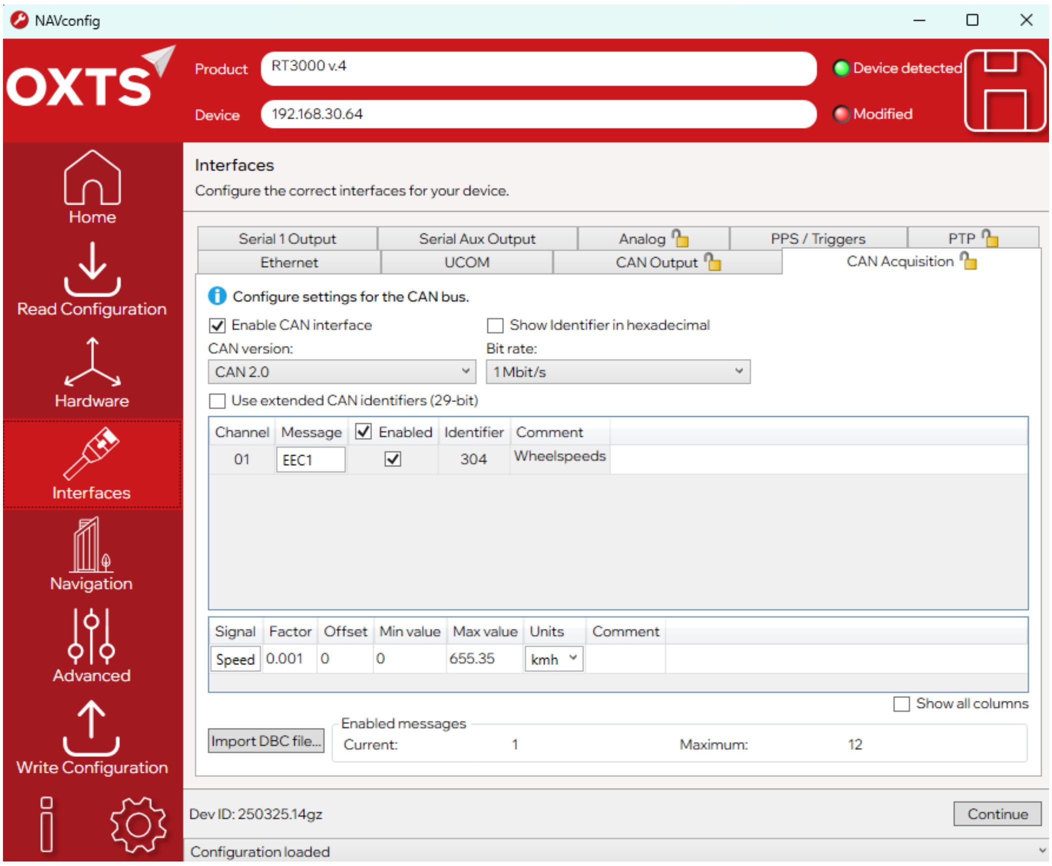

Figure 1: CAN Acquisition tab after dbc file has been imported

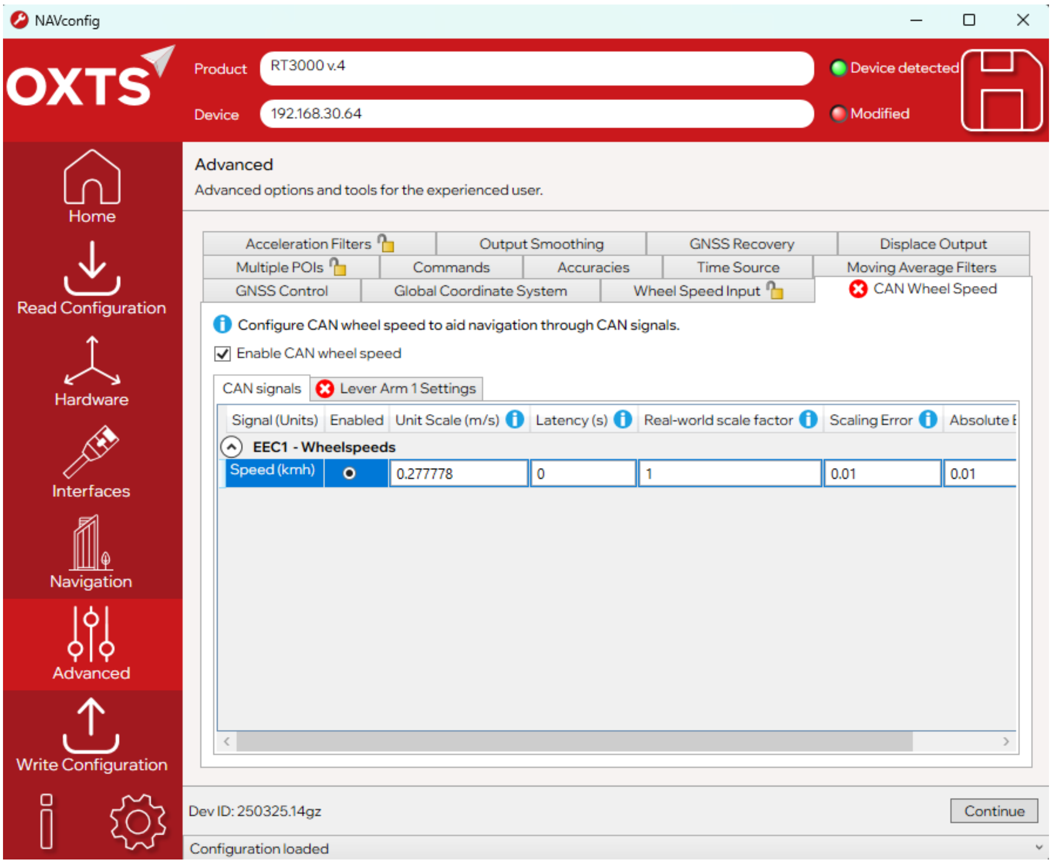

Figure 2: CAN Wheel Speed tab - CAN Signals

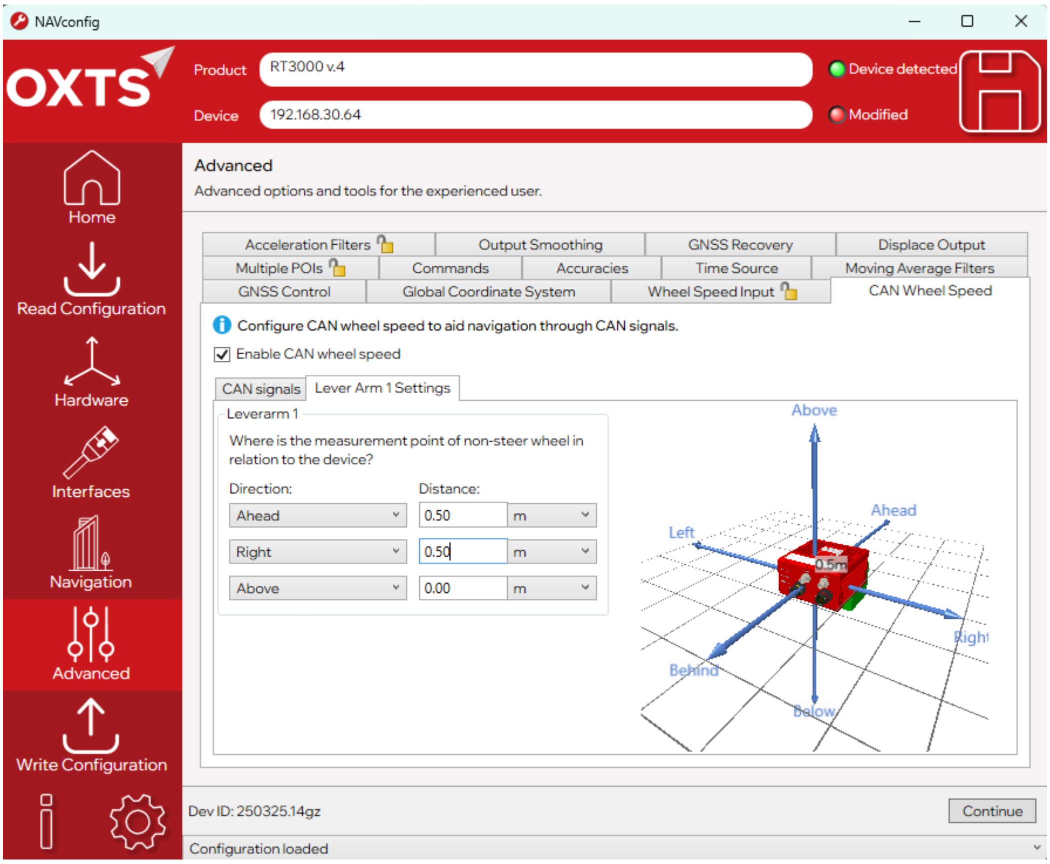

Figure 3: CAN Wheel Speed tab - Lever Arms Settings

The next tab of interest is the CAN wheel speed tab. Here we can specify/adjust some parameters related CAN wheel speed updates. There is an information tab for each in NavConfig, which describes what each parameter is. There is also an option to view lever arms for the device capturing the wheel speeds. The values that should be used here can be measured directly from each physical set up. These two options/tabs are shown in Figure 2 and Figure 3, respectively.

