Datasheet xOEMcore

Subtitle:	xOEMcore datasheet
Version:	150120
Author:	BW
Document owner:	Marketing manager
Keep informed:	Changes to the datasheet will not be automatically communicated

List of contents

1. Features

The features of the xOEMcore are:

- 6-axis IMU suitable for navigation systems
- No export restrictions (Wassenaar, MTCR, ITAR free)
- Integrated navigation and Kalman filtering
- Exceptional accuracy MEMS inertial sensors
- Full calibration for linearity, axis alignment, bias, scale factor
- Wide temperature range
- Digital outputs

2. General description

The xOEMcore is an inertial measurement unit (IMU) consisting of 3 high-accuracy MEMS-based angular rate sensors (gyros), 3 high stability accelerometers, full inertial navigation algorithms and Kalman filter for navigation corrections. Add external aiding, such as GPS or Wifi positioning, to create a fully aided inertial navigation system (INS).

For the IMU all axes are factory calibrated for bias, scale factor, linearity and misalignment over the temperature range −10°C to 70°C. Sampling can be synchronised to an external 1PPS pulse to align the measurements to an external source such as GNSS. The unit runs from a single +5 V supply.

The measurements are output at 100 Hz and can be synchronised to an external timing source such as GNSS. The single output serial port can contain several data streams so that a simple microcontroller can change the electrical characteristics of the output without needing to interpret the data.

3. Ordering information

Table 1. Ordering information

Part	Description
xOEMcore	xOEMcore with IMU capability only
xOEMnav	Optional: navigation option added to xOEMcore
xOEMpp	Optional: logging and post-processing option added to xOEMcore, requires xOEMnav option
xOEMgxix	Optional: differential and gx/ix™ tight-coupling option added to xOEMcore, requires xOEMnav option
xOEMrtk	Optional: RTK gx/ix™ tight-coupling option added to xOEMcore, requires xOEMnav and xOEMgxix options

4. Pin description

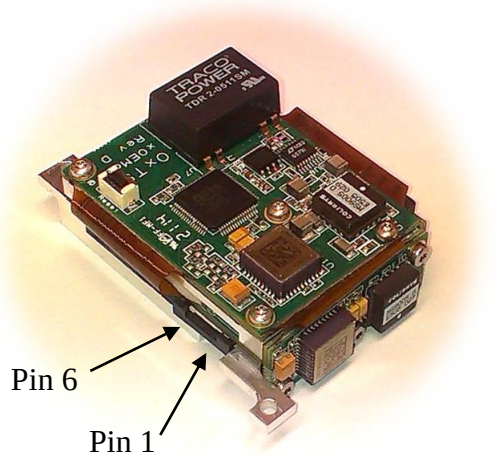
The xOEMcore uses a 6-way, 1 mm pitch FCC connector from Wurth Electronik, part number 68610614422. Example cables that can be used with this connector include part number 686606050001 (50 mm long) and 686606200001 (200 mm long).

Table 2. Pin connections

#	Name	Function	I/O level
1	VDD	Power supply input	
2	GND	Ground, 0 V supply	
3	SRX	Serial receive	3.3 V
4	STX	Serial transmit	3.3 V
5	TIN	Time input sync	3.3 V
6	TOUT	IMU sample output sync	3.3 V

Inputs are 5 V tolerant

Figure 1. Pin orientation



5. Absolute maximum ratings

Stresses above those listed as "absolute maximum ratings" may cause permanent damage to the xOEMcore. This is a stress rating only and functional operation of the device under these conditions is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

