

Product name	Description	Version
ST-1612i-DBX	Datasheet of ST-1612i-DBX multiple GNSS module	1.8



## 1 Introduction

The LOCOSYS ST-1612i-DBX Dead Reckoning (DR) module is the perfect solution for automotive application. The ST-1612i-DBX is embedded 3D accelerometer, 3D gyroscope Micro-Electro-Mechanical Systems (MEMS) sensor, equipped with DR software, and powered by STMicroelectronics Teseo III. With adverse GNSS conditions in urban canyons, tunnels, or parking garages where DR boosts the accuracy, and the software fills the gaps. It supports three dimensional DR. This module can simultaneously acquire and track multiple satellite constellations that include GPS, BeiDou, GALILEO and QZSS. It features high sensitivity, low power and ultra small form factor, and provide user the superior performance.

ST-1612i-DBX modules use GNSS chips qualified according to AEC-Q100 and manufactured in IATF 16949 certified sites.

## 2 Features

- STMicroelectronics Teseo III high sensitivity solution
- AEC-Q100 qualified for automotive grade
- Support GPS, BeiDou, GALILEO and QZSS
- Capable of SBAS (WAAS, EGNOS, MSAS)
- Fast TTFF at low signal level
- Built-in Dead Reckoning (DR) software
- Built-in MEMS sensor (3-axis Gyroscope and 3-axis Accelerometer)
- Support Odometer (wheel-tick pulse) input
- Support Forward / Reverse Direction (FWD) input
- Support sensors data feed through the UART port
- Support SAE J2945/1 On-board system requirement for V2V safety communication
- IATF 16949 quality control
- Support CAN Bus ( ISO 15765-4 )
- Small form factor 16 x 12.2 x 2.3 mm
- SMD type, RoHS compliant

### 3 Application

- Automotive Navigation
- Marine Navigation
- M2M
- Track patrol and examine car

### 4 System Block Diagram

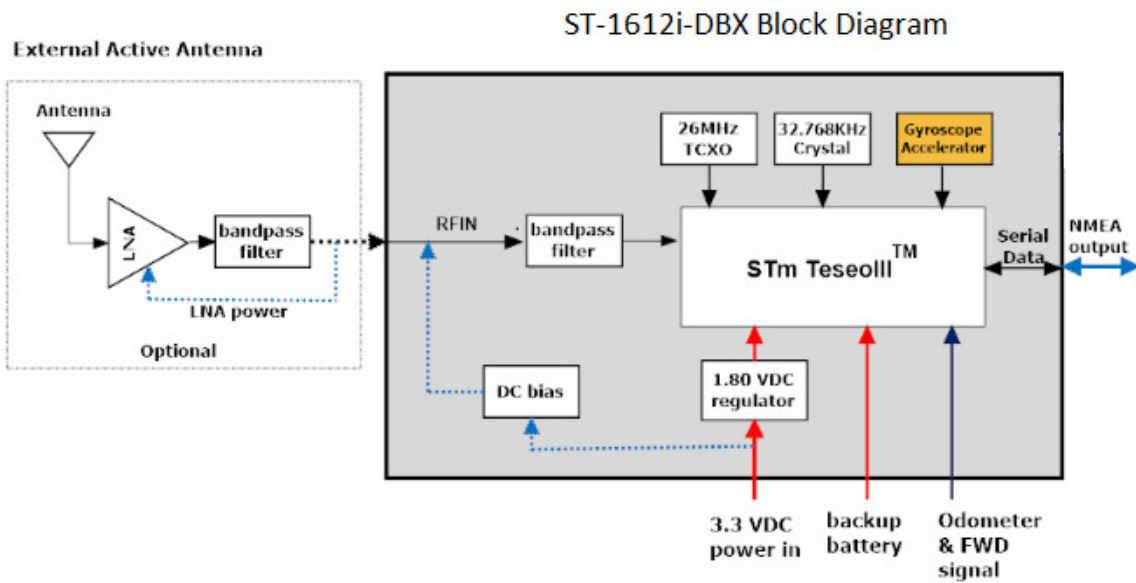


Fig 4-1 System block diagram.

## 5 GNSS receiver

### 5.1 GNSS receiver

Chip	Teseo III Series	
Frequency	GPS, GALILEO, QZSS: L1 1575.42MHz, C/A code BeiDou: B1 1561.098MHz, C code	
Channels	Support 48 channels	
Update rate	1Hz default	
Sensitivity	Tracking <sup>(1)</sup>	-163dBm
	Cold start <sup>(2)</sup>	-147dBm
Acquisition Time	Cold Start (Open Sky)	36s (typical)
	Hot start (Open Sky)	1s (typical)
Position Accuracy	Autonomous	1.8m CEP
	SBAS	1.5m (depends on accuracy of correction data)
Max. Altitude	< 18,000 m	
Max. Velocity	< 515 m/s	
Protocol Support	NMEA 0183 ver 3.01	115200 bps <sup>(3)</sup> , 8 data bits, no parity, 1 stop bits (default) 1Hz <sup>(4)</sup> : GGA, GLL, GSA, GSV, RMC, VTG, ZDA

Note1: For signal tracking sensitivity. With a good external LNA (Noise figure <=1.0dB, Gain>=13dB).

Note2: With a good external LNA (Noise figure <=1.0dB, Gain>=13dB).

Note3: Both baud rate and output message rate are configurable to be factory default.

Note4: If needs changes update rate and NMEA sentence, please consult us.

### 5.2 MEMS Sensor

The ST-1612i-DBX receiver support Dead Reckoning (DR) function, it composed by a 3-axis Gyroscope and 3-axis Accelerometer. To provide optimal solution it needs to calibrate sensor parameters. The sensor parameters are initialized to default values allowing system to start up when no other better information is available, if system is started for the first time in an unknown vehicle gyroscope dedicated maneuvers are required and algorithms able to estimate these parameters.

### 5.3 Sensor Data Input

The receiver needs two more sensor input, odometer and forward/reverse direction signal.

There are three methods to feed datum to receiver as below descriptions.

- (1) Feed through vehicle Odometer (wheel-tick pulse) input.
- (2) Feed through the UART port; please contact LOCOSYS for more details.
- (3) Feed through CAN Bus (ISO 15765-4).

## 6 Pin assignment and descriptions

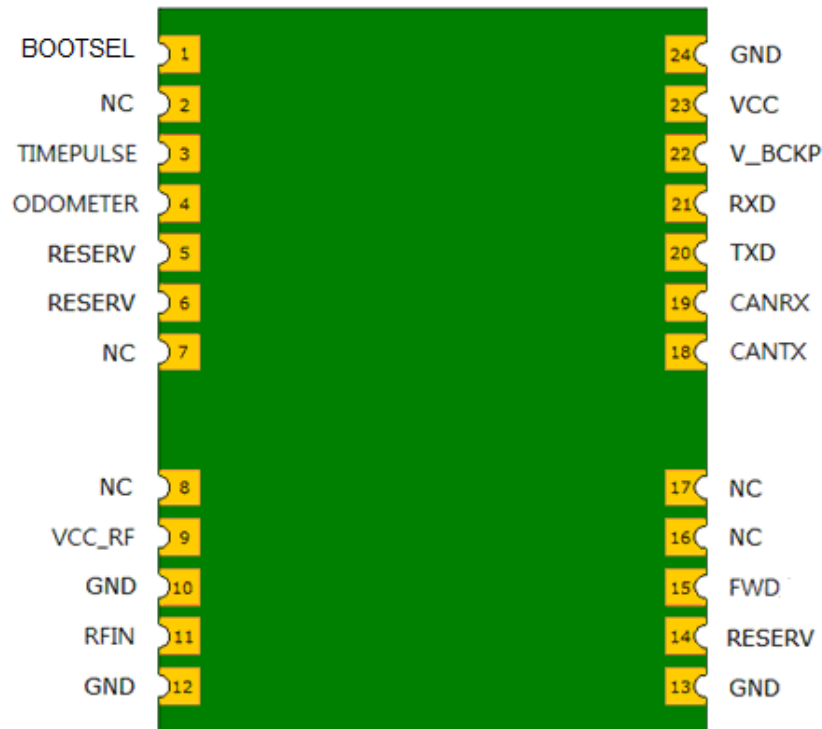


Table 6-1 Pin descriptions

Pin #	Name	Type	Description	Note
1	BOOTSEL		For firmware update use only, please keep floating	6
2	NC		Not connected	
3	TIMEPULSE	O	Time pulse (1PPS, default 500 ms pulse/sec, 3.3V Typical)	
4	ODOMETER	I	Odometer wheel-tick input	2
5	RESERV		Reserved, keep floating	
6	RESERV		Reserved, keep floating	
7	NC		Not connected	
8	NC		Not connected	
9	VCC_RF	O	Output voltage for active antenna	1
10	GND	P	Ground	
11	RFIN	I	GNSS RF signal input	
12	GND	P	Ground	
13	GND	P	Ground	
14	RESERV		Reserved, keep floating	
15	FWD	I	Forward/Reverse signal input	2
16	NC		Not connected	

17	NC		Not connected	
18	CANTX	O	CAN bus transmit data output	3
19	CANRX	I	CAN bus receive data input	3
20	TXD	O	UART, asynchronous output (Default NMEA)	
21	RXD	I	UART, asynchronous input	
22	V_BCKP	P	Backup battery supply voltage	4
23	VCC	P	DC supply voltage	5
24	GND	P	Ground	

Note1: The VCC\_RF pin short-circuit protection is built-in the module.

