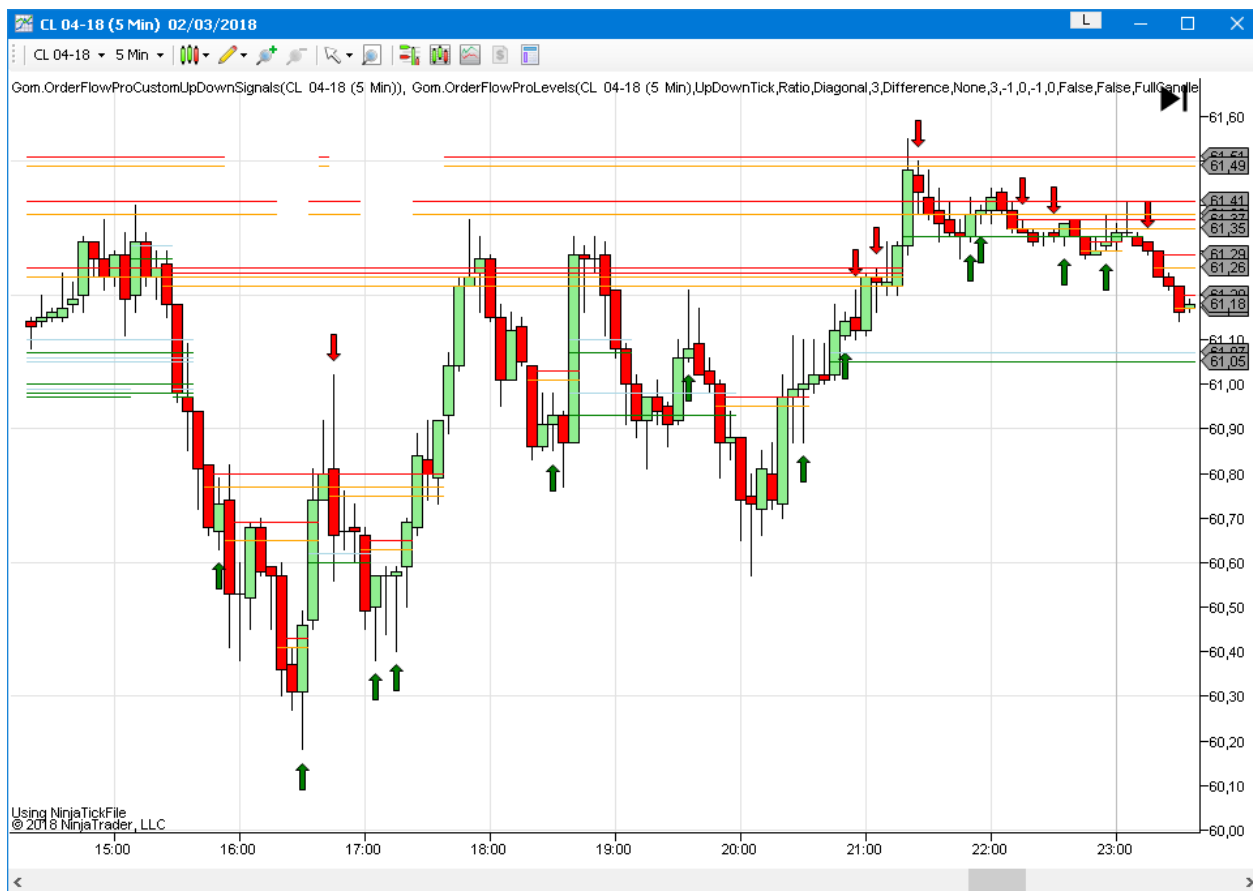


# GOMORDERFLOW PRO ADDITIONAL INDICATORS

## DOCUMENTATION OF ORDERFLOW ADDITIONAL INDICATORS

Version 2.3

Copyright Gomex 2016,2017,2018



# GOMORDERFLOWPRO-LEVELS

## DESCRIPTION

This indicator exposes levels from the orderflow as plots, these levels can be called form custom indicators & strategies

Each exposed level has 5 associated plots, they are ordered from nearest to farthest from current market close