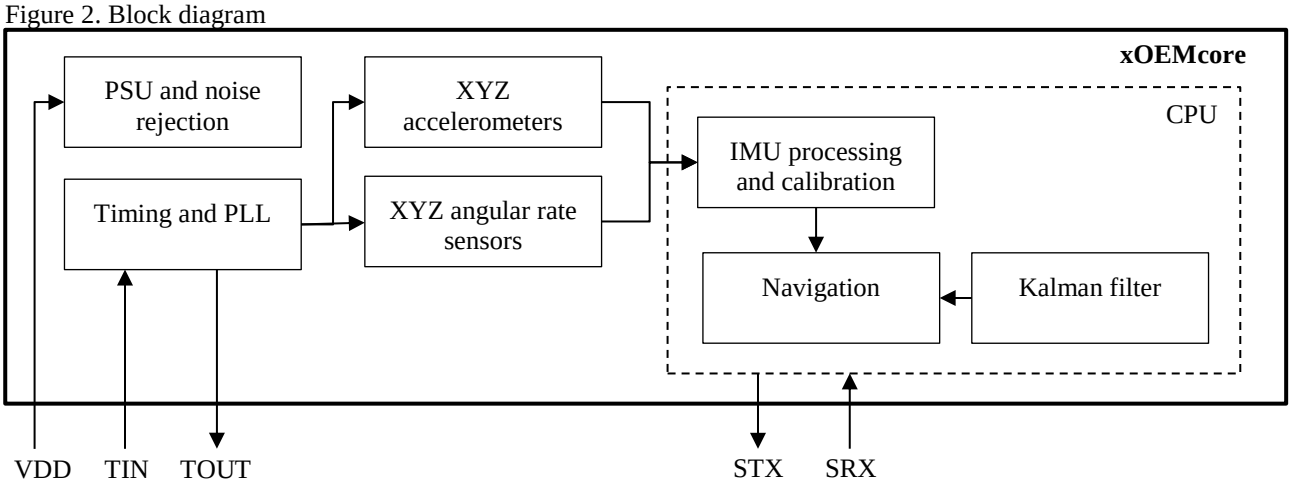
Table 3. Absolute maximum ratings

Symbol	Ratings	Maximum value	Unit
VDD	Supply voltage	−0.1 to 5.25	V
Tstg	Storage temperature	−40 to 85	°C
Top	Operating temperature	−40 to 85	°C
Sg	Acceleration shock (0.5 ms half-sine)	10000	m/s²
ESD	Electrostatic discharge protection	±2	kV
Vin	Input voltage on any control pin	−0.1 to 5.25	V

6. Block diagram

The xOEMcore includes 3 accelerometers and 3 angular rate sensors combined with an ARM-based processor and navigation algorithms. The block diagram is shown in Figure 2.

Figure 2. Block diagram



The xOEMcore includes a separate low noise power supply for the analogue sensors so it can be connected to standard digital power supplies without needing additional filtering. The TIN pin can be used to adjust the IMU synchronisation and the TOUT pin shows when the IMU samples the data.

The STX and SRX pins are standard logic level UART lines that transmit the navigation outputs and receive commands to control the function of the xOEMcore. In IMU-mode the STX transmits the IMU-mode output message and in navigation-mode STX transmits XCOM. SRX always receives commands in the CCOM format.

7. Specifications

7.1. Operating conditions

Table 4. Operating conditions

Symbol	Parameter	Min.	Typ.	Max.	Unit
Win	Input range, angular rate	−300		300	°/s
Ain	Input range, acceleration	−50		50	m/s²
Top	Operating temperature	−10		70	°C
VDD	Power supply	4.75		5.25	V

7.2. Functional specifications

Table 5. Functional specifications

Symbol	Parameter	Min.	Typ.	Max.	Unit
PDD	Power consumption (25°C, navigation running)		2.2		W
TB1	Power on to CPU bootloader output	860	870	880	ms
TB2	CPU boot to first firmware update character	1.25		4	s
TB3	Power on to firmware output		14		s
RTIN	TIN pull-up resistor to internal 3.3 V		10		kΩ