In what follows we will specify each of these parameters, give details on how to get the data required to tune them and finally provide hints on what tuning can be applied.

3.2 Parameters to be specified

Unit scale (m/s):

This converts the measurements in the selected speed signal to m/s.

Latency (s):

This is an estimate of the time taken between the speeds being measured and reaching the OXTS unit.

Real-world scale factor

This is a scaling factor between the measured CAN speeds and real-world speeds. As an example, if the CAN speeds are consistently reporting speeds 5% below the real-world speeds, then a value of 1.05 is selected for this scaling factor.

Scaling error

Measured estimated percentage error on the signal, which is specified as a decimal (e.g. 0.05 for 5% error).

Measured absolute error on the signal measured (m/s)

Would be 0.001 for 1mm/s error. This is separate to the scaling error above.

3.3 Obtaining CAN and NCOM speeds to use for scaling

Enable CAN acquisition in configuration.

Configure the unit as discussed above with the parameters discussed above selected to not impact on the speed values measured. In other words, the real-world scale factor set to 1 and latency 0.

Record dataset with corrections, or post process with a Rinex file. Unit should be configured to output csv file of GAD updates. This is done with the use of the following command:

-gad_csv_output_resolved

Open the resulting NCOM file in NavGraph and export the horizontal speeds to a csv file (right click on graph and export CSV format option).

NCOM speeds (with corrections) can be used as good indication of real-world speed, and the CAN speeds should now be available in the mobile.gad file, which can be saved a csv file for ease of use.

These speeds can then be compared, either visually or via calculations (ratio between speed values, error between the two etc). Note, if calculating errors or ratios it is important to make sure that the NCOM and CAN speeds are on a common time basis. This may involve some interpolation being required. In addition, only periods of non-zero valued speeds should be considered for finding the ratio. If visually comparing to get scaling factors etc, the CAN speeds can just be multiplied by the scaling factor and replotted meaning common timestamps are not necessarily required.

3.4 Parameter tuning

For each parameter below a guide on what (if any) tuning could be considered. Some further trial and error may be required. Note, care should also be taken to ensure only data where there is a reasonable starting match between NCOM and CAN speeds should be considered when tuning the parameters. In other words, we wouldn't consider trying to make zero valued can speed try to match non-zero valued NCOM speed by tuning the parameters as there has likely to have been an underlying issue we cannot solve by tuning.

Unit Scale (m/s)

Should be read in from the dbc file. Note, it is important to ensure that this value matches what is expected to change the units displayed in the CAN Acquisition tab to m/s.

Latency (s)

Visual inspection of a plot of the NCOM and CAN speeds can give a good starting indication of any issues. Consider if there is a common delay between peaks in speed or from when stationary periods end.

Real-world scale factor

A visual inspection of the speed plots can give an indication of if the CAN speeds is consistently above or below the NCOM/real world speed. This information can be used to iteratively adjust the scale factor and replot the speeds until they match. Alternative the ratio of CAN speed compared to NCOM speed can be found to provide the scale factor (we could start by considering the average over the dataset). An example of this being applied to a dataset is shown in Figure 4, here the scale factor applied is the mean of the ratio of the two speeds.