Example with NakedPOC



## SETTINGS

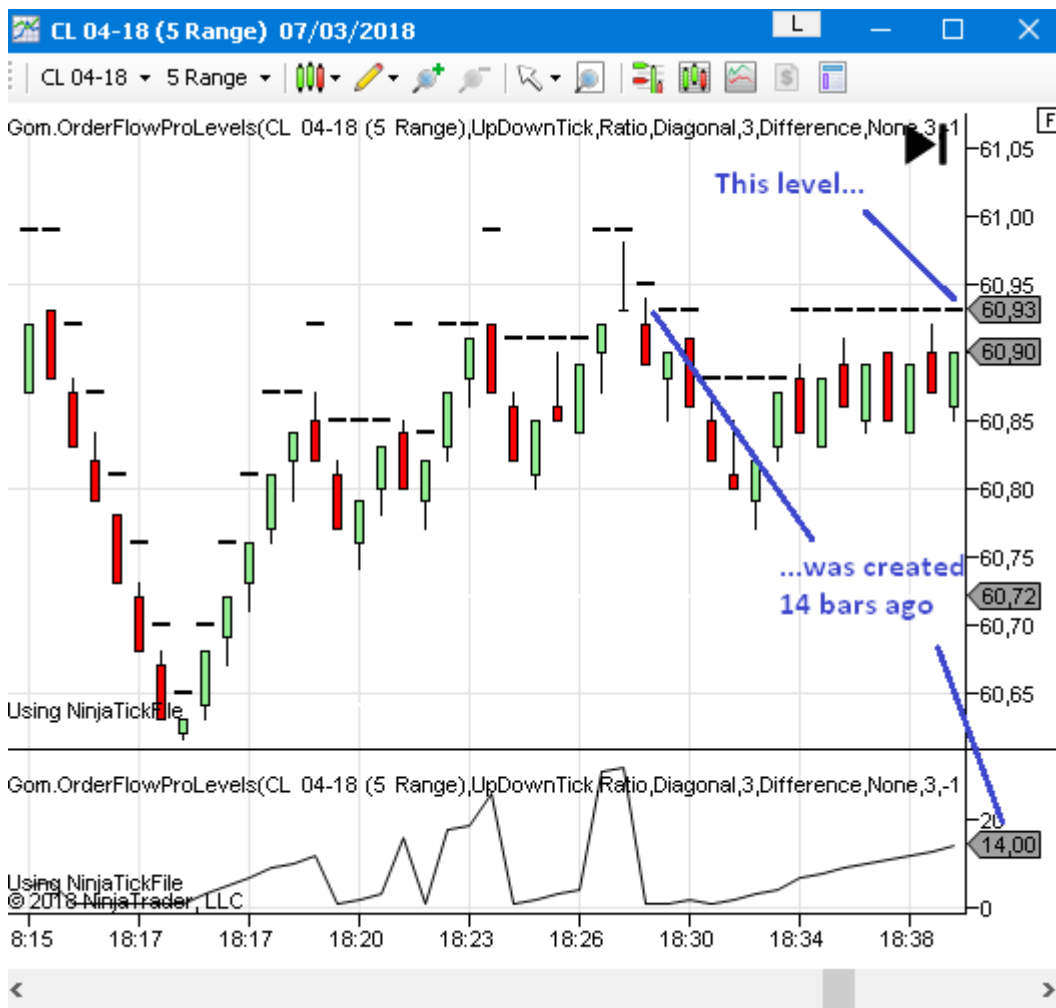
Same as Orderflowpro, see the relevant doc.

Added : **Plot POC** , **Plot SR Zones**, **Plot Unfinished Auctions** to select plots to plot.