7.3. Angular rate sensor specifications

Table 6. Angular rate sensor specifications

Symbol	Parameter	Min.	Typ.	Max.	Unit
	Full scale	−300		300	°/s
	Calibration range	−200		200	°/s
	Resolution		18		bits
	Scale factor accuracy (−10°C to 70°C)		500		ppm
	Linearity (±200°/s, −10°C to 70°C)		500		ppm
	Misalignment, non-orthogonality (−10°C to 70°C)		0.02		°
	Bias error (−10°C to 70°C)		80		°/h
	In-run bias stability (Allan variance, ~25°C)		3		°/h
	Angular random walk (Allan variance, ~25°C)		0.5		°/√h
	Bandwidth (−3 dB)		≥40		Hz
	Internal sample rate		1000		Hz

7.4. Accelerometer specification

Table 7. Accelerometer specifications

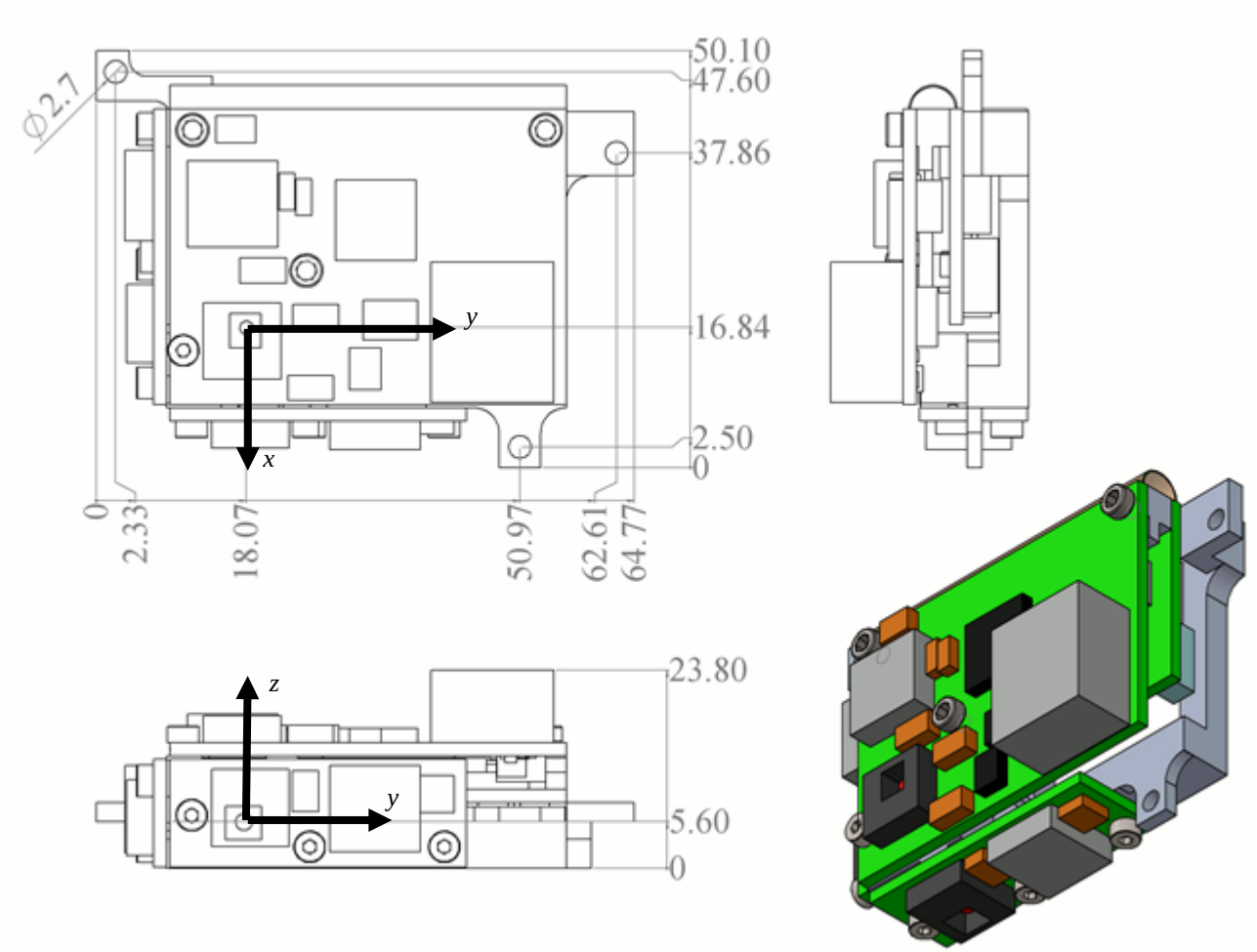
Symbol	Parameter	Min.	Typ.	Max.	Unit
	Full scale	−50		50	m/s²
	Calibration range	−9.8		9.8	m/s²
	Resolution		18		bits
	Scale factor accuracy (±1 *g*, −10°C to 70°C)		500		ppm
	Linearity (±1 *g* range, −10°C to 70°C)		500		ppm
	Misalignment, non-orthogonality (−10°C to 70°C)		0.02		°
	Bias error (−10°C to 70°C)		0.01		m/s²
	In-run bias stability (Allan variance, ~25°C)		0.0005		m/s²
	Velocity random walk (Allan variance, ~25°C)		0.05		m/s/√hr
	1 year bias repeatability			0.04	m/s²
	Bandwidth (−3 dB)			≥40	Hz
	Internal sample rate		1000		Hz

