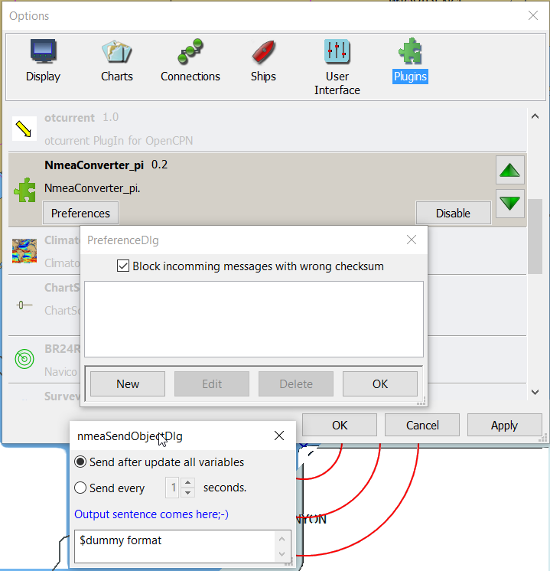
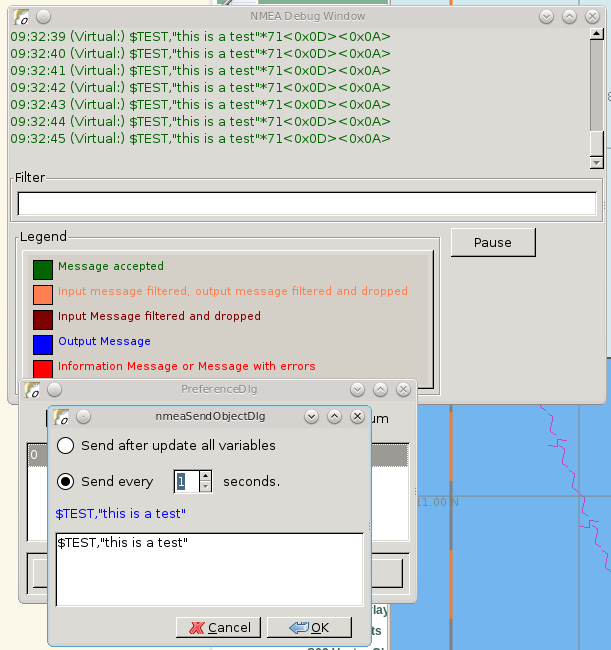
**NMEA Converter**

[**http://opencpn.org/ocpn/Plugins\_external\_nema-converter**](http://opencpn.org/ocpn/Plugins_external_nema-converter) **NMEA Converter can convert incoming NMEA Sentences.**

**Be aware you need at least some knowledge of** [**NMEA Sentences**](http://opencpn.org/ocpn/Basic_data-connections_nmea-sentences) **.**

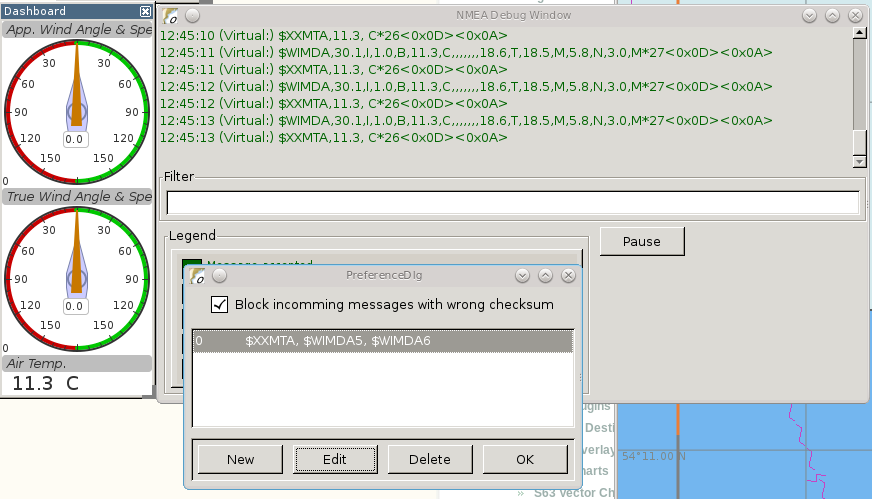
Install the plugin in accord with the [Install and Enable](http://opencpn.org/ocpn/toolbar_plugins_install-enable) Instructions.

Then open the Preference Dialog to set your conversions.