## EXPOSED PLOTS

**NakedPOCHighX** : the 5 closest naked POCs above current close.

**NakedPochHighBarsAgoX** : shows how many bars ago the level N° X was created



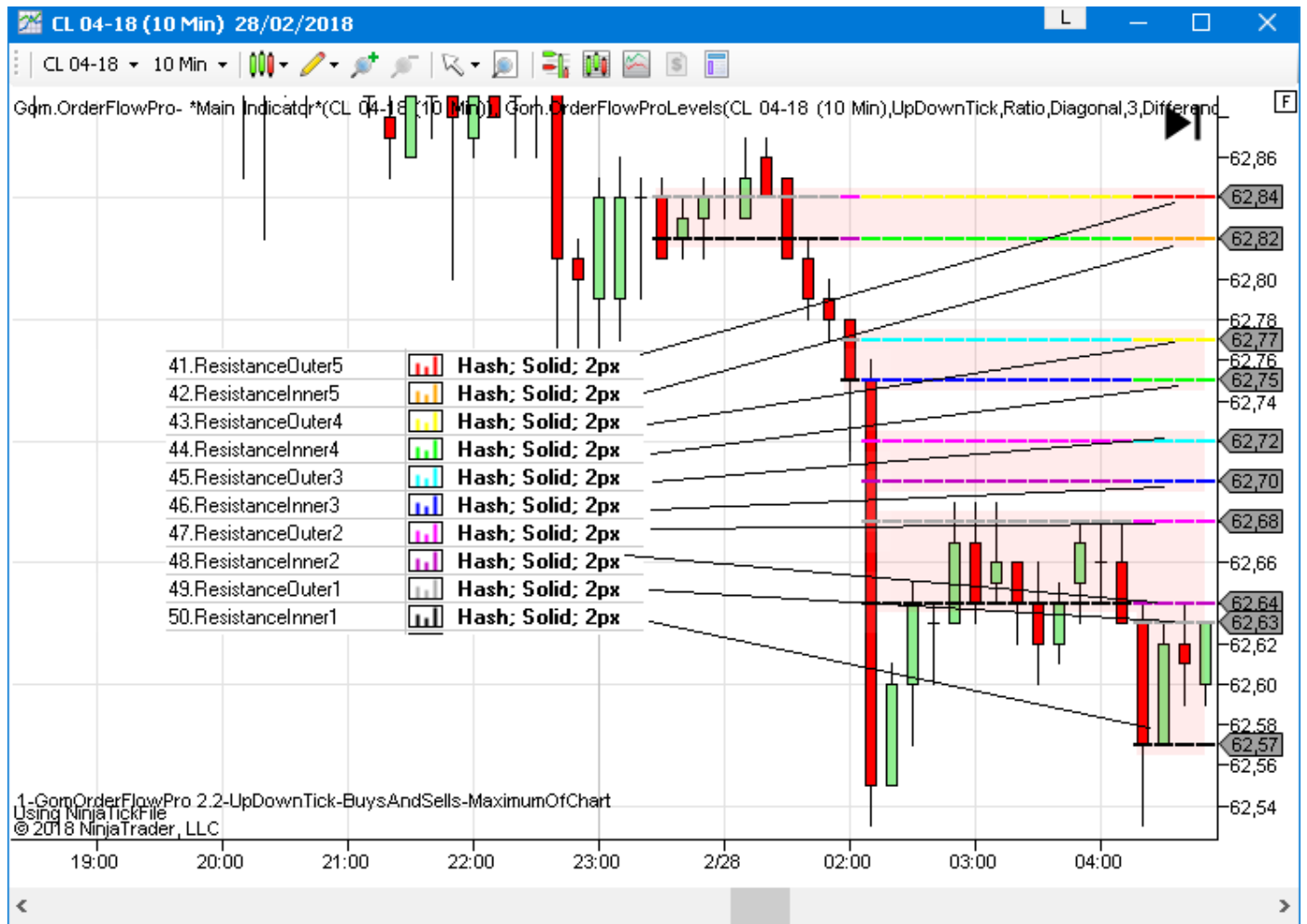
**NakedPOCLowX**, **NakedPOCLowBarsAgoX** :

Same with low POCs

**NakedUAHighX**, **NakedUAHighXBarsAgo**, **NakesUALowX**, **NakesUALowXBarsAgo** :

Same with naked Unfinished Auctions

**ResistanceInnerX,ResistanceOuterX** : 5 closest resistance zones



**SupportInnerX,SupportOuterX** : same with supports

## CONSTRUCTOR NT7

```
GomOrderFlowProLevels Strategy.GomOrderFlowProLevels(DeltaCalcType deltaMode, ImbalanceComparisonType imbalanceCriterion1Comparison, ImbalanceDirectionType imbalanceCriterion1Direction, double imbalanceCriterion1Value, ImbalanceComparisonType imbalanceCriterion2Comparison, ImbalanceDirectionType imbalanceCriterion2Direction, double imbalanceCriterion2Value, int imbalanceMaxLosingSideVolume, int imbalanceMinWinningSideVolume, int maxTickSize, int minTickSize, bool resetNakedOnSessionStart, bool sRCancelOnNewSession, SRZoneEndConditionType sRExit, int sRHiLoThreshold, int sRMinTotalVolume, int sRNbLevels, int ticksPerPriceLevel)
```

Example (check GomOFNakedPOCSampleStrat)

```
gofl=GomOrderFlowProLevels (Gom.DeltaLib.DeltaCalcType.UpDownTick,
    Gom.OrderFlow.ImbalanceComparisonType.Ratio,Gom.OrderFlow.ImbalanceDirectionType.D
    iagonal,3,Gom.OrderFlow.ImbalanceComparisonType.Ratio,Gom.OrderFlow.ImbalanceDirec
    tionType.None, 3, -1, 0, -1,0,false, false,
    Gom.OrderFlow.SRZoneEndConditionType.FullCandleCross, 1, 0, 3, 1);
```

## CONSTRUCTOR NT8

```
GomOrderFlowProLevels Strategy.GomOrderFlowProLevels(DeltaCalcType deltaMode, int maxTickSize, int minTickSize, int ticksPerPriceLevel, bool resetNakedOnSessionStart, ImbalanceDirectionType imbalanceCriterion1Direction, ImbalanceComparisonType imbalanceCriterion1Comparison, double imbalanceCriterion1Value, ImbalanceDirectionType imbalanceCriterion2Direction, ImbalanceComparisonType imbalanceCriterion2Comparison, double imbalanceCriterion2Value, int imbalanceMinWinningSideVolume, int imbalanceMaxLosingSideVolume, int sRNbLevels, int sRMinTotalVolume, SRZoneEndConditionType sRExit, int sRHiLoThreshold, bool sRCancelOnNewSession)
```

