

Alerter_Tasks

STOCHASTIC:

MTF_ STOCH SET:

1.

for BOTH STOCH SETS: we do NOT calculated nor consider the Signal line at all

2.

for EACH STOCH SETS: the Stoch MAMethod and Stoch PriceField can be adjusted (per set and not for each individual stochastic)

3.

For each individual Stoch within a Set one can make this inputs:

```
incl_Set1_Stoch1__M1 = false;
```

(note you can not anymore adjust the TimeFrame: it is all fix : also no input mistake.

You can just disable one stoch timeframe or enable it to be considered in the overall Set tasks)

```
Set1_Stoch1_KPeriod = 14;
```

```
Set1_Stoch1_Slowing = 3;
```

```
Set1_Stoch1_Long_Minimum =0;
```

```
Set1_Stoch1_Long_Maximum =100;
```

```
Set1_Stoch1_Short_Minimum =0;
```

```
Set1_Stoch1_Short_Maximum =100;
```

4.

WITHIN a STOCH SET: there are stochastic for each of the **9 possible TF M1-MN1**

5.

Additionally to the two sets there are 3 individual extra Stochastic which can be enabled:

These can have also D% Signal lines which must be extra enabled.

```
incl_Extra_Stoch1 = false;
```

```
Extra_Stoch1_TimeFrame=0;
```

```
Extra_Stoch1_KPeriod = 100;
```

```
use_Extra_Stoch1_DLine = false;
```

```
Extra_Stoch1_DPeriod = 3;
```

```
Extra_Stoch1_Slowing = 3;
```

```
Extra_Stoch1s_MAMethod = 0;
```

```
Extra_Stoch1_PriceField = 0;
```

```
Extra_Stoch1_Long_Minimum =0;
```

```
Extra_Stoch1_Long_Maximum =100;
```

```
Extra_Stoch1_Short_Minimum =0;
```

```
Extra_Stoch1_Short_Maximum =100;
```

Alerter_Tasks

SUGGESTED ACTION (TASKS/CHECKS) FOR STOCHASTIC:

FIRST: Important point of understanding:

I do not follow strictly what one would do looking at multiple charts with different Time Periods.

e.g. to explain this:

What does it mean a line (Stoch/MA) points in a direction: e.g. UP:

simple said: if the current stoch is higher than the previous stoch.

The issue I want to point to is: **What do I (we) understand by PREVIOUS in a MTF setup.**

If one trades of the M1 and looks at multiple charts:

On the main M1 Chart: previous would mean 1 minute ago.

Then he looks at the other charts.

On a Daily chart: previous would mean the previous day.

On a M15 chart: previous would mean the bar before: 15 minutes before

On a M5 chart previous would mean 5minutes ago.

I on the other hand follow a different line:

I follow the point that one has applied different MTF indicators to ONE chart.

And now has all the info MTF info on this single chart. A reason why many prefer MTF indies than 5 charts.

A trader looking at that single chart will base his decisions usually a bit differently.

e.g. let's say a MTF indi trader has applied to a M1 chart:

MTF STOCH TF: D1

MTF STOCH TF: M15

MTF STOCH TF: M5

The trader looking at such a chart bases his decision surely not by scrolling backwards a whole day (1440 M1 bars: and then makes his decision on the difference to the current live M1 candle.)

Checking not even counting 15 bars back for M15.

For such a setup: previous would mean for most: 1 Chart bar ago (in this case M1). And that applies to all MTF indies D1 , M15, M5: Previous means on a M1 chart 1 Minute bar ago, no matter what MTF period is applied.

On the other hand if one changes the chart period to M5: previous would mean 1 M5 bar ago. And that applies to all MTF indies D1 , M15, M5

Alerter_Tasks

Mostly a trader looks at the chart with a kind of Vertical Line mind: just go through any forum where they post charts with MTF- indis applied. Mostly there is a Vline and than some explanation why this was all supporting such and such case. It's quite a different approach than a strictly Multiple charts approach.

Also some other issues: like Past data will often be readjusted (which by Multiple charts does never happen for closed candles.) but in MTF indi charts it happens even for the previous Closed chart bar.

But the argument goes: That's why one uses MTF- indicators.

What is right or wrong: None – it is just a bit of a different approach.

Does it ALWAYS produce the same results: I don't know.

NOTE: the given *e.g. usages* are not the only one this are pure examples.

1. MTF-SETS=====

MTF-SETS are always treated as a unity.

One can NOT perform a task on an individual stochastic (of course if you disable all other 8 TF than the SET consist of only 1 stochastic

// SET1 MTStochs ACTION/TASKS/CHECKS

Set1_STO_SameDirection:

(e.g. usage: simple one wants to get all TF to have the same direction not much to say)

TASK is: check that all included TF within Set1 MUST point into the same direction (LONG (UP), SHORT (DOWN)):

RESULT: Only ONE can be true: SHORT or LONG (or none which we are not interested in)

Set1_STO_SameDirection:

(e.g. usage: simple one wants to get all TF to have the same direction not much to say)

TASK is: check that all included TF within Set1 MUST point into the same direction (LONG (UP), SHORT (DOWN)):

RESULT: Only ONE can be true: SHORT or LONG (or none which we are not interested in)

Set1_STO_WithinMinMaxLevels:

(e.g.: before one considers a possible setup: one wants to make sure that none of the stoch has move for instance already to the opposite extreme: long overbought are – but there is still enough room to ride the stochs up)

TASK is: check that all included TF within Set1 are within each defined Levels.