Note2: Pin4 and Pin15 need external level shift circuits to meet DC electrical characteristics. The Forward / Reverse input signal from the vehicle must be low for Forward and high for Reverse.

Note3: Need an external CAN Bus Transceiver (Maxim MAX3051, STM L9616D...etc) and vehicle CAN Bus protocol.

Note4: In order to stabilize the V\_BACKP power supply, placement of 10uF and 0.1uF capacitor is required to be close to the module.

Note5: The module is operated in fully functions, if customer want to turn module off for power saving reason, please add a PMOS or LDO (with enable pin) to control the module on or off.

Note6: If needs update firmware, please pull high into boot mode, else keep floating.

## 7 DC & Temperature characteristics

### 7.1 Absolute maximum ratings

Parameter	Symbol	Ratings	Units
Input Voltage	VCC	3.6	V
Input Backup Battery Voltage	V_BCKP	3.6	V
Operating Temperature Range	Topr	-40 ~ 85	°C
Storage Temperature Range	Tstg	-40 ~ 85	°C

### 7.2 DC Electrical characteristics

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Units
Input Voltage	VCC		3.0	3.3	3.6	V
Input Backup Battery Voltage	V_BCKP		1.8		3.6	V
VCC_RF Output Voltage	VCC_RF			VCC		V
Supply Current	I <sub>ss</sub>	VCC = 3.3V, w/o active antenna, Peak Acquisition Tracking			284	mA
				71		mA
				68 <sup>(1)</sup>		mA
Backup Battery Current	I <sub>bat</sub>	VCC = 0V		37		uA
VCC_RF Output Current	I <sub>out</sub>	V <sub>IN</sub> = 3.3V			25	mA
High Level Input Voltage	V <sub>IH</sub>		2.0		3.6	V
Low Level Input Voltage	V <sub>IL</sub>		-0.3		0.8	V
High Level Output Voltage	V <sub>OH</sub>		2.6			V
Low Level Output Voltage	V <sub>OL</sub>				0.4	V

Note1: Measured when position fix (1Hz) is available and input voltage is 3.3V with UART interface.

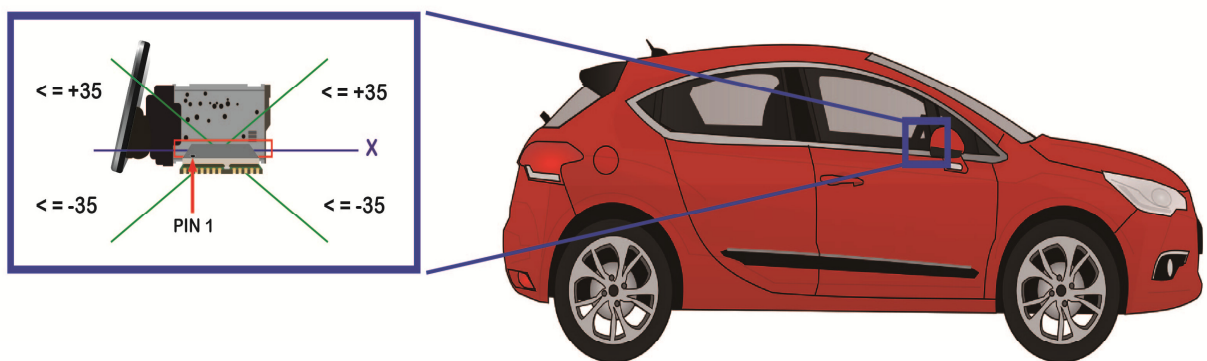
### 7.3 Temperature characteristics

Parameter	Symbol	Min.	Typ.	Max.	Units
Operating Temperature	Topr	-40	-	85	°C
Storage Temperature	Tstg	-40	25	85	°C

## 8 Mechanical specification

### 8.1 Recommended Mounting

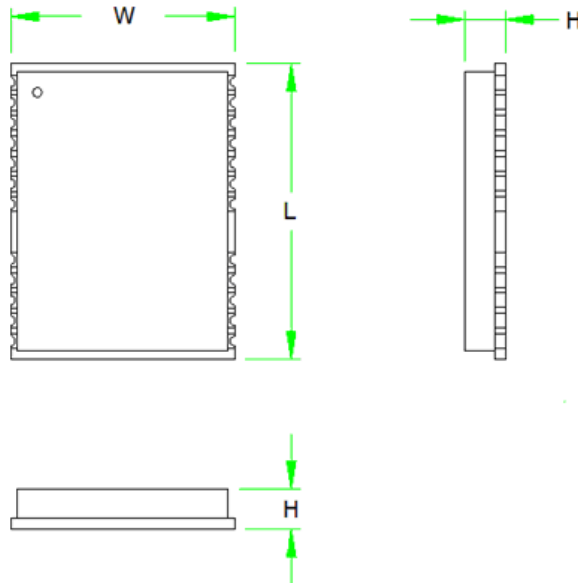
Please refer to the following figure to mount ST-1612i-DBX on vehicle. The module should be securely mounted to a stable part of the vehicle. The best position is over the center of the vehicle. For optimal performance, it should be mounted flat (level when the vehicle is on a level surface), but can deviate up to  $\pm 35^\circ$  in pitch (about the lateral axis).



**Note1: ST-1612i-DBX module MUST mount horizontally on vehicle (when the vehicle is on a level surface) and toward the front of vehicle. (Default is standard installation method)**

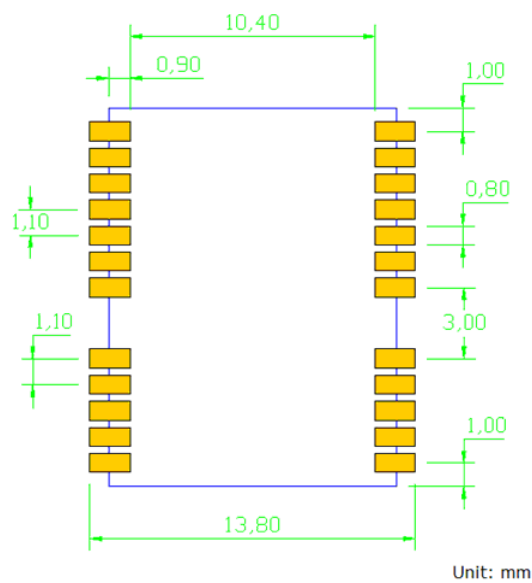
**Note2: If the module is not mounted as the above figure, please consult LOCOSYS in advance.**

### 8.2 Outline dimensions



Symbol	Min. (mm)	Typ. (mm)	Max. (mm)
W	12.1	12.2	12.3
L	15.7	16.0	16.4
H	2.1	2.3	2.5

### 8.3 Recommended land pattern dimensions



Note: The recommended land pattern dimensions are shown for reference only, as actual pad layouts may vary depending on application.



## 9 Software interface

### 9.1 NMEA output message

Table 9.1-1 NMEA output message

NMEA record	Description
GGA	Global positioning system fixed data
GLL	Geographic position - latitude/longitude
GSA	GNSS DOP and active satellites
GSV	GNSS satellites in view
RMC	Recommended minimum specific GNSS data
VTG	Course over ground and ground speed
ZDA	UTC, day, month and year.