8. Mechanical

Table 8. Mechanical specifications

Symbol	Parameter	Min.	Typ.	Max.	Unit
	Mass		50		g
	Recommended torque			3	Nm
	Mounting surface flatness			0.1	mm
	Length (*x*-dimension)		50.1		mm
	Width (*y*-dimension)		64.8		mm
	Depth (*z*-dimension)		23.8		mm

Figure 3. Dimensions

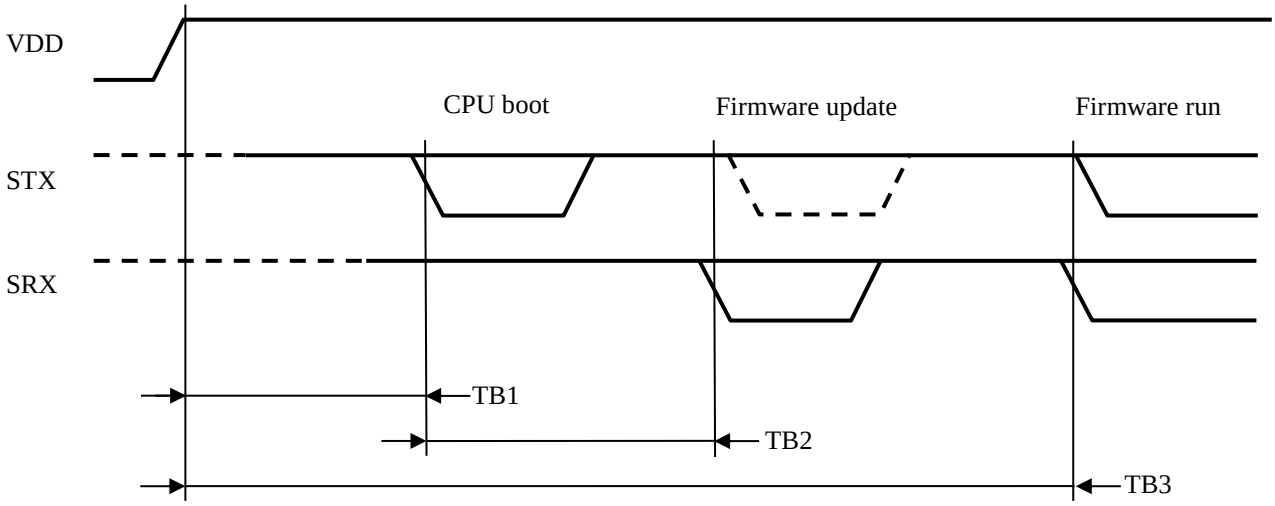


Acceleration measurement of each accelerometer has been moved to the measurement point before being output.