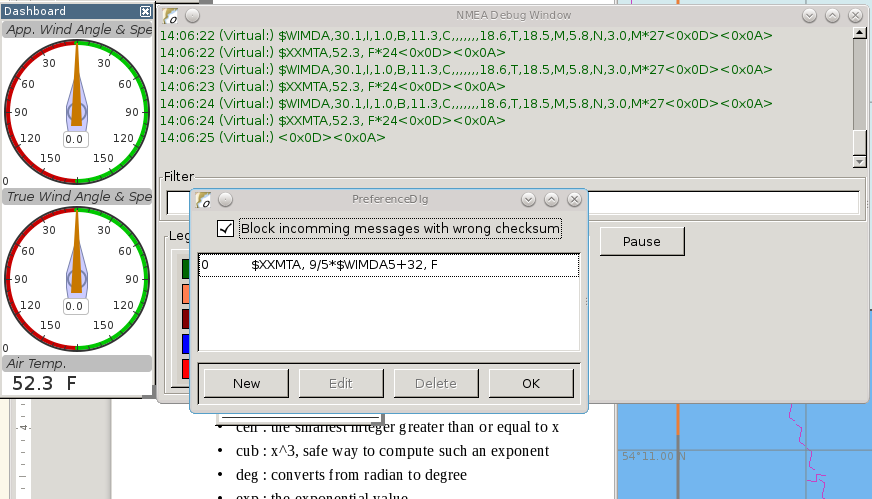
  
If you see something like above please enlarge the nmeaSendDlg window so it will show some buttons.  
Lets make a first test.[](http://opencpn.org/ocpn/sites/default/files/users/rooiedirk/testmessage1.png)  
This is a completely useless sentence that is send every second to OpenCPN. Check also the nmeadebug window in the picture above, and note that a checksum is added to the sentence.  
It is easy to change this one into something like:  
$GPRMC,085437,A,5324.3052,N,00611.5704,E,3.3,68.9,070614,0.3,E,A  
That will fake a GPS position into OpenCPN, but still not very usefull.

### ****Use incoming data to send a different NMEA sentence.****

The real power of this plugin is that you can change the data of the send sentence by cherrypicking from incoming sentences.  
The reason I did write this plugin was I wanted dashboard to show the airtemperature.  To get the data, dashboard needs a $xxMTA sentence, however I had a Airmar PB200 unit that is sending $xxMDA sentences.  
An example: We have from the windset,  
$WIMDA,30.1,I,1.0,B,11.3,C,,,,,,,18.6,T,18.5,M,5.8,N,3.0,M\*27   
and want:  
$xxMTA,11.3,C\*<Checksum>  
Note that the data needed in in fields 5 and 6 of the MDA sentence. (11.3 degrees Celcius)  
Setting the format to,  
***$XXMTA, $WIMDA5, $WIMDA6***  
will give an output sentence: ***$XXMTA, 11.3, C\*26***  
Where **$WIMDA5** is replaced by the 5th field of the $WIMDA sentence. etc.

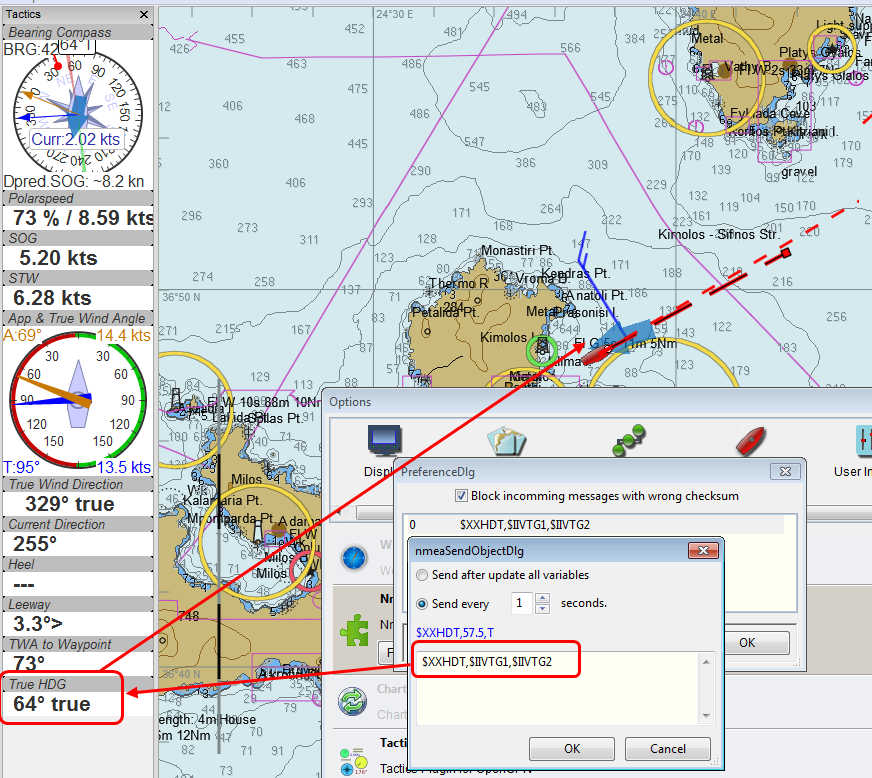
  
As seen above, the $XXMTA sentece is send each time a $WIMDA sentence is received, an read by dashboard.

### Do math - Convert units.   EG: Centigrade to Fahrenheit

But what if I was living in the US and prefered the reading to be in degrees Fahrenheit??  
Just change the format to:  
***$XXMTA, 5 / 9 \* $WIMDA5 + 32, F  
[](http://opencpn.org/ocpn/sites/default/files/users/rooiedirk/xxmtamessage3.png)***  
As you see the value is recalculated and now displayed as degrees Fahrenheit.