Example (check GomOFNakedPOCSampleStrat)

```
gof= GomOrderFlowProLevels (Close, Gom.DeltaLib.DeltaCalcType.BidAsk, -1, 0, 1,
    false,Gom.OrderFlow.ImbalanceDirectionType.Diagonal,
    Gom.OrderFlow.ImbalanceComparisonType.Ratio, 3,
    Gom.OrderFlow.ImbalanceDirectionType.None,
    Gom.OrderFlow.ImbalanceComparisonType.Ratio, 3,0,-1, 3, 0,
    Gom.OrderFlow.SRZoneEndConditionType.FullCandleCross, 1, false);
```

## GOMORDERFLOWPRO-VALUES

This indicator exposes values and raw ladder data (**NEW in 2.3**) from the orderflow, which can be used in custom indicators or strategies

## SETTINGS

Same as Orderflowpro, see the relevant doc.

## EXPOSED PLOTS

`Delta` : Delta of the bar  
`Buys` : Total Buys of the Bar  
`Sells` : Total Sells of the Bar  
`TotalVolume` : Total Volume of the Bar  
`MinDelta` : Minimum value reached by the delta on bar creation  
`MaxDelta` : Maximum value reached by the delta on bar creation  
`HighCOT` : Current high COT  
`LowCOT` : Current low COT  
`Finish` : Delta Finish  
`POCPrice` : POC of current bar.  
`POCVolume` : Volume at the POC  
`IsHighUA` : 1 if we have an Unfinished Auction on High of Bar  
`IsLowUA` : 1 if we have an Unfinished Auction on Low of Bar  
`NbBuyImbalances` : Number of Buy Imbalances on the bar  
`VolBuyImbalances` : Total volume of Buy Imbalances  
`NbSellImbalances` : Number of Sell Imbalances  
`VolSellImbalances` : Volume of Sell Imbalances  
`NbResistZones` : Number of Resistance Zones created in the bar  
`NbSupportZones` : Number of Support Zones created in the bar  
`IsBuyingReversal` : 1 if there is a buy reversal signal  
`IsSellingReversal` : 1 if there is a sell reversal signal  
`IsSupportZoneOnHigh` : 1 if a support zone was created on the high of the bar  
`IsResistanceZoneOnLow` : 1 if a resist zone was created on the low of the bar.  
`IsSellImbalanceOnLow` : 1 if there is a sell imbalance on the low of the bar  
`IsBuyImbalanceOnLow` : : 1 if there is a buy imbalance on the low of the bar  
`IsSellImbalanceOnHigh` : 1 if there is a sell imbalance on the high of the bar  
`IsBuyImbalanceOnHigh` : 1 if there is a buy imbalance on the high of the bar  
`RatioHigh` : Ratio of the 2 highest buys  
`RatioLow` : Ratio of the 2 lowest sells.  
`SellsAtHigh` : Volume of Sells at top of bar  
`BuysAtHigh` : Volume of Buys at top of bar  
`SellsAtLow` : Volume of Sells at bottom of bar  
`BuysAtLow` : Volume of Buys at bottom of bar

## CONSTRUCTOR NT7

```
GomOrderFlowProValues Strategy.GomOrderFlowProValues(bool decliningTotVolSmallerThanOppositeSideTotVol, DeltaCalcType deltaMode, ImbalanceComparisonType imbalanceCriterion1Comparison, ImbalanceDirectionType imbalanceCriterion1Direction, double imbalanceCriterion1Value, ImbalanceComparisonType imbalanceCriterion2Comparison, ImbalanceDirectionType imbalanceCriterion2Direction, double imbalanceCriterion2Value, bool imbalanceInDecliningZoneAuthorized, int imbalanceMaxLosingSideVolume, int imbalanceMinWinningSideVolume, int maxTickSize, int minNumOfDecliningLevels, int minSizeOfCounterSideImbalance, int minTickSize, ReversalPocPositionType reversalPocPosition, int sRMinTotalVolume, int sRNbLevels, int ticksPerPriceLevel)
```

Example (check GomOFDeltaSampleStrat)

```
gof= GomOrderFlowProValues (true, Gom.DeltaLib.DeltaCalcType.UpDownTick, Gom.OrderFlow.ImbalanceComparisonType.Ratio, Gom.OrderFlow.ImbalanceDirectionType.Diagonal, 3, Gom.OrderFlow.ImbalanceComparisonType.Ratio, Gom.OrderFlow.ImbalanceDirectionType.None, 3, false, -1, 0, -1, 3, 1, 0, Gom.OrderFlow.ReversalPocPositionType.Anywhere, 0, 3, 1);
```