- **GGA--- Global Positioning System Fixed Data**

Table 9.1-2 contains the values for the following example:

\$GNGGA,013654.000,2503.71447,N,12138.74593,E,1,16,0.7,130.00,M,15.3,M,,\*72

Table 9.1- 2 GGA Data Format

Name	Example	Units	Description
Message ID	\$GNGGA		GGA protocol header
UTC Time	013654.000		hhmmss.sss
Latitude	2503.71447		ddmm.mmmmm
N/S indicator	N		Latitude Direction: North or South
Longitude	12138.74593		dddmm.mmmmm
E/W Indicator	E		Longitude Direction: East or West
Position Fix Indicator	1		See Table 9.1-3
Satellites Used	16		Satellites in use
HDOP	0.7		Horizontal Dilution of Precision,max:99.0
MSL Altitude	130.00	meters	Height above mean sea level
Units	M	meters	Reference Unit for Altitude (“M” = meters)
Geoidal Separation	15.3	meters	Geoidal Separation measure in “M” = meters
Units	M	meters	Reference Unit for Geoidal Separation (“M” = meters)
DGPS Age			Not supported
DGPS Reference			Not supported
Checksum	*72		
<CR> <LF>			End of message termination

Table 9.1-3 Position Fix Indicators

Value	Description
0	Fix not available or invalid
1	GNSS fix valid
2	Differential GNSS fix valid
3-5	Not supported
6	Estimated (Dead Reckoning) Mode

- **GLL--- Geographic Position – Latitude/Longitude**

Table 9.1-4 contains the values for the following example:

\$GNGLL,2503.71447,N,12138.74593,E,013654.000,A,A\*42

Table 9.1-4 GLL Data Format

Name	Example	Units	Description
Message ID	\$GNGLL		GLL protocol header
Latitude	2503.71447		ddmm.mmmmm
N/S indicator	N		Latitude Direction: North or South
Longitude	12138.74593		dddmm.mmmmm
E/W indicator	E		Longitude Direction: East or West
UTC Time	013654.000		hhmmss.sss
Status	A		Validity of Data; A=data valid or V=data invalid
Mode	A		A = Autonomous mode, D = Differential mode, E = Estimated (Dead Reckoning) Mode, N=Data invalid,
Checksum	*42		
<CR> <LF>			End of message termination

- **GSA---GNSS DOP and Active Satellites**

Table 9.1-5 contains the values for the following example:

\$GNGSA,A,3,25,26,22,10,16,32,14,29,31,,1.1,0.6,0.9\*2B

\$GNGSA,A,3,193,,,,,,,,,1.1,0.6,0.9\*18

\$GNGSA,A,3,324,312,309,,,,,,,,,1.1,0.6,0.9\*1C

\$GNGSA,A,3,141,143,142,144,146,147,145,149,150,,1.1,0.6,0.9\*1E

Table 9.1-5 GSA Data Format

Name	Example	Units	Description
Message ID	\$GNGSA		GSA protocol header
Mode 1	A		See Table 9.1-6
Mode 2	3		See Table 9.1-7

ID of satellite used	25		Sv on Channel 1
ID of satellite used	26		Sv on Channel 2
....			....
ID of satellite used			Sv on Channel 12
PDOP	1.1		Position Dilution of Precision,max:99.0
HDOP	0.6		Horizontal Dilution of Precision, max:99.0
VDOP	0.9		Vertical Dilution of Precision, max:99.0
Checksum	*2B		
<CR> <LF>			End of message termination

Table 9.1-6 Mode 1

Value	Description
M	Manual: forced to operate in 2D or 3D mode
A	Automatic: allowed to automatically switch 2D/3D

Table 9.1-7 Mode 2

Value	Description
1	No Fix available
2	2D
3	3D

## ● GSV---GNSS Satellites in View

Table 9.1-8 contains the values for the following example:

```
$GNGSV,7,1,26,14,76,112,42,26,69,286,45,149,67,249,38,309,63,224,41*64
$GNGSV,7,2,26,147,60,338,42,143,58,205,39,141,55,140,38,146,54,191,39*61
$GNGSV,7,3,26,31,51,017,47,32,49,137,42,157,49,024,,150,48,296,36*63
$GNGSV,7,4,26,324,49,307,40,16,46,243,44,144,39,117,36,142,38,240,32*5A
$GNGSV,7,5,26,29,30,060,44,193,25,177,39,22,25,278,41,312,20,140,31*62
$GNGSV,7,6,26,03,18,297,,145,15,256,29,148,13,168,,27,11,192,*65
$GNGSV,7,7,26,25,09,042,37,50,51,134,35,,,,,,,,,*6E
```

Table 9.1-8 GSV Data Format

Name	Example	Units	Description
Message ID	\$GNGSV		GSV protocol header
Total number of messages <sup>1</sup>	7		Range 1 to 8
Message number <sup>1</sup>	1		Range 1 to 8
Satellites in view	26		Total Number of Satellites in view
Satellite ID	14		Channel 1 (Range 01 to 330)

Elevation	76	degrees	Channel 1 (Range 00 to 90)
Azimuth	112	degrees	Channel 1 (Range 000 to 359)
SNR (C/No)	42	dB-Hz	Channel 1 (Range 00 to 99, null when not tracking)
.....			.....
Satellite ID	309		Channel 4 (Range 01 to 330)
Elevation	63	degrees	Channel 4 (Range 00 to 90)
Azimuth	224	degrees	Channel 4 (Range 000 to 359)
SNR (C/No)	41	dB-Hz	Channel 4 (Range 00 to 99, null when not tracking)
Checksum	*64		
<CR> <LF>			End of message termination

Note1: Depending on the number of satellites tracked multiple messages of GSV data may be required.

Note2: GPS ID: 01~32, SBAS ID: 33~51, QZSS ID: 183~197, BeiDou ID: 141~172, GALILEO: 301~330

## ● RMC---Recommended Minimum Specific GNSS Data

Table 9.1-9 contains the values for the following example:

\$GNRMC,015924.000,A,2503.71417,N,12138.74623,E,0.1,0.0,230217,,D\*77

Table 9.1-9 RMC Data Format

Name	Example	Units	Description
Message ID	\$GNRMC		RMC protocol header
UTC Time	015924.000		hhmmss.sss
Status	A		A=data valid or V=data invalid
Latitude	2503.71417		ddmm.mmmmm
N/S Indicator	N		Latitude Direction: North or South
Longitude	12138.74623		dddmm.mmmmm
E/W Indicator	E		Longitude Direction: East or West
Speed over ground	0.1	knots	Speed over ground in knots
Course over ground	0.0	degrees	
Date	230217		ddmmyy
Magnetic variation		degrees	
Variation sense			Magnetic Variation Direction: East or West
Mode	D		A = Autonomous mode, D = Differential mode, E = Estimated (Dead Reckoning) mode, N=Data invalid
Checksum	*77		
<CR> <LF>			End of message termination

- **VTG---Course Over Ground and Ground Speed**

Table 9.1-10 contains the values for the following example:

\$GNVTG,0.0,T,,M,0.1,N,0.1,K,D\*16

Table 9.1-10 VTG Data Format

Name	Example	Units	Description
Message ID	\$GNVTG		VTG protocol header
Course over ground	0.0	degrees	Reference to “true” earth poles
Reference	T		Indicates “terrestrial”
Course over ground		degrees	Reference to “magnetic” earth poles
Reference	M		Indicates “Magnetic”
Speed over ground	0.1	knots	Speed over ground in knots
Units	N		Indicates “Knots”
Speed over ground	0.1	km/h	Speed over ground in kilometers per hour
Units	K		Indicates “Kilometers per hour”
Mode	D		A = Autonomous mode, D = Differential mode, E = Estimated (Dead Reckoning)mode
Checksum	*16		
<CR> <LF>			End of message termination

- **ZDA---UTC, day, month and year.**

Table 9.1-11 contains the values for the following example:

\$GNZDA,020849.00,23,02,2017,00,00\*78

Table 9.1-11 ZDA Data Format

Name	Example	Units	Description
Message ID	\$GNZDA		ZDA protocol header
Timestamp	020849.00		hhmmss.ss
Day	23		Decimal, 2 digits Day of month (01 to 31)
Month	02		Decimal, 2 digits Month (01 to 12)
Year	2017		Decimal, 4 digits Year (1994 - ...)
Local zone hour	00	hour	Local time zone offset from UTC (set to 00)
Local zone minutes	00	minute	Local time zone offset from UTC (set to 00)
Checksum	*78		
<CR> <LF>			End of message termination

## 9.2 Proprietary NMEA and DR input/output message

Table 9.2-1 DR output message

Software command	Command descriptions
\$PSTMCOLD<CR><LF>	Perform a Cold start
\$PSTMWARM<CR><LF>	Perform a Warm start
\$PSTMHOT<CR><LF>	Perform a Hot start
\$PSTMSRR <CR><LF>	Executes a system reset. The GNSS firmware is rebooted.
\$PSTMRESTOREPAR<CR><LF>	Restore System Parameters (Factory Settings).
\$PSTMSETPAR <CR><LF>	Set System Parameter in the configuration data block.
\$PSTMSAVEPAR <CR><LF>	Save System Parameters in the GNSS backup memory
\$PSTMNVMITEMINV<CR><LF>	Clear calibration status <sup>(1)</sup>

Note1: Optional Function

- **\$PSTMCOLD**

Perform a COLD start.