9. Power up sequence

After power is applied to VDD the xOEMcore boots as shown in Figure 4.

Figure 4. Boot sequence



The ARM processor in the xOEMcore runs its bootloader after TB1. The xOEMcore is ready to accept new firmware after TB2; if firmware is received then it is checked for errors, saved to FLASH memory and then run, which extends the time of TB3. The xOEMcore starts running its firmware after TB3.

When the xOEMcore firmware runs, it will start in one of two modes: IMU-mode or navigation-mode. The mode is determined by the features that were purchased with the xOEMcore. It is possible to upgrade from IMU-mode to navigation-mode using a code supplied by OxTS.

10. ARM bootloader

The ARM bootloader will output a sequence of approximately 58 bytes after TB1. These bytes are output at 115.2 kbaud, 8 data bits, even parity and 1 stop bit. Do not send any bytes to the xOEMcore at this time or the firmware in the xOEMcore may not run.

11. IMU-mode operation

The basic xOEMcore in IMU-mode operates as an inertial measurement unit (IMU) with outputs of 'change in velocity' and 'change in angle'. The xOEMcore will start in this mode unless the xOEMnav software option has been ordered and stored.

11.1. IMU-mode basic operation

The xOEMcore is very simple to use in IMU-mode. The simplest connection is shown in Figure 5.

Figure 5. IMU-mode free-running electrical connection diagram

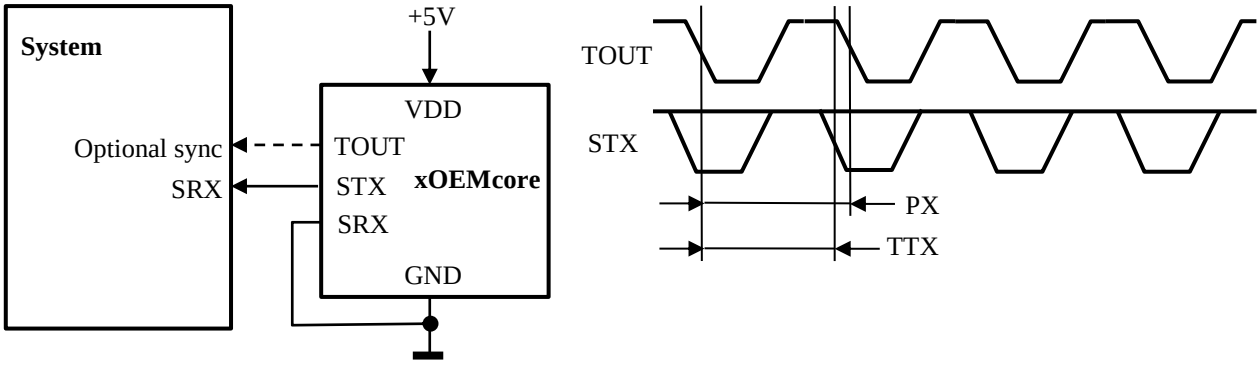


Table 9. IMU-mode free-running timing

Symbol	Parameter	Min.	Typ.	Max.	Unit
TTX	TOUT low to STX start	9.165	9.171	9.178	ms
PX	IMU output period		0.01		ms

The TIN pin can be left floating since it is pulled high, to the internal 3.3 V supply, with RTIN (Table 5). The value in the "time from 1PPS" field in the IMU-mode output message will not be valid when using the free-running mode. For the duty cycle of TOUT see Figure 6 and Table 10.

The TOUT pin indicates the sample time of the first sample in the period. The underlying sample rate of the xOEMcore is 1 kHz, so 9 additional epochs are sampled before the outputs can be computed and the serial output begins.