## CONSTRUCTOR NT8

```
GomOrderFlowProValues Strategy.GomOrderFlowProValues(DeltaCalcType deltaMode, bool forceIntrabarCalc, int maxTickSize, int minTickSize, int ticksPerPriceLevel, ImbalanceDirectionType imbalanceCriterion1Direction, ImbalanceComparisonType imbalanceCriterion1Comparison, double imbalanceCriterion1Value, ImbalanceDirectionType imbalanceCriterion2Direction, ImbalanceComparisonType imbalanceCriterion2Comparison, double imbalanceCriterion2Value, int imbalanceMinWinningSideVolume, int imbalanceMaxLosingSideVolume, int sRNbLevels, int sRMinTotalVolume, int minNumOfDecliningLevels, bool imbalanceInDecliningZoneAuthorized, int minSizeOfCounterSideImbalance, ReversalPocPositionType reversalPocPosition, bool decliningTotVolSmallerThanOppositeSideTotVol)
```

gof=

Example (check GomOFDeltaSampleStrat)

```
GomOrderFlowProValues (Gom.DeltaLib.DeltaCalcType.BidAsk, false, -1, 0, 1, Gom.OrderFlow.ImbalanceDirectionType.Diagonal, Gom.OrderFlow.ImbalanceComparisonType.Ratio, 3, Gom.OrderFlow.ImbalanceDirectionType.None, Gom.OrderFlow.ImbalanceComparisonType.Ratio, 3, 0, -1, 3, 0, 3, false, 1, Gom.OrderFlow.ReversalPocPositionType.InDecliningSideHalfOfBar, true);
```

## RAW LADDER DATA API (NEW IN 2.3)

This is intended for advanced developers.

The Values indicator also exposes raw data (buys and sells on each price level) allowing the coding of custom indicators using the individual buy/sell data.

The exposed property is a collection named *BarData*, each individual bar is accessed using *BarData[barsago]* like any other Ninja time series.

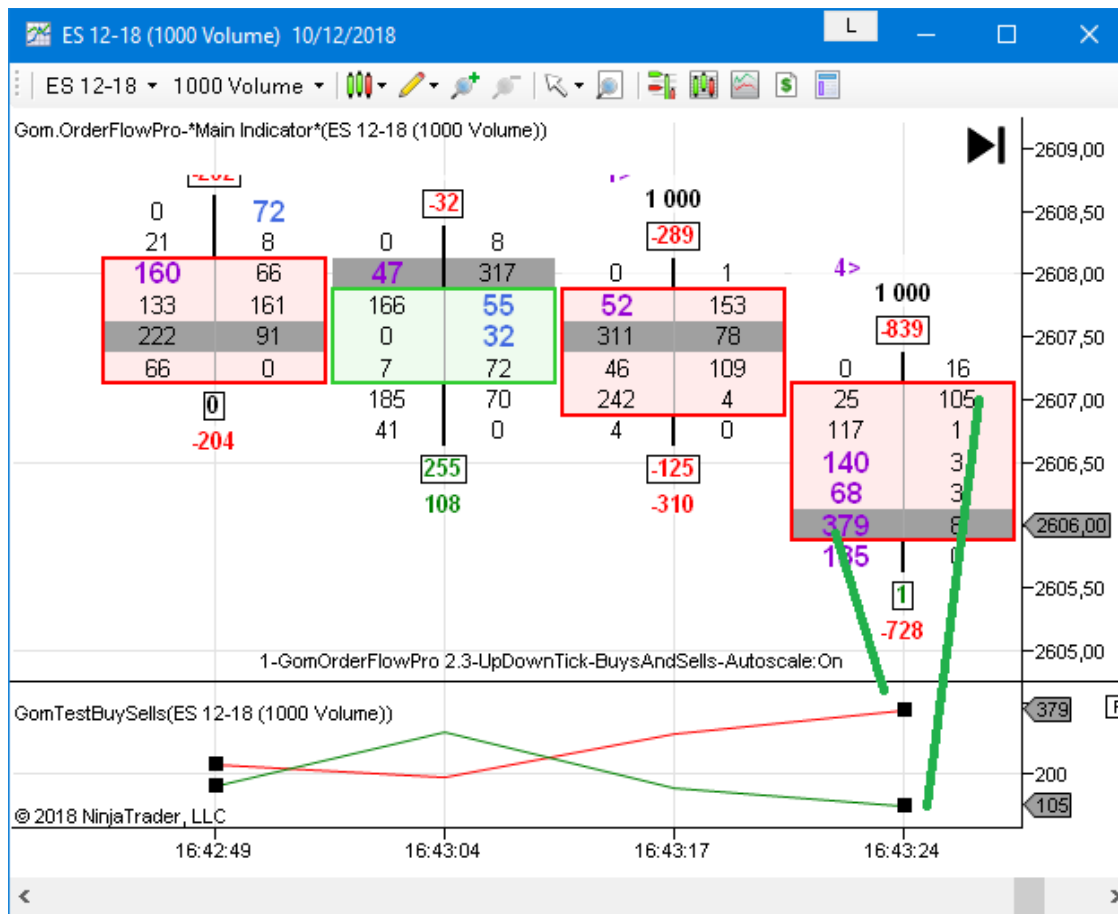
On each *BarData[i]*, you access each lever of price from *BarData[i].MinTick* to *BarData[i].MaxTick*