### 

### Force True Heading to COG

In NMEAconverter create a HDG (compass) sentence from your COG.  
Possible use when the autopilot is old and does not recieve heading (true or magnetic).  
  
Note: Since computation of the drift and surface current requires HDG, it is understood that while using Tactics\_pi value for Current speed and direction will not be useful. All you will see is the speed difference between SOG and STW, which would force Current to be the vector component normal to the keel. That means the "current" display will always be directly from bow or stern, in other words the component of the current along the centerline of the boat, assuming your STW(speed through water) is calibrated correctly.  
See [Tactics\_pi thread in Cruiser Forum  
  
[](http://www.cruisersforum.com/forums/f134/tactics-plugin-166909.html#post2140937)](http://www.cruisersforum.com/forums/f134/tactics-plugin-166909.html#post2140937)  
Tactics\_pi and NMEAconverter\_pi shown with    $XXHDT,$IIVTG1,$IIVTG2  
  
  
Remember that setting HDT to COG will give you no drift angle at all (except maybe a small portion via the manual heel input), so you will not see any real [current](http://www.cruisersforum.com/forums/tags/current.html) display.  
The only thing that remains is the speed difference between STW and SOG, which means that you will get only the vector of [current](http://www.cruisersforum.com/forums/tags/current.html) along the [keel](http://www.cruisersforum.com/forums/tags/keel.html) line of your boat.  
  
I suggest to set a fixed heel angle to 0 (see screenshot above), to avoid side effects with manual heel input at very low speeds and small speed differences between SOG and STW

### ****Calculate True Wind from Apparent Wind****

**Known:**  
Boat Speed  ground (kts) = SOG  
Apparent wind speed (degrees) = AWS  
Apparent wind angle (degrees) = AWA

**MWV - Wind Speed and Angle - Input**

$--MWV,x.x,a,x.x,a\*hh<CR><LF>

Field Number:

1. Wind Angle, 0 to 360 degrees
2. Reference, R = Relative, T = True
3. Wind Speed
4. Wind Speed Units, K/M/N
5. Status, A = Data Valid
6. Checksum

**RMA - Recommended Minimum Navigation Information - Input**

$--RMA,A,llll.ll,a,yyyyy.yy,a,x.x,x.x,x.x,x.x,x.x,a\*hh<CR><LF>

Field Number:

1. Blink Warning
2. Latitude
3. N or S
4. Longitude
5. E or W
6. Time Difference A, uS
7. Time Difference B, uS
8. Speed Over Ground, Knots
9. Track Made Good, degrees true
10. Magnetic Variation, degrees
11. E or W
12. Checksum

**MWV - Wind Speed and Angle - Output TW in Knots**

$--MWV,x.x,T,x.x,K\*hh<CR><LF>

Field Number:

1. Wind Angle, 0 to 360 degrees
2. Reference, R = Relative, T = True
3. Wind Speed
4. Wind Speed Units, K/M/N
5. Status, A = Data Valid
6. Checksum

Nmea0183 from http://www.catb.org/gpsd/NMEA.html

**Calculations Needed:**  
x =AWS\*sin(AWA\*3.14159265/180)  
y =(x/tan(AWA\*3.14159265/180)) - SOG  
TWS (True Wind Speed) =x/sin(TWA\*3.14159265/180)  
TWA (True Wind Angle) =atan(x/y)\*180/3.14159265  
\*Formula from <http://www.bethandevans.com/calculators.htm>  
  
The formula now needs to be completed:  
**$XXMWV,**

### Contribute your Formulas & Conversions

If you have an usefull format formula pls. feel free to add it here .

Another future possibility is to do a comparison lookup of a boat''s Polar data and send a special NMEA sentence of the comparative result to the Dashboard using a special dedicated performance instrument, for example.

### Calculations Supported

For calculations the following functions are suported:

#### Normal functions

abs : the absolute value, it removes the sign  
ceil : the smallest integer greater than or equal to x  
cub : x^3, safe way to compute such an exponent  
deg : converts from radian to degree  
exp : the exponential value  
hvi : Heaviside's function, =0 if x<0, =1 if x>=0  
int : the largest integer less than or equal to x  
inv : inverts, 1/x  
floor : not supported, see int  
ln : the neperian logarithm  
lnep : see ln  
log : the logarithm base 10  
rad : converts from degree to radian  
sgn : the sign, -1 or +1. Zero returns +1.  
sqr : x^2, safe way to compute such an exponent  
sqrt : the root square

#### Trigonometric functions

acos : the secant (inverse of cosine)  
acsh : the inverse of the hyperbolic cosine  
asin : the cosecant (inverse of sine)  
asnh : the inverse of the hyperbolic sine  
atan : the cotangent (inverse of tangent)  
atnh : the inverse of the hyperbolic tangent  
cos : the cosine  
cosh : the hyperbolic cosine  
sin : the sine  
sinc : the cardinal sine (radian)  
sinh : the hyperbolic sine  
tan : the tangent  
tanh : the hyperbolic tangent  
  
This makes it possible to do spreadsheel like calculations. And as in a spreadsheet the trigonometric functions are using radians.