

ZING

USER MANUAL

ZING



ZING

DETAILED EXPLANATION OF INPUT TAB PARAMETERS

The following pages explain all parameters in the input tab. Don't be overwhelmed by the number of input parameters. The more you use the EA, the less intimidated they are.

Nomenclature:

CP=current price

OP=open price at entry

TP=take profit price

SL=Stop Loss price

SLatEntry=Stop Loss at the trade entry (opening)

ATR= Average true range (latest value)

RPP= Risk per pip (computed based on risk amount, SL and lot size)

MAxx = Moving Average of xx periods, e.g. MA20

ASK=ask price

BID=bid price

EA=Expert Advisor

****All explanations are for a BUY trade. SELL trades follow the same logic with proper changes.**

Add any comment for your reference

This is a string variable that the user defines. It is also displayed in the chart top-left corner.

TIME PARAMETERS

Time frame (bar period)

All time frames from MT5 are available. The recommended time frames for this EA are M15, M20, M30, H1. This variable can be included in the optimization process.

ServerStartTime_hr, Start trading at this server time (hour) [0,23]

This is the server-time hours at which the EA is allowed to start trading every day. It is an integer value.

TradeWindow_hr, Trade for this many hours each day [1,24]

This is the time window in hours (integer value) that the EA is allow to trade.

User could optimize which hours to trade if so desired.

tradeOnMonday, Allow to open trades on Monday (server time)

tradeOnTuesday, Allow to open trades on Tuesday (server time)

tradeOnWednesday, Allow to open trades on Wednesday (server time)

tradeOnThursday, Allow to open trades on Thursday (server time)

tradeOnFriday, Allow to open trades on Friday (server time)

When any of these five variables is TRUE, the EA is allowed to open trades on the corresponding day of the week. When FALSE, no trades are open on that day, however, a trade opened on a TRUE day, can still stay open in a FALSE day if the trade has not hit the stop loss or the take profit. User could optimize which days are better to trade if so desired.

ZING STRATEGY PARAMETERS

i0_zing [1,1,10]

b1_zing [false,true] stochastic 1

b3_zing [false,true] stochastic 2

b5_zing [false,true] stochastic 3

b7_zing [false,true] stochastic 4

*These parameters controls the **ZING** entry strategy. i0_zing indicates which time frame is used for stochastics #3 and #4. Stochastics #1 and #2 use the base time frame. The variable bx_zing indicates when to activate each stochastic signal. User may play with different ranges and steps in the optimization, but the range values shown above will work in most Forex symbols.*

NOTE: **ZING** allows trading of non-Forex symbols, such as ETFs, indices, equities, but be aware that **ZING** was not tested for non-Forex symbols. Use **ZING** with care in such cases.

TP <OPEN> PARAMETERS

Note: In general, for TP or SL methods, each has a specific set of variables that are needed. For instance, in the TP method to open trades called “TPopMult”, there are two variables needed: TP_opMult and TP_opNbars. Any other variable is ignored when using this method. The list of variables used in each method is listed inside parentheses next to the method’s name.

In all TP methods, if TP_opNbars > 0, it compares the computed TP with the highest high of previous TP_opNbars bars, and uses the lowest between the highest high and the calculated TP as the final open TP. If TP_opNbars ≤ 0, no comparison is done.

At Opening TP method

These are the four methods to set the open TP:

TPopMult: TP_opMult,TP_opNbars

This is the most popular method used by traders. It multiplies (OP-SL) by TP_opMult to get (TP-OP). If TP_opMult is zero, there is no TP for this trade and the position will be only closed by a trailing SL. It then compares the TP computed with the highest price in the previous TP_opNbars bars, and selects the lower of the two values. If TP_opNbars ≤ 0, no comparison is done.

TPopPct: TP_opPct,TP_opNbars

TP is computed by adding TP_opPct % of the current price to OP. If TP_opPct is zero, there is no TP for this trade and the position will be only closed by a trailing SL. It then compares the TP computed with the highest price in the previous TP_opNbars bars, and selects the lower of the two values. If TP_opNbars ≤ 0, no comparison is done.

TPopAtr: TP_opAtr,TP_opNbars

TP is computed by adding TP_opAtr*ATR to OP. If TP_opAtr is zero, there is no TP for this trade and the position will be closed only by a trailing SL. It then compares the TP computed with the highest price in the previous TP_opNbars bars, and selects the lower of the two values. If TP_opNbars ≤ 0, no comparison is done.

TPopFractal: TP_opAtr,TP_opNbars

First, the most recent fractal High price is found; call it *tp1*. Second, set TP to the maximum between *tp1* and OP+TP_opAtr*ATR. If TP_opAtr is zero, there is no TP and the position will be closed only by a trailing SL. If OP is above all previous fractal High prices, TP is set to TP_opAtr*ATR above OP. Lastly, it then compares the TP computed with the highest price in the previous TP_opNbars bars, and selects the lower of the two values; if TP_opNbars ≤ 0, no comparison is done.