Buys and Sells on each price levels can be accessed using *BarData[i].Buys[j]* and *BarData[i].Sells[j]*, with *BarData[i].MinTick <= j <= BarData[i].MaxData*

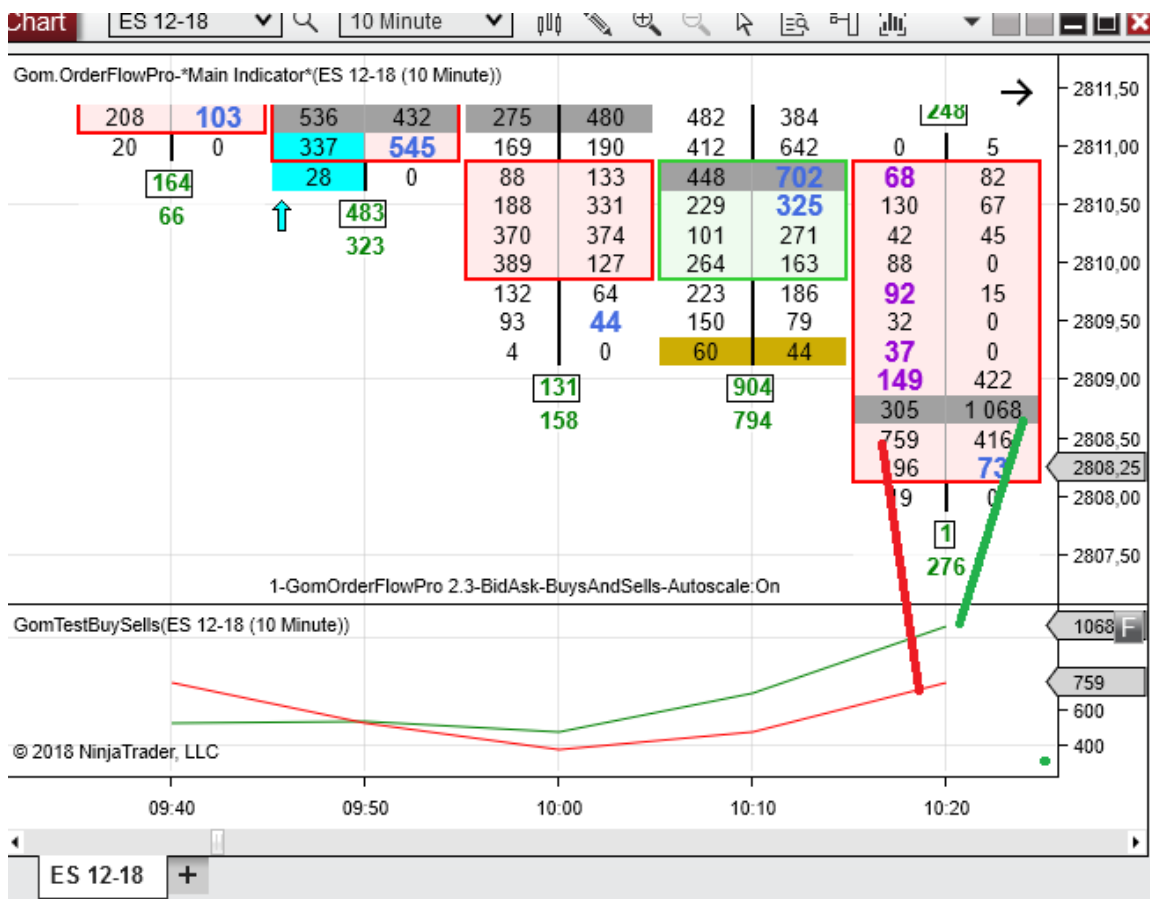
2 demo indicators named **GomTestBuySell.cs** are available in <https://www.gomicators.com/download/support-files/>.

These indicators compute the maximum buy and maximum sell volume on each bar.

NT7 :



NT8





## GOMORDERFLOWPRO-CUSTOMUPDOWNSIGNALS (NINJATRADER ONLY)

This indicator allows creating custom up/down signals using the OrderFloProValues exposed data. Standard NinjaTrader likes Open, High, Low, Close, Volume can be used as well. No coding is required.

This indicator uses the flee expression evaluator to create custom classes on startup and provide native performance.

See <https://archive.codeplex.com/?p=flee> and <https://github.com/mparlak/Flee>

### ORDERFLOW SETTINGS

Same as Orderflowpro, see the relevant doc.