For LONG to be true between (values inclusive): Long_Minimum and Long_Maximum

For SHORT to be true between (values inclusive): Short_Minimum and Short_Maximum

RESULT: BOTH can be true: SHORT AND/OR LONG

Alerter_Tasks

Set1_STO_BeyondMinMaxLevels:

(e.g.: Spud had requirement that STO 100,3,3 must be in overbought OR oversold : here we follow a similar thought: all included TF must be in one of the extremes)

TASK is: check that all included TF within Set1 are outside/beyond each defined Levels.

For LONG to be true: Stoch below Long_Minimum **OR** Stoch above Long_Maximum

For SHORT to be true: Stoch below Short_Minimum **OR** Stoch above Short_Maximum

RESULT: BOTH can be true: SHORT AND/OR LONG (It's possible that also only one is true if different settings for long and short are used.)

Set1_STO_BLongMaxShortMinLevels: (Set1_STO_BeyondLongMaxShortMinLevels)

(e.g.: this difference from the above that we want to make sure for Long to be true it is above the LONG MAX level and not at the opposite side.)

TASK is: check that all included TF within Set1 are beyond each defined Levels.

For LONG to be true: Stoch above Long_Maximum

For SHORT to be true: Stoch below Short_Minimum

RESULT: BOTH can be true: SHORT AND/OR LONG (Most unlikely but still.) e.g.

Stoch = 81

Long_Minimum =20;

Long_Maximum =80; Stoch above true

Short_Minimum =85; Stoch below true

Short_Maximum =95;

Set1_STO_BLongMinShortMaxLevels: (Set1_STO_BeyondLongMinShortMaxLevels)

(e.g.: this difference from the above that we want to make sure for Long to be true it is below the LONG MIN level – like Spud's 100,3,3 Elastic.)

TASK is: check that all included TF within Set1 are beyond each defined Levels.

For LONG to be true: Stoch below Long_Minimum

For SHORT to be true: Stoch above Short_Maximum

RESULT: BOTH can be true: SHORT AND/OR LONG (Most unlikely but still.)

Set1_STO_AllInOrder: STO_AllInOrder is: Lower TF (leading Higher TF)

(e.g.: Spuds Elastic setup (I don't remember now if he really required it – but there are a couple of usages)

For LONG to be true: Lower TF is higher up than Higher TF: e.g. M5= 83, M15=78, M30=45

For SHORT to be true: Lower TF is lower up than Higher TF: e.g. M5= 13, M15=38, M30=45

RESULT: Only ONE can be true: SHORT or LONG

Alerter_Tasks

1. REAL PRICE=====

// REAL PRICE ACTION/TASKS/CHECKS

RealPrice_Direction:

(e.g. usage: simple get the direction)

TASK is: check the the Real Price Direction

For LONG to be true: RealPrice points up

For SHORT to be true: RealPrice points down

RESULT: Only ONE can be true: SHORT or LONG

RealPrice_TwoBarsDirection:

(e.g. usage: simple get the direction)

TASK is: check the the Real Price Direction over two bars

For LONG to be true: RealPrice points up (real price now is greater than it was one before previous)

For SHORT to be true: RealPrice points down (real price now is less than it was one before previous)

RESULT: Only ONE can be true: SHORT or LONG

RealPrice_ThreeBarsDirection:

(e.g. usage: simple get the direction)

TASK is: check the the Real Price Direction over three bars

For LONG to be true: RealPrice points up

For SHORT to be true: RealPrice points down

RESULT: Only ONE can be true: SHORT or LONG

RealPrice_FiveBarsDirection:

(e.g. usage: simple get the direction)

TASK is: check the the Real Price Direction over 5 bars

For LONG to be true: RealPrice points up

For SHORT to be true: RealPrice points down

RESULT: Only ONE can be true: SHORT or LONG

RealPrice_BeyondExtra_MA1:

(e.g. usage: If MA gives direction we might want to make sure this direction is not yet broken.)

TASK is: check the the Real Price is beyond the EXTRA_MA1

For LONG to be true: RealPrice above EXTRA_MA1

For SHORT to be true: RealPrice below EXTRA_MA1

RESULT: Only ONE can be true: SHORT or LONG

Alerter_Tasks

RealPrice_BeyondExtra_MA2:

(e.g. usage: If MA gives direction we might want to make sure this direction is not yet broken.)

TASK is: check the the Real Price is beyond the EXTRA_MA2

For LONG to be true: RealPrice above EXTRA_MA2

For SHORT to be true: RealPrice below EXTRA_MA2

RESULT: Only ONE can be true: SHORT or LONG

RealPrice_BeyondExtra_MA3:

(e.g. usage: If MA gives direction we might want to make sure this direction is not yet broken.)

TASK is: check the the Real Price is beyond the EXTRA_MA3

For LONG to be true: RealPrice above EXTRA_MA3

For SHORT to be true: RealPrice below EXTRA_MA3

RESULT: Only ONE can be true: SHORT or LONG