**Synopsis:**

\$PSTMCOLD,<Mask><CR><LF>

Table 9.2-2 \$PSTMCOLD Data format

Parameter	Format	Description
Mask	Integer	Optional parameter to invalidate time, position, ephemeris and almanac: 0x1-clear almanac 0x2-clear ephemeris 0x4-clear position 0x8-clear time

Note: If Mask parameter is not used, default is 0xE (clear ephemeris, time and position).

**Example:**

\$PSTMCOLD,6

- **\$PSTMWARM**

Perform a WARM start.

**Synopsis:**

\$PSTMWARM<CR><LF>

**Example:**

\$PSTMWARM

- **\$PSTMHOT**

Perform a HOT start.

**Synopsis:**

```
$PSTMHOT<CR><LF>
```

**Example:**

```
$PSTMHOT
```

- **\$PSTMSRR**

Executes a system reset. The GNSS firmware is rebooted.

**Synopsis:**

```
$PSTMSRR<CR><LF>
```

**Arguments:**

None.

**Results:**

The GNSS firmware reboots. No message will be sent as reply.

**Example:**

```
$PSTMSRR
```

- **\$PSTMRESTOREPAR**

Restore the factory setting parameters. The configuration data block stored in NVM, if present, will be invalidated. Any changed parameter will be lost.

**Synopsis:**

```
$PSTMRESTOREPAR<CR><LF>
```

**Results:**

None.

**Results:**

The current configuration data block, including changed parameters, will be stored into the backup memory (NVM).

If there are no error the following message is returned.

```
$PSTMSAVEPAROK
```

In case of errors, the error message is returned

```
$PSTMSAVEPARERROR
```

Note the factory setting parameters can be restored using the \$PSTMRESTOREPAR command.

**Example:**

```
$PSTMRESTOREPAR
```

● **\$PSTMSETPAR**

This command sets the defined parameter (indicated by “ID”) to the value provided as “param value” in the commands parameter.

**Synopsis:**

\$PSTMSETPAR,<ConfigBlock><ID>,<param\_value>[,<mode>]<CR><LF>

**Arguments:**

Table 9.2-3 \$PSTMSETPAR Data format

Parameter	Format	Description
ConfigBlock	Decima1,1 digit	Indicates one of configuration blocks 1 = Current Configuration, 2 = Default Configuration, 3 = NVM Stored configuration.
ID	Decimal, 3 digits	ID - Identifier
param_value	1 up to 80 bytes	Parameter to be set, see “Allowed values” as described in FW Configuration document.
mode	Decimal, 1 digit	This parameter is optional. It allows to perform bit-to-bit “OR” or “AND” operations between the selected parameter in the configuration block and the param_value in input. It has the following meaning: 0: the parameter in the configuration block is overwritten by the param_value. This is the default action as in the case mode is omitted. 1: the parameter in the configuration block is the result of bit-to-bit “OR” between old value and the param_value. This is useful for bit mask setting. 2: the parameter in the configuration block is the result of bit-to-bit “AND” between old value and NOT (param_value). This is useful for bit mask resetting.
<CR> <LF>		End of message termination

**Results:**

The parameter indicated by the ID value is set according to the parameters included in param\_value.

In case of no errors, the following message is returned

\$PSTMSETPAROK ,<ConfigBlock><ID>\*<Checksum><CR><LF>

In case of errors, the error message is returned

\$PSTMSETPARERROR\*<Checksum><CR><LF>

Table 9.2-4 \$PSTMSETPAROK Data format

Parameter	Format	Description
ConfigBlock	Decima1,1 digit	Indicates one of configuration blocks 1 = Current Configuration, 2 = Default Configuration, 3 = NVM Stored configuration.
ID	Decimal, 3 digits	ID - Identifier
Checksum	Hexadecimal, 2 digits	
<CR> <LF>		End of message termination

Note 1: The configuration block parameter is ignored by the “SET” command because only the current configuration, stored in the RAM memory, can be written. It is used only to keep same syntax as for the “GET” command. The configuration block stored in NVM will be overwritten by current configuration after the \$PSTMSAVEPAR command.



Note 2: There is no comma and no space between ConfigBlock and ID parameters.

Note 3: The input param\_value must be expressed in hexadecimal format without “0x” prefix for any integer value except DOP configuration. It must be decimal for any not integer value and DOP setting.

**Example:**

Issuing the command:

```
$PSTMSETPAR,1121,10
```

You could have this answer:

```
$PSTMSETPAROK,1121*30
```

**● \$PSTMSAVEPAR**

Save current configuration data block into the backup memory.

**Synopsis:**

```
$PSTMSAVEPAR<CR><LF>
```

**Arguments:**

None.

**Results:**

The current configuration data block, including changed parameters, will be stored into the backup memory (NVM).

If there are no error the following message is returned.

If there are no error the following message is returned

```
$PSTMSAVEPAROK
```

In case of errors, the error message is returned

```
$PSTMSAVEPARERROR
```

Note: the factory setting parameters can be restored using the \$PSTMRESTOREPAR command.

**Example:**

```
$PSTMSAVEPAR
```

**● \$PSTMNVMITEMINV (Optional Function)**

Clear calibration status

**Synopsis:**

```
$PSTMNVMITEMINV<ConfigBlock>,<ID>< CR><LF>
```

**Arguments:**

None.

**Example:**

```
$PSTMNVMITEMINV 80,1
```

### 9.3 Dead Reckoning (DR) output message

Table 9.3-1 DR output message

DR record	Description
\$PSTMDRSENMSG <CR><LF>	Format: for Msg ID = 1、 3、 14、 24、 30、 31
\$PSTMDRCAL<CR><LF>	Reports the calibration status of the DR calibration parameters.

- **\$PSTMDRSENMSG,1**

**Example:**

\$PSTMDRSENMSG,1,478872124,23556\*16

Table 9.3-2 \$PSTMDRSENMSG,1 Data Format

Name	Example	Description
Message	\$PSTMDRSENMSG	Reports DR sensor message data, which is specific to the message id for each specific DR sensor configuration.
Message ID	1	Odometer count
cpu timestamp	478872124	microseconds
odometer	23556	unsigned odometer count
Checksum	*16	
<CR> <LF>		End of message termination

- **\$PSTMDRSENMSG,3**

**Example:**

\$PSTMDRSENMSG,3,3377701539,557802,0\*0A

Table 9.3-3 \$PSTMDRSENMSG,3 Data Format

Name	Example	Description
Message	\$PSTMDRSENMSG	Reports DR sensor message data, which is specific to the message id for each specific DR sensor configuration.
Message ID	3	Odometer count and reverse signal input
cpu timestamp	3377701539	microseconds
odometer	557802	unsigned odometer count
Reverse	0	0 = forward, 1 = reverse
Checksum	*0A	
<CR> <LF>		End of message termination

- **\$PSTMDRSENMSG,14**

**Example:**

\$PSTMDRSENMSG,14,1173575113,28.0\*30

Table 9.3-4 \$PSTMDRSENMSG,14 Data Format

Name	Example	Description
Message	\$PSTMDRSENMSG	Reports DR sensor message data, which is specific to the message id for each specific DR sensor configuration.