### GRAPHICAL SETTINGS

5. Graphical Settings	
01.Symbol Type	<b>Arrow</b>
02.Price Offset in Ticks	<b>3</b>
03.Up Color	<b>Green</b>
04.Down Color	<b>Red</b>

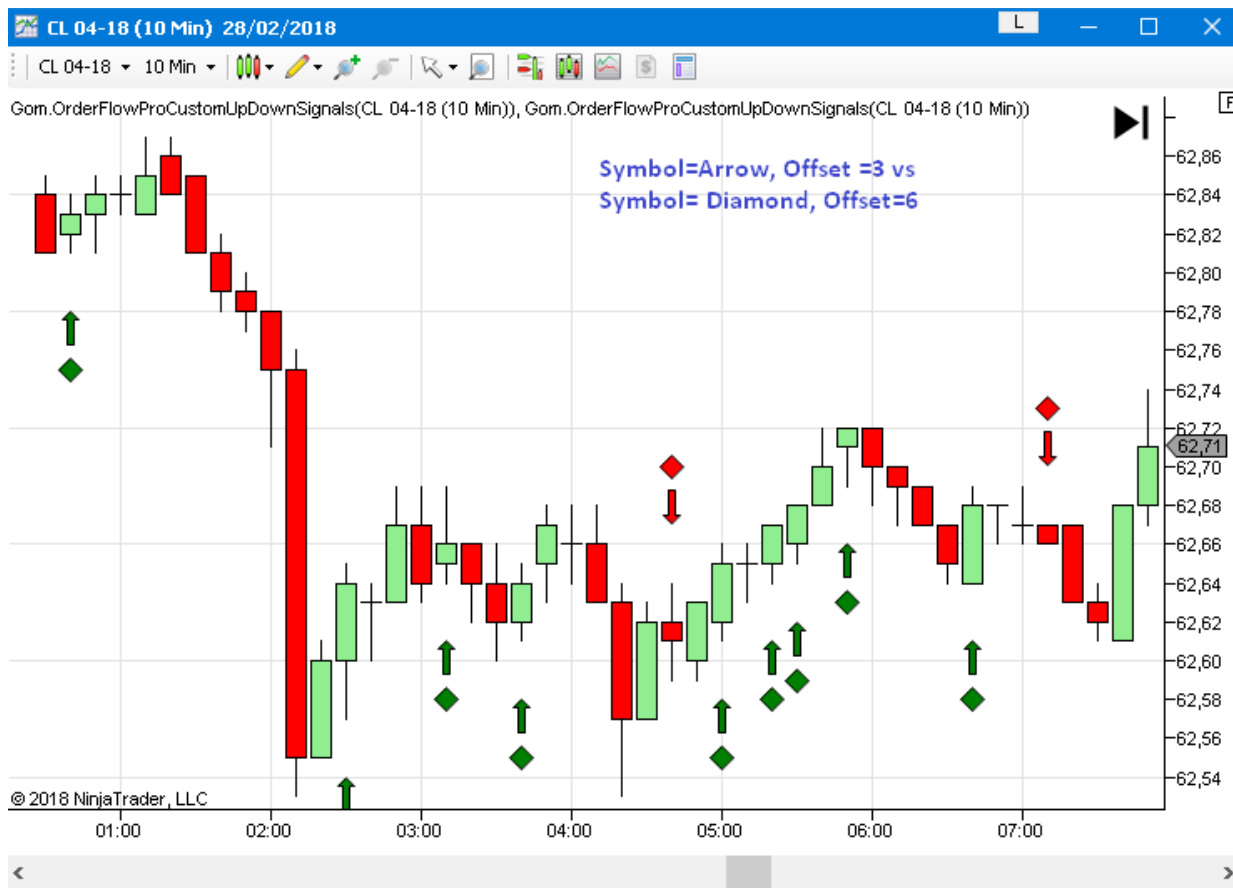
**Graphical Settings**  
Symbol Type: Arrow  
Price Offset in Ticks: 3  
Up Color: Green  
Down Color: Red

**Symbol Type** : Arrow, Dot ,Diamond,Square,Triangle

Defines the symbol used to show the signals

**Price Offset in Ticks :**

Defines the offset from bar top/bottom to plot the symbol



## SIGNAL SETTINGS

This is where are defined the formulas that define the signals.

Any exposed plot from GomOrderFlowProValues can be used , in addition to High,Low,Open, Close

For instance, the default settings show a delta divergence, formulas are :