11.2. IMU-mode synchronised electrical connection diagram

The xOEMcore can be synchronised to the falling edge of a 1PPS pulse, normally from a GNSS receiver. In this mode a digital phase locked loop will adjust the sampling time of the IMU so it lines up with the 1PPS.

Figure 6. IMU-mode synchronised electrical connection diagram

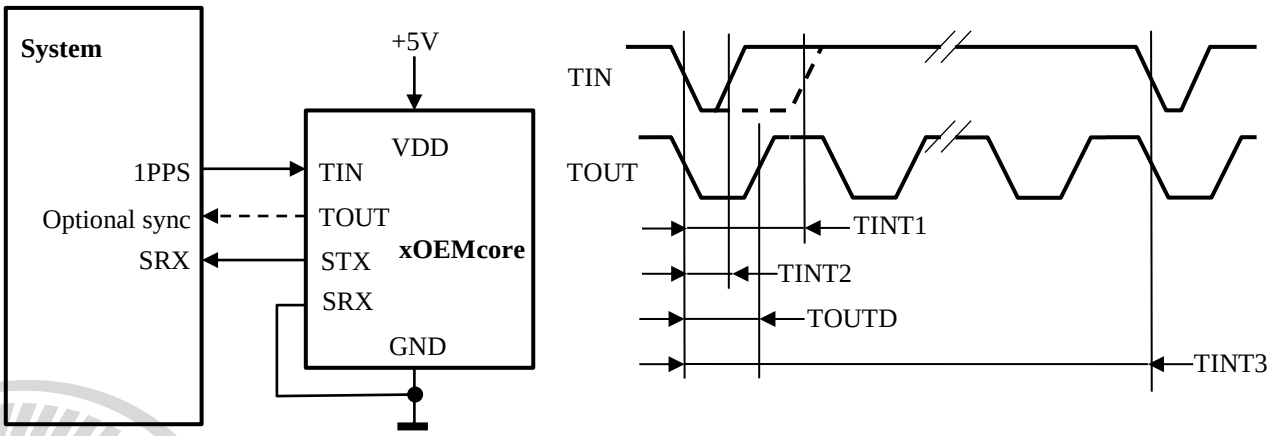


Table 10. IMU-mode synchronised timing

Symbol	Parameter	Min.	Typ.	Max.	Unit
TINT1	TIN minimum pulse width	0.5			ms
TINT2	TIN maximum pulse width			9	ms
TINT3	TIN period	995	1000	1005	ms
TOUTD	TOUT duty cycle	45	50	55	%

If the 1PPS pulse stops then the digital phase-locked loop will free-run. The value in the "time from 1PPS" field of the IMU-mode output message continues to increase until it saturates.

If the 1PPS pulse changes then the digital phase-locked loop will resynchronise on the new pulse and the timing parameters will be different during the transition period. The time from the IMU will count from the new pulse on the next output cycle.

11.3. IMU-mode output message

In IMU-mode the serial port will operate at 115.2 kbaud, with 8 data bits, no parity and 1 stop bit. The output message will be a sequence of binary bytes described in Table 11.

Table 11. IMU-mode output message description

Offset	Length	Format	Description
0	1		Sync character, always 45h
1	3	uint24	Time from 1PPS in µs. Time will saturate if no 1PPS pulse is received
4	3	int24	Change in velocity along the *x*-axis at the end of the interval in units of 10^{-6} m/s
7	3	int24	Change in velocity along the *y*-axis at the end of the interval in units of 10^{-6} m/s
10	3	int24	Change in velocity along the *z*-axis at the end of the interval in units of 10^{-6} m/s
13	3	int24	Change in angle around the *x*-axis at the end of the interval in units of 10^{-6} degrees
16	3	int24	Change in angle around the *y*-axis at the end of the interval in units of 10^{-6} degrees
19	3	int24	Change in angle around the *z*-axis at the end of the interval in units of 10^{-6} degrees
22	1	uint8	Saturation bits
23	2	int16	Temperature in units of 0.01°C
25	1	uint8	Checksum, sum of all the bytes (from 0 to 24) modulo 256

The saturation bits are described in Table 12.