Message ID	14	Vehicle speed
cpu timestamp	1173575113	microseconds
Vehicle speed	28.0	1 Kph resolution
Checksum	*30	
<CR> <LF>		End of message termination

- **\$PSTMDRSENMSG,24**

**Example:**

\$PSTMDRSENMSG,24,1174901488,5184,1\*3D

Table 9.3-5 \$PSTMDRSENMSG,24 Data Format

Name	Example	Description
Message	\$PSTMDRSENMSG	Reports DR sensor message data, which is specific to the message id for each specific DR sensor configuration.
Message ID	24	Gyroscope sensor temperature
cpu timestamp	1174901488	microseconds
temperature	5184	Gyro sensor temperature
validity	1	
Checksum	*3D	
<CR> <LF>		End of message termination

- **\$PSTMDRSENMSG,30**

**Example:**

\$PSTMDRSENMSG,30,2827546184,407,-125,16522\*3D

Table 9.3-6 \$PSTMDRSENMSG,30 Data Format

Name	Example	Description
Message	\$PSTMDRSENMSG	Reports DR sensor message data, which is specific to the message id for each specific DR sensor configuration.
Message ID	30	3D accelerometer
cpu timestamp	2827546184	microseconds
raw_x	407	$407\text{mg} / \text{LSB} * 0.061 = 24.827\text{mg} * 0.0098 \text{ m/s}^2 = 0.2433046 \text{ m/s}^2$
raw_y	-125	$-125\text{mg} / \text{LSB} * 0.061 = -7.625\text{mg} * 0.0098 \text{ m/s}^2 = -0.074725 \text{ m/s}^2$
raw_z	16522	$16522\text{mg} / \text{LSB} * 0.061 = 1007.842\text{mg} * 0.0098 \text{ m/s}^2 = 9.8768516 \text{ m/s}^2$
Checksum	*3D	
<CR> <LF>		End of message termination

Note: For unit  $\text{mg}/\text{LSB} * 0.061$ ,  $1\text{g} = 1000\text{mg}$ ;  $1\text{g} = 9.8 \text{ m/s}^2$ ;  $1\text{mg} = 0.0098 \text{ m/s}^2$

- **\$PSTMDRSENMSG,31**

**Example:**

\$PSTMDRSENMSG,31,2827547603,360,-807,-526\*17

Table 9.3-7 \$PSTMDRSENMSG,31 Data Format

Name	Example	Description
Message	\$PSTMDRSENMSG	Reports DR sensor message data, which is specific to the message id for each specific DR sensor configuration.
Message ID	31	3D gyroscope
cpu timestamp	2827547603	microseconds
raw_x	360	360 mdps *4.375 = 1575 mdps = 1.575 dps = 1.575 degree/s
raw_y	-807	-807 mdps *4.375 = 3530.625 mdps = 3.530625 dps = 3.530625 degree/s
raw_z	-526	-526 mdps *4.375 = 2301.25 mdps = 2.30125 dps = 2.30125 degree/s
Checksum	*17	
<CR> <LF>		End of message termination

Note: For unit mdps\*4.375,1dps = 1000mdps = 1000m degree/s

- **\$PSTMDRCAL**

**Example:**

\$PSTMDRCAL,0,1,0,1,ff,1,1,L\*0E

Table 9.3-8 \$PSTMDRCAL Data Format

Name	Example	Description
Message ID	\$PSTMDRCAL	Reports the calibration status of the DR calibration parameters.
dr_is_calib	0	0 = DR isn't fully calibrated 1 = DR is fully calibrated
odo_is_calib	1	0 = Odo scale isn't fully calibrated 1 = Odo scale is calibrated
gyro_gain_is_calib	0	0 = gyro gain isn't fully calibrated 1 = gyro gain is calibrated
gyro_offset_is_calib	1	0 = gyro offset isn't fully calibrated 1 = gyro offset is calibrated
imu_flag	ff	Only showing "ff" or "3f"
gyro_integrity_flag	1	1: gyro signal is healthy , 0: gyro signal is faulty
acc_integrity	1	1: acc signal is healthy , 0: acc signal is faulty
dr_calib_status	L	N: calibration Not available , L: Light calibration , F: Full calibration
Checksum	*0E	
<CR> <LF>		End of message termination

## 9.4 Dead Reckoning (DR) Calibration Process

- **Phase A : Empty Calibrated data if needed**

Input command: \$PSTMGPSSUSPEND< CR><LF>

Input command: \$PSTMNVMITEMINV,80,1< CR><LF>

Input command: \$PSTMNVMITEMINV,81,1< CR><LF>

Input command: \$PSTMSRR< CR><LF>

- **Phase B : Enable DR message Output**

Input command: \$PSTMSETPAR,1228,10800000,0< CR><LF>



Input command: \$PSTMSAVEPAR< CR><LF>

Input command: \$PSTMSRR< CR><LF>

- **Phase C : Check \$PSTMDRCAL message form Step 1 to Step 3**

\$PSTMDRCAL message format like \$PSTMDRCAL,A,B,C,D,....

**Step 1**





**vehicle speed = 0 Km/h**  
**wait GNSS position fixed**  
**time >= 2 min**

\$PSTMDRCAL,0,0,0,**1**,ff,1,1,L\*0E → until field D change to “1” then

**Step 1 finished.**

**Step 2**



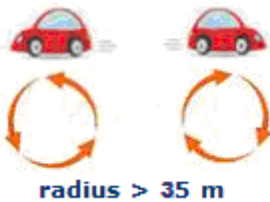
**vehicle speed >= 30 Km/h**  
**drive in straight road sections**  
**time >= 5 min**

\$PSTMDRSENMSG,14,292254154,**35.0**\*04 → check vehicle speed over 30 km/h

\$PSTMDRCAL,0,**1**,0,1,ff,1,1,L\*0E → until field B change to “1” then

**Step 2 finished.**

**Step 3**



**vehicle speed  $\geq$  30 Km/h**  
**drive in circular route**  
**counterclockwise\*3**  
**clockwise\*3**

\$PSTMDRSENMSG,14,292254154,**35.0**\*04 → check vehicle speed over 30 km/h

\$PSTMDRCAL,**1,1,1,1**,ff,1,1,L\*0E → until all field (A,B,C,D) change to “1” then