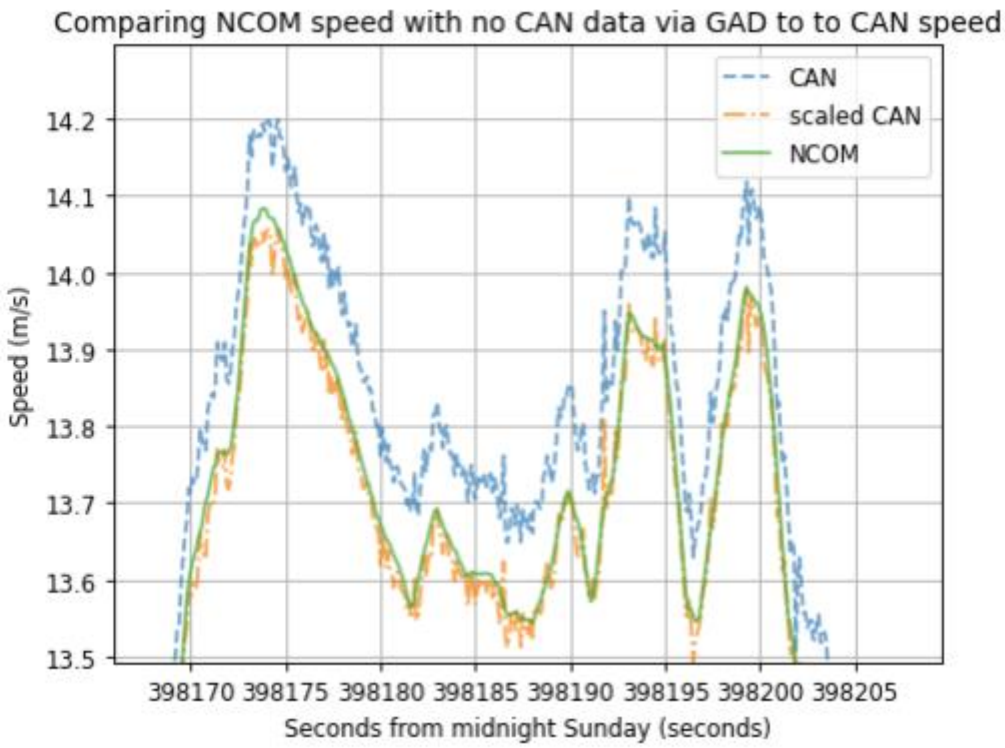

Figure 4: Effects of applying scale factor to CAN speeds

Scaling error

As a starting point for the scaling error, we can use the standard deviation of the values found when dividing NCOM speeds by CAN speeds. This ratio should be calculated after the latency and real-world scale factor values have been tuned.

However, this is assuming a linear speed sensor and if there are any deviations from this there is going to be a need for further tuning of this value to ensure that the desired performance is achieved in the final processed NCOM. By this we mean checking that the unit is actually stationary when it is known to be etc.

Measured absolute error on the signal measured (m/s)

In this case we would expect to get a reasonable starting point for the absolute error from NCOM speeds being compared with the CAN speed. In particular we will be looking at the regions when the NCOM speeds are zero. For these time points we simply can take the mean of the values from the CAN signal. This may well be very close to zero, in which case a particularly low value should be used as a starting point (e.g. 1mm/s or similar).

An alternative approach for a speed sensor giving quantised outputs is to find the smallest possible non-zero speed and divide this by 2 to get the measured absolute error. This is the largest amount of error possible when the vehicle is stationary before the sensor will report motion.

However, this is assuming a linear speed sensor and if there are any deviations from this there is going to be a need for further tuning of this value to ensure that the desired performance is achieved in the final processed NCOM. By this we mean checking that the unit is actually stationary when it is known to be etc. If the latency and real-world scale factor values are correctly tuned, then we would expect this value to be close to zero.