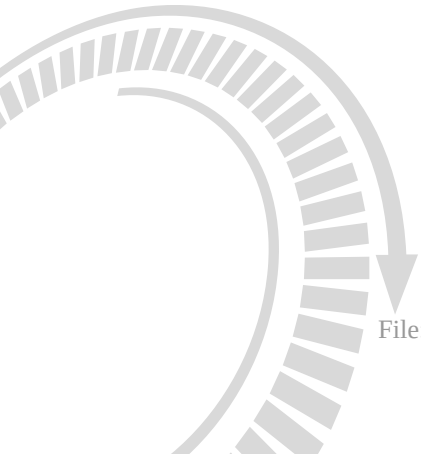


Table 12. IMU-mode output saturation bits description

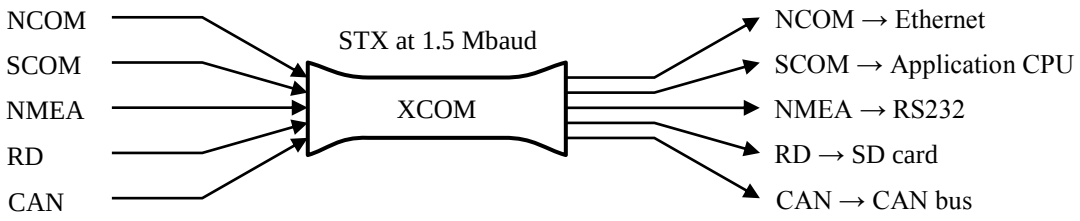
Bit	Description
0	Set to 1 if the *x*-axis accelerometer saturates on any sample during the interval
1	Set to 1 if the *y*-axis accelerometer saturates on any sample during the interval
2	Set to 1 if the *z*-axis accelerometer saturates on any sample during the interval
3	Set to 1 if the *x*-axis angular rate sensor saturates on any sample during the interval
4	Set to 1 if the *y*-axis angular rate sensor saturates on any sample during the interval
5	Set to 1 if the *z*-axis angular rate sensor saturates on any sample during the interval
6	Reserved
7	Reserved

All multi-byte values are in little-endian format (Intel) and signed values are encoded using 2's compliment arithmetic.

12. Navigation-mode operation

When running in navigation-mode the xOEMcore will change its serial port to 1.5 Mbaud, 8 data bits, no parity and 1 stop bit. It will output messages using the XCOM container format, which is very similar to the standard Ogg container format. The XCOM container format allows several data streams of different types to be sent on the same serial line, as shown in Figure 7.