**Step 3 finished.**

● **Phase D : Disable DR message Output and Finish DR Calibration**

Input command: \$PSTMRESTOREPAR< CR><LF>

Input command: \$PSTMSRR< CR><LF>

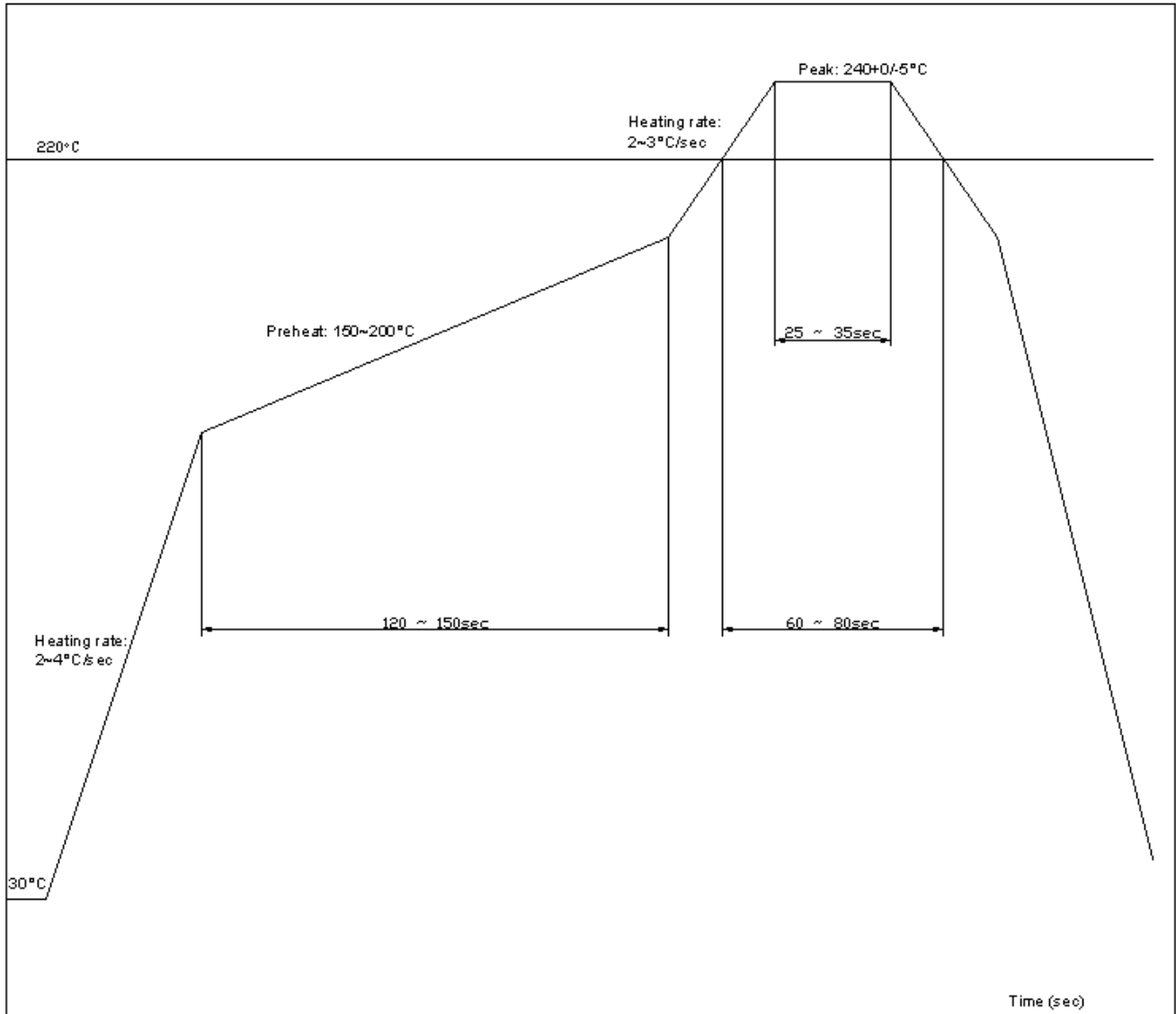
10 Ordering Information

Model	Interface
ST-1612i-DBA	CAN Bus ( ISO 15765-4 )11bit , 250k Bit/s
ST-1612i-DBB	CAN Bus ( ISO 15765-4 )11bit , 500k Bit/s
ST-1612i-DBC	CAN Bus ( ISO 15765-4 ) 29bit , 250k Bit/s
ST-1612i-DBD	CAN Bus ( ISO 15765-4 ) 29bit , 500k Bit/s
ST-1612i-DBE	CAN Bus ( ISO 15765-4 )11bit , 125k Bit/s
ST-1612i-DBO	Odometer signal input
ST-1612i-DBU	Sensor data over UART input

### 11 Recommended soldering reflow profile

The module belongs to RoHS device. The maximum of reflow temperature, real on top of PCB, is not over 240 Celsius.

#### Lead-free Processes

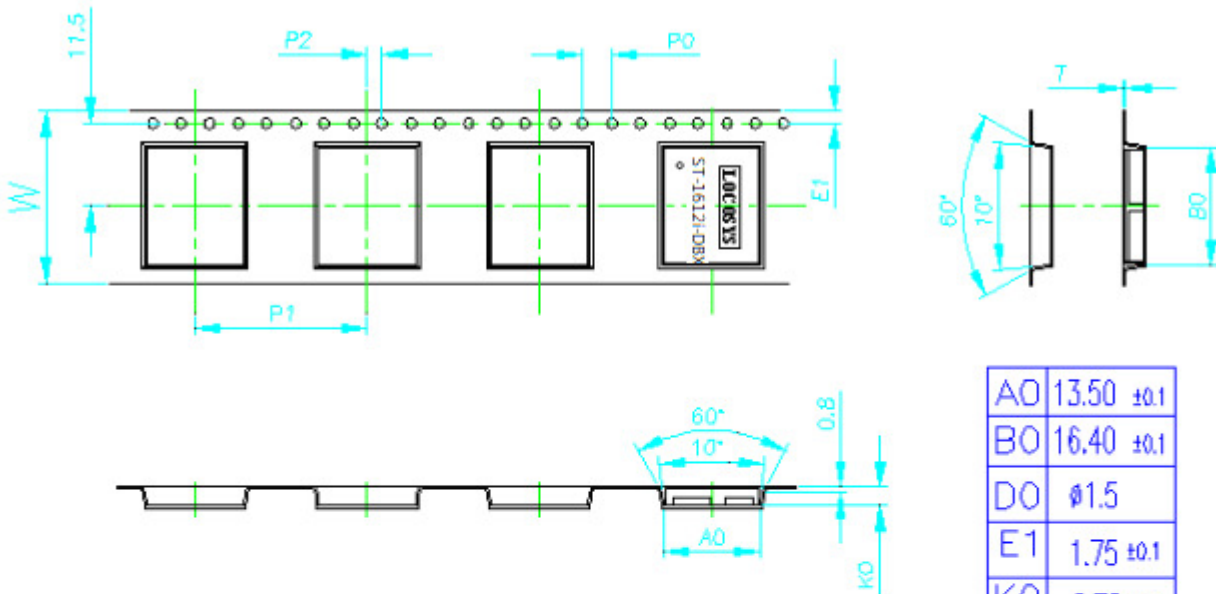


Lead-Free Solder Paste (Sn 96.5-Ag 3.0-Cu 0.5)

Cycle Interval: 300 sec

Note: The ST-1612i-DBX module should be soldered on the topside in the soldering process to prevent from falling down.

## 12 Reel Packing information



A0	13.50 ±0.1
B0	16.40 ±0.1
D0	∅1.5
E1	1.75 ±0.1
K0	2.70 ±0.1
P0	4.0 ±0.1
P1	24.00 ±0.1
P2	2.0 ±0.10
T	0.3 ±0.10
W	24.0 ±0.30

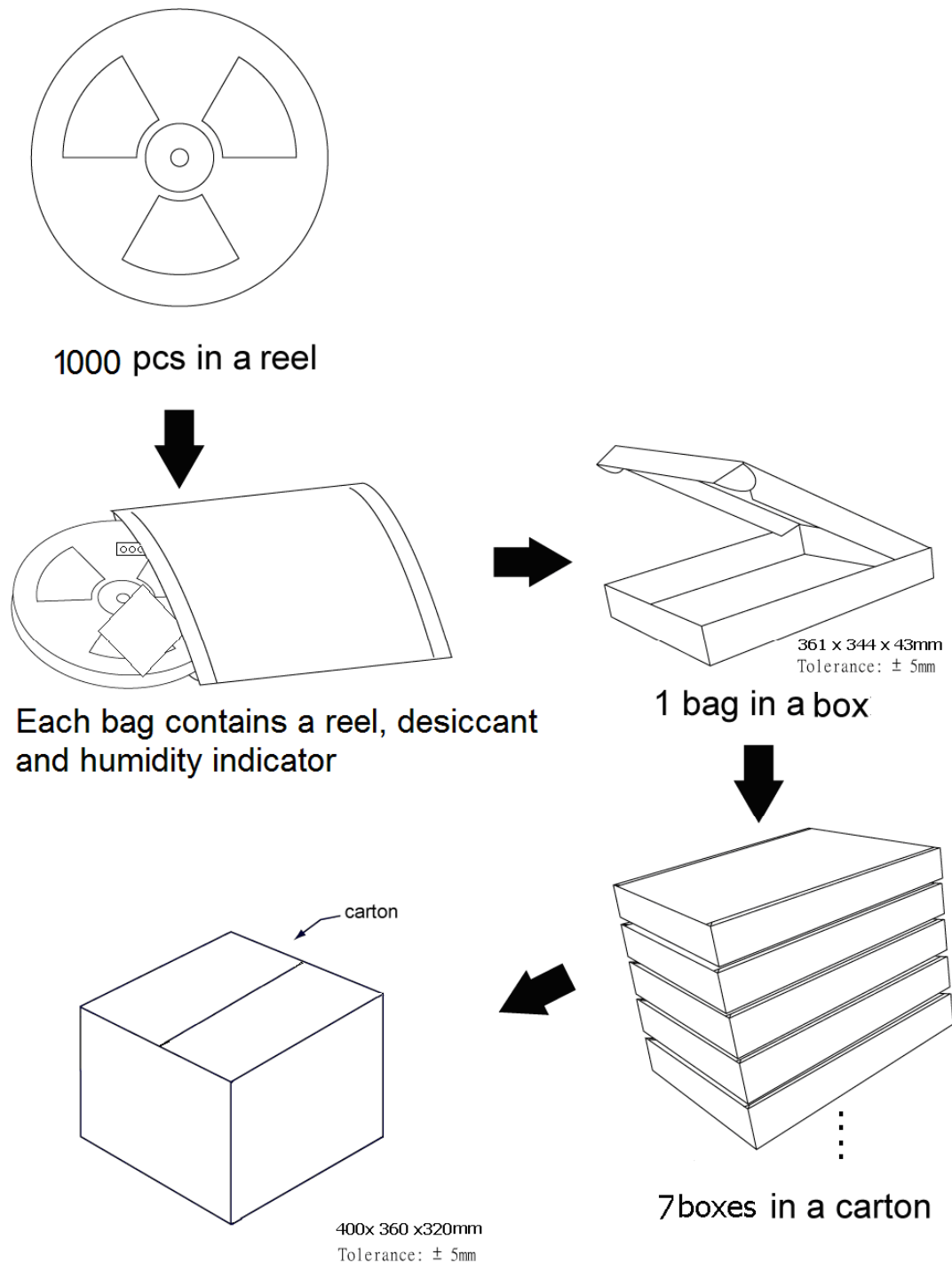
1. 10 sprocket hole pitch cumulative tolerance  $\pm 0.2$
2. Camber not to exceed 1mm in 100mm
3. A0 and B0 measured on a plane 0.3mm above the bottom of the pocket
4. K0 measured from a plane on the inside bottom of the pocket to the top surface of the carrier .
5. pocket position relative to sprocket hole measured as true position of pocket, not pocket hole.
6. Component load per 13" reel: 1000 pcs
7. Packing length per 22" reel: 75 M