Examples of what is seen for poorly tuned parameters

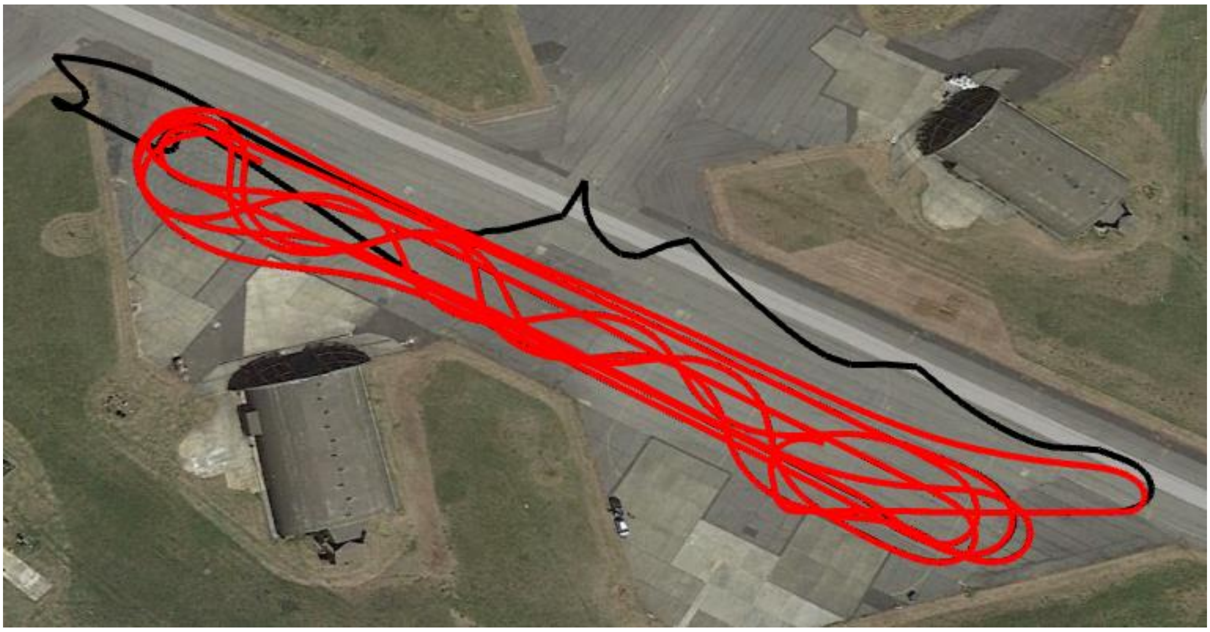

Figure 5: Example kml traces for appropriate error terms (red line) and inappropriate (black line) - parameter values too low in this case for the inappropriate values in this case

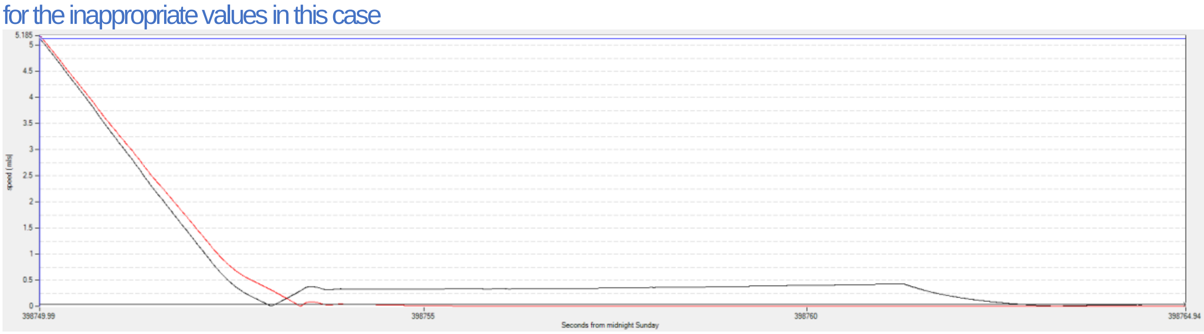

Figure 6: Example speed plots highlighting effects of inappropriate error terms (red line) as compared to appropriate values (black line) around a stationary period

As an example of the effects of poorly selected scale and absolute error values see the KML traces in Figure 5. Here inappropriately low scaling error and measured absolute error parameter values have severely degraded the overall performance. Figure 6 shows the undesirable changes in horizontal speed that these inappropriately low parameter values have had around a stationary period.

However, similar effects to what are seen in Figure 6 may also be noticed when the scaling and absolute errors are too large and there is no GNSS present. As a result it is important to have an accurate estimate of the errors in the system.

3.5 Reconfigure and double check

Once you have been through the tuning process discussed above, you can modify the configuration for your unit to include the parameter values you have obtained. Then subsequent runs can be performed with the system to check performance with the accurate CAN inputs.

4. Bibliography

[1] csselectronics, 31 10 2024. [Online]. Available: https://www.csselectronics.com/pages/can-dbc-file-database-intro.

5. Revision History

Revision	Changes
20/10/24	**Initial Draft**
13/11/24	**Figures Added**
02/05/25	**Revisions for external use**

Connect with us today

Global Headquarters

Park Farm Business Centre,
Middleton Stoney, OX25 4AL,
United Kingdom
+44 (0) 1869 814 253

US

Office 304, Regus Business Center
41000 Woodward Avenue
Suite 350 East, Bloomfield Hills
Michigan 48304
+1 248 260 1981

China

Room 901-902, 9th Floor, Building C,
Guanghualu SOHO II, Building No.9,
Guanghualu Chaoyang District,
Beijing, 100020,
China

---

linkedin.com/company/oxts/
x.com/oxts_official

---

info@oxts.com
oxts.com

OXTS