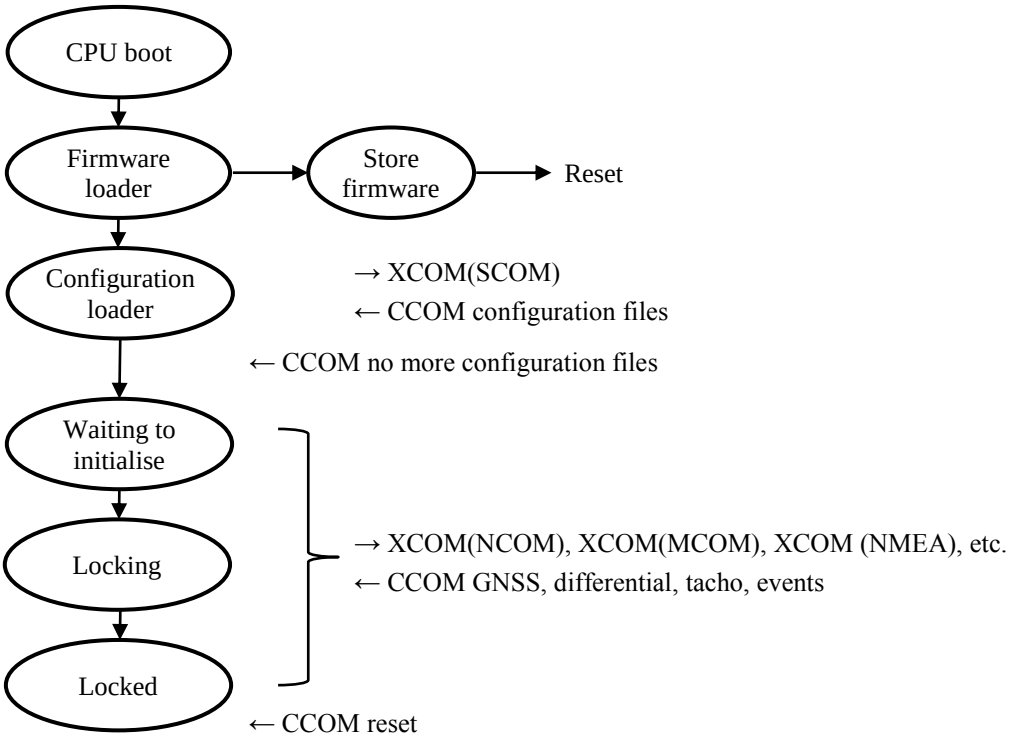
Figure 7. XCOM container



Using this technique the xOEMcore can generate several popular data streams and a simple PIC microcontroller can be used to extract the correct stream and send it on another electrical interface such as Ethernet or RS232.

The start-up sequence in navigation mode is shown in Figure 8.

Figure 8. Navigation mode start-up sequence



The xOEMcore will start by requesting the configuration files using an XCOM(SCOM) packet. This request can be used to identify that the xOEMcore firmware has booted and is ready. No commands should be sent to the xOEMcore before this packet is received.

The configuration is sent to the xOEMcore using CCOM format (not wrapped in XCOM). Both the hardware configuration files and the standard configuration files should be sent to the xOEMcore at this time. The standard configuration files can be created using NAVconfig, saved to a folder and then encoded in CCOM and sent. The hardware configuration file fields are described in the user manual.

Once all the configuration files have been sent, the CCOM "No more configuration file" message is sent. The xOEMcore will interpret the configuration files and start to output other messages as requested. CCOM messages for GNSS updates, wheel speed odometer (tacho), events and commands can now be sent to the xOEMcore.

The navigation algorithms can only start once the xOEMcore has estimates for time, position, velocity and orientation. Typically time, position and velocity are given by GNSS, roll and pitch are assumed to be close to zero at the start and heading is found by forward motion. Commands can be used to send position, velocity and orientation instead.

While waiting to initialise the xOEMcore runs and outputs with a 1 second delay. When all the conditions for initialisation are available, the xOEMcore starts 1 second behind and then catches up over a period of 10 seconds; this is known as the "locking" state. After 10 seconds the xOEMcore outputs its measurements with very little delay.

The data load on the STX serial line can be variable and xOEMcore prioritises low latency packets (NCOM, NMEA) over other packets (RD) so that the delay to the customer is kept as small as possible.

Detailed information on how to control the xOEMcore can be found in the user manual and the definition of the communication messages can be found in the communication manual.

13. Firmware update

The firmware for the xOEMcore can be updated by sending the firmware file from OxTS at TB2, see Table 5 and Figure 4. The serial port is set to 1.5 Mbaud, 8 data bits, 1 stop bit and no parity. The firmware file already contains synchronisation and integrity information so no additional encoding is required. A PC utility is available for updating the firmware. Once a valid firmware file has been received, the firmware is programmed into the FLASH memory and then executed.

14. Revision history

Table 13. Revision history

Revision	Changes
150112	First release.
150120	Correction to YZ axis directions.

15. Disclaimer