**Up Formula** :  $\text{Close}[0] > \text{Open}[0]$  and  $\text{Delta}[0] < 0$

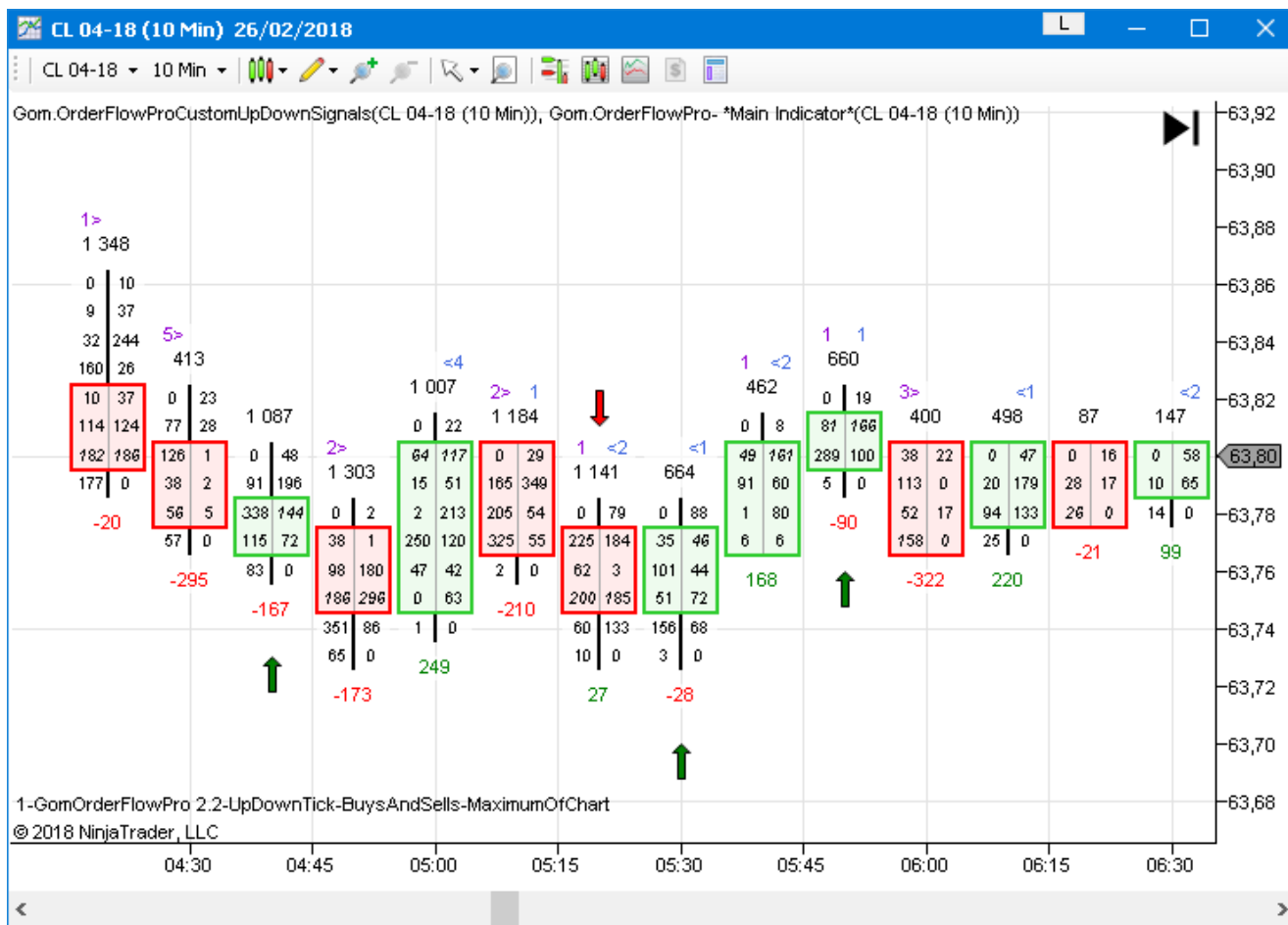
**Down Formula** :  $\text{Close}[0] < \text{Open}[0]$  and  $\text{Delta}[0] > 0$

**New in 2.3** : NinjaTrader indicators can be used as well, so  $\text{SMA}(\text{Close}, 12)[0]$  is a valid formula

**New in 2.3** You can access the indicator's own previous values using  $\text{CustomUpSignal}[i]$  and  $\text{CustomDownSignal}[i]$

You can check syntax examples on the provided free web sites.

Math expressions can also be used, like  $\text{Abs}(\text{Delta}[0])$



## GOMORDERFLOWPRO-CUSTOMUPDOWNSIGNALS (NINJATRADER ONLY)

Like the CustomUpDown indicator, this indicator allows to create custom indicator values with no coding, using values from the GomOrderFlowValues

### ORDERFLOW SETTINGS

Same as Orderflowpro, see the relevant doc.

### SIGNAL SETTINGS

This is where are defined the formulas that define the custom value

Any exposed plot from GomOrderFlowProValues can be used , in addition to High,Low,Open, Close

For instance, the default settings show a delta % of the bar

**Formula** :  $\Delta[0]/\text{Volume}[0]*100$

**New in 2.3** : NinjaTrader indicators can be used as well, so  $\text{SMA}(\text{Close},12)[0]$  is a valid formula

**New in 2.3** You can access the indicator's own previous values using  $\text{CustomValue}[i]$

You can check syntax examples on the provided free web sites.