## 13 Packing and Handling

GNSS modules, like any other SMD devices, are sensitive to moisture, electrostatic discharge, and temperature. By following the description sketched in the document for LOCOSYS GNSS module storage and handling, it is possible to reduce the chances of them being damaged during production.

### 13.1 Packing



## 13.2 Moisture Sensitivity

The module belongs to moisture sensitive device (IPC/JEDEC J-STD-020C Level III). If it is not used by then, we strong recommended storing the GNSS modules in dry places such as dry cabinet. The approximate shelf life for LOCOSYS GNSS modules packages is 6 months from the bag seal date, when store in a non-condensing storage environment (<30°C/60% RH)

## 13.3 ESD Handling



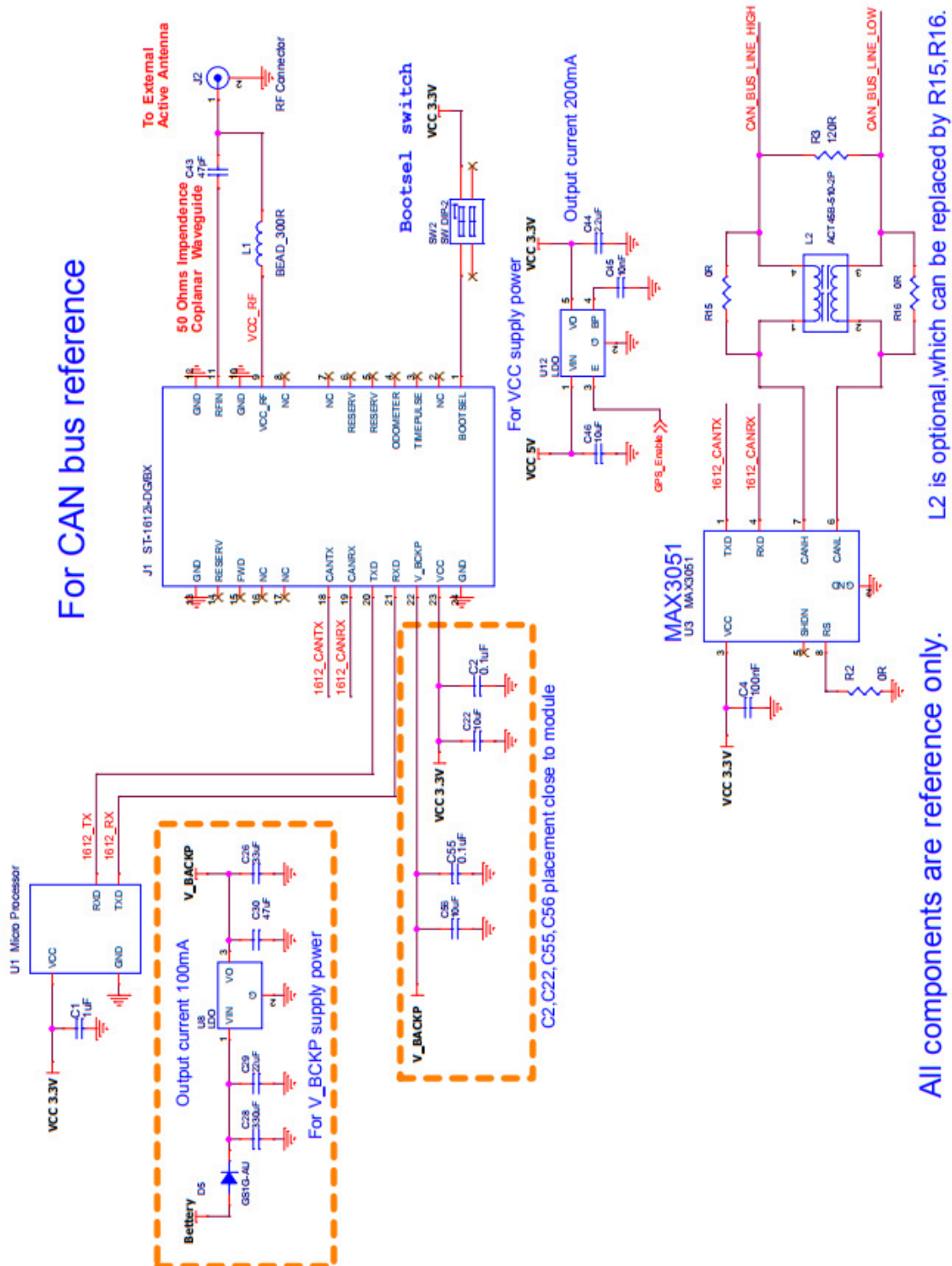
**Please carefully follow the following precautions to prevent severe damage to GNSS modules.**

LOCOSYS GNSS modules are sensitive to electrostatic discharges, and thus are Electrostatic Sensitive Devices (ESD). Careful handling of the GNSS modules and in particular RFIN pin must follow the standard ESD safety protections:

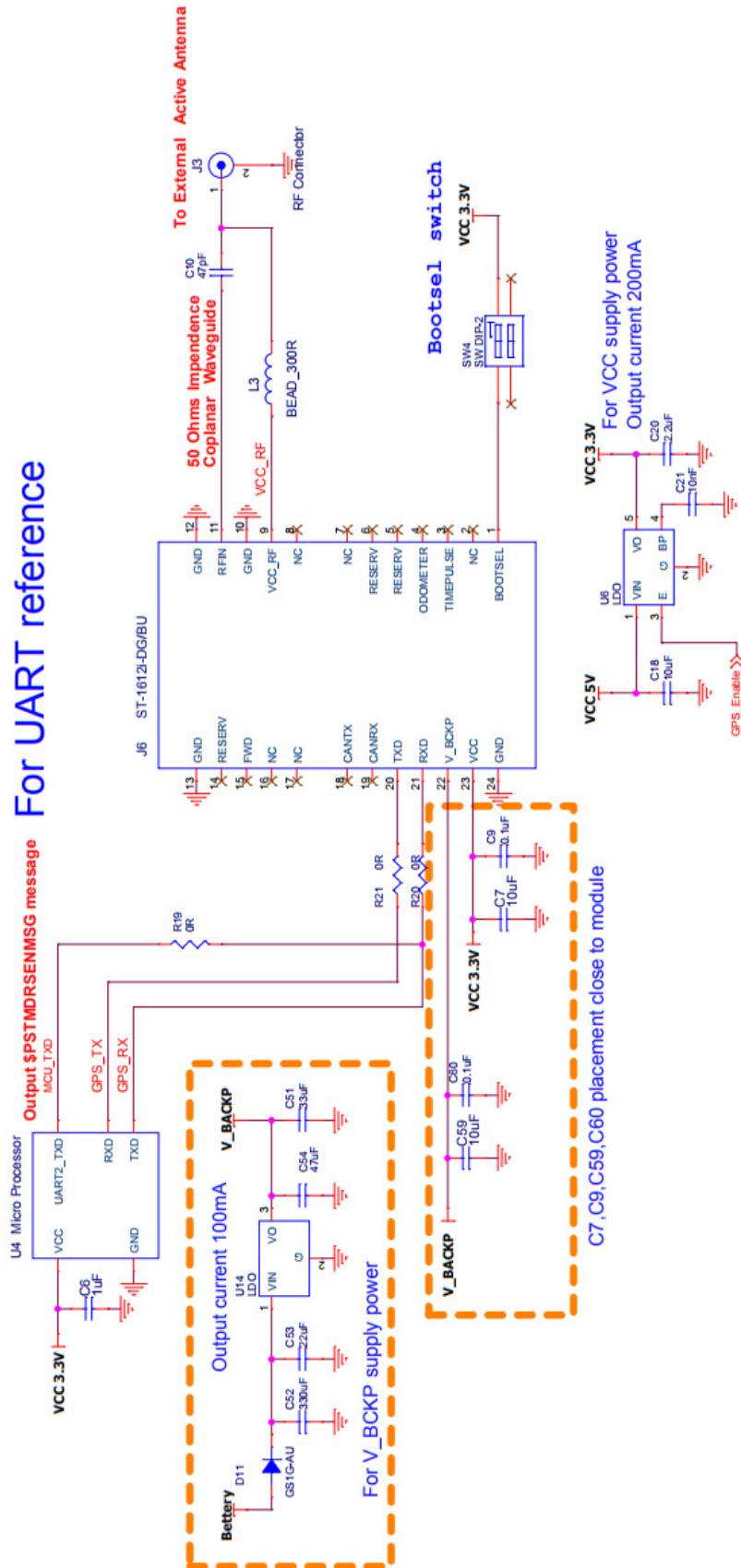
- Unless there is a galvanic coupling between the local GND and the PCB GND, then the first point of contact when handling the PCB shall always be between the local GND and PCB GND.
- Before working with RFIN pin, please make sure the GND is connected.
- When working with RFIN pin, do not contact any charges capacitors or materials that can easily develop or store charges such as patch antenna, coax cable, soldering iron.
- When soldering RFIN pin, please make sure to use an ESD safe soldering iron (tip).



14.2 For CAN Bus Reference



14.3 For UART input Reference



For UART reference

All components are reference only.

## 15 Odometer Specification

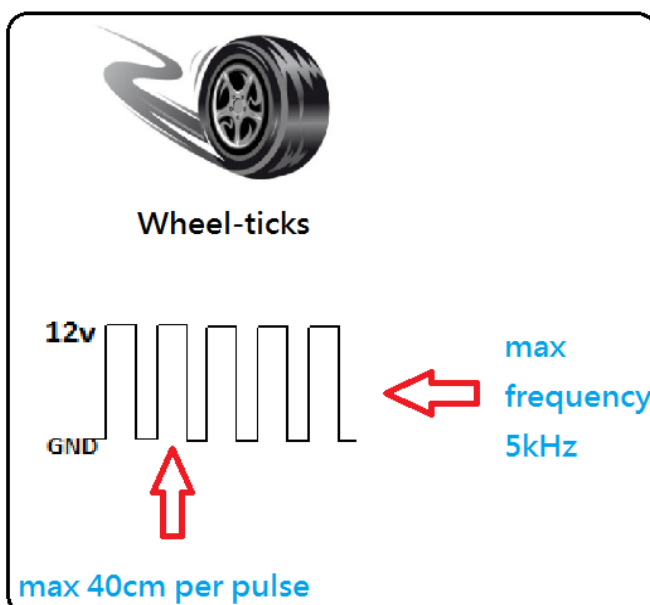
### 15.1 Introduction

The odometer provides information on traveled ground distance (in meter) using solely the position and Doppler-based velocity of the navigation solution. For each computed traveled distance since the last odometer reset, the odometer estimates a 1-sigma accuracy value. The total cumulative ground distance is maintained and saved in the NVM memory.

### 15.2 WHEEL-TICK description

The wheel-tick input, also known as HW interface, is used to provide speed pulse (Wheel tick) information to ST-1612i-DGO/DBO modules. By default the wheel-tick count is based on the rising edge of the wheel-tick pulse signal. To improve performance with lower rate mechanically derived wheel-tick signals, the receiver may be configured to use both the rising and falling edges of the wheel-ticks signal on the condition that the wheel-tick pulses have approximately 1:1 mark: space ratio regardless of speed. The maximum recommended pulse frequency is 5KHz.

The pulse interval (wheel-ticks resolution) should be less than 40cm per tick over distance traveled. There is no minimum distance. The wheel tick output shall change linearly with the change in speed (navigation filter estimates only the linear scale factor ). If the vehicle is standing still , there should be no wheel tick pulses. This is particularly important at system shut down and power up. If there is a dead band (wheel tick pulse does not change or is not output below a certain speed), performance will be affected at low speed.



"Max 40cm per pulse" refers to a pulse is a maximum of 40cm within a single tire diameter size, when the tire turn a circle, will return a pulse into the speed of the calculus, if the tire more than 40cm, will continue for 2 pulse Return calculus.

## 16 Sensor over UART Reference

### 16.1 Operating Instructions

- (1) The user needs to get two kinds of data. One is the **Vehicle Speed** and the other is **Direction of Vehicle** (Forward or Backward).
- (2) Vehicle speed command: `$PSTMDRSENMSG,1,0,<Odometer>\r\n`
  - `<Odometer>` is for vehicle speed
  - `\r\n` is for carriage return and line feed
- (3) Direction of travel: `$PSTMDRSENMSG,2,0,<FwdRev>\r\n`
  - `<FwdRev>` is for Direction of Vehicle, forward=0, backward=1
  - `\r\n` is for carriage return and line feed
- (4) Programming description: Take below for example.
  - unsigned short u16Spd=0;
  - unsigned char u8FwdRev=0;
  - unsigned int u32OdoCounter=0;

**//Get Vehicle Speed**

u16Spd=GetVehicleSpeed;

**//Get Direction of Vehicle**

**//The setting is 0, if you cannot get Direction of Vehicle.**

u8FwdRev=GetDirectionOfVehicle;

**//Set up the counter** (This setting mode is the measured value of LOCOSYS.)

u32OdoCounter=u32OdoCounter+(u16Spd\*4);

if(u32OdoCounter>99999) u32OdoCounter=0;

**//The initial value of the Direction of Vehicle is recommended to be set up as “0”. It only needs to be transmitted when the Direction of Vehicle changes.**

printf("\$PSTMDRSENMSG,2,0,%d\r\n",u8FwdRev);

**//It is recommended to transmit 5 times per second (5Hz)**

printf("\$PSTMDRSENMSG,1,0,%d\r\n",u32OdoCounter);

## Document change list

### Revision 1.0

- First release on Feb.26.2018

### Revision 1.1 (Aug.03.2018)

- Remove Support tow alert on page 1.
- Remove interrupt signal from system block diagram on page 2.
- Remove Pin 2 interrupt signal on page 4.
- Modify Backup Battery Current on page 6.
- Remove interrupt signal from reference circuit on page 20 ~ 22.

### Revision 1.2 (Dec.10.2018)

- Add AEC-Q100 qualified for automotive grade on page 1.

### Revision 1.3 (Jan.14.2019)

- Modify product's appearance photo on page 1.
- Add standard specifies for SAE J2945/1 on page 1.
- Modify Supply Current - Tracking and Backup Battery Current on page 6.
- Modify Recommended Mounting photo on page 7.

### Revision 1.4 (Mar.15.2019)

- Modify pin1 description on page 4 ~ 5.
- Modify reference circuit on page 20 ~ 22.

### Revision 1.5 (May.03.2019)

- Modify Protocol Support on page 3.
- Add proprietary output sentence on page 14 ~ 15.

### Revision 1.6 (Jul.16.2019)

- Modify Proprietary NMEA and DR input/output message on page 14 ~ 17.
- Add Dead Reckoning (DR) output message on page 18 ~ 20.
- Add Dead Reckoning (DR) Calibration Process on page 21 ~ 22.

### Revision 1.7 (Jul.31.2019)

- Modify section 7.1 Recommended Mounting on page 7.
- Modify section 13.1 Reference Circuit on page 27.
- Add section 14 Odometer Specification on page 30.
- Add section 15 Sensor over UART Reference on page 31.



Revision 1.8 (Oct.20.2020)

- Added application description in section 3.