TPopGap: TP_opConvex01, TP_opNbars

After finding the latest gap in price, it sets the TP to a convex combination between the low of the pre-gap candle and high of the post-gap candle. If no gap is found, the TP is set to the high of the previous candle plus 0.5*ATR. TP_opConvex01 is the convex combination weight: near 1, it sets the TP closer to the OP. It then compares the TP computed with the highest price in the

previous TP_opNbars bars, and selects the lower of the two values; if TP_opNbars<=0, no comparison is done.

Input Parameters for Open-TP methods

After selecting the Open-TP method, set the two corresponding parameters for the chosen method. TP_opNbars is used in all methods.

TP_opMult if 0 => no TP

It is a unitless quantity. If TP_opMult is zero, there is no TP for this trade and the position will be only closed by a trailing SL. A range of [0,10] should be enough.

TP_opAtr [0,2,20](if 0 => no TP)

It is measured in units of ATR. If it is zero, there is no TP for the trade and the position will be only closed by a trailing SL. The range shown [0,20] is a recommendation, but you can surely go outside it.

TP_opConvex01 [0,0.2,1](higher=>near to OP)

It is a unitless quantity. Close to 1, it will set TP nearer to OP. The range shown [0,1] is highly recommended. Outside this range, there will be unpredictable behavior.

TP_opPct [0.5,0.5,2]

It is measured in units of percent of the open price. The range shown [0.5,2] is a recommendation, but you can surely go outside it.

TP_opNbars # bars support/resistance |0|

It is measured in units of number of bars in the past. A range of [0, 50] is reasonable, but you can surely go outside it.

SL <OPEN> PARAMETERS

At the Opening SL Method

There are several method to calculate the SL at the opening of the trade. They are:

FixPips: SL_opFixPips

This is the most popular method used by traders. It sets SL at SL_opFixPips below from the OP.

ATR: SL_opATR

This is another popular method. It sets SL at SL_opATR*ATR below from OP.

PrevBar: SL_opPrevBar

It sets SL below the lowest bar low in the previous SL_opPrevBar bars.

ATR+FP+PB: SL_opATR, opFixPips, opPrevBar

It compares ATR , FixPips and PrevBar SL calculations and selects the closest to OP if SL_opAggr is true, otherwise selects the farthest away.

FP+PB: SL_opFixPips, SL_opPrevBar

It compares FixPips and PrevBar SL calculations and selects the closest to OP if SL_opAggr is true, otherwise selects the farthest away.

FixPip+ATR: SL_opFixPips, SL_opATR

It compares FixPip and ATR SL calculations and selects the closest to OP if SL_opAggr is true, otherwise selects the farthest away.

PB+ATR: SL_opPrevBar, SL_opATR

It compares PrevBar and ATR SL calculations and selects the closest to OP if SL_opAggr is true, otherwise selects the farthest away.

MA20: SL_opConvex01

It sets SL between the MA20 and the low of the previous bar. SL_opConvex01 determines how close to MA20 the SL is set.

AtrMA20: SL_opATR

It sets SL at the lowest between the low of the previous bar and MA20 – SL_opATR*ATR.

MA20200: SL_opConvex01

It sets SL between the MA20 and MA200. SL_opConvex01 determines how close to MA20 is set.

Pct: SL_opPricePct

It sets SL at SL_opPricePct % of ASK below from OP.

PctPB: SL_opPricePct, SL_opPrevBar

It compares SL computed with SL_opPricePct and SL_opPrevBar and selects the closest to OP if SL_opAggr is true, otherwise selects the farthest away.

OneBar: no params

It sets SL below the previous bar low.

Swing: SL_opFixPips

It sets SL SL_opFixPips pips below the previous fractal bar low.

Gap: SL_opConvex01

It finds the nearest non-filled gap and sets SL inside that gap. SL_opConvex01 near 1 will set the SL farther from the OP; SL_opConvex01 near 0, will set SL nearer to OP.

Input parameters for open-SL methods:

Once you select a method for the open-SL, you need to also set, as most, two parameters that control the method you selected. See the method above to know which parameters to set.

SL_opFixPips [10,10,120]

It is measured in pips. The range shown [10,120] is a recommendation, but you can surely go outside it.

SL_opATR [1,1,10]

It is measured in units of ATR. The range shown [1,10] is a recommendation, but you can surely go outside it.

SL_opPrevBar [3,3,11]

It is measured in units of bars count in the past. The range shown [3,11] is a recommendation, but you can surely go outside it.

SL_opConvex01 [0.1,0.2,1]

It has no units. The range shown [0.1,1] is a highly recommended. Values outside this range may produce unexpected results.

SL_opPricePct [0.1,0.1,0.5]

It is measured in units of percent of price. The range shown [0.1,0.5] is a recommendation, but you can surely go outside it.

SL_opAggr (True/False)

It is a boolean parameter. If true, more aggressive SL are selected, meaning, SL is closer to the open price.

SL <TRAILING> PARAMETERS

SL Trailing Method (at each new bar)

There are several methods to calculate the trailing SL once the position is open. These methods are executed at the beginning of every new bar, unless otherwise indicated in the description below. Once you select a SL-trailing method, you need to set up to two additional parameters which are listed next to the method's label. This means that a maximum of two parameters will have to be set for a given method. All other parameters are then ignored.

Static: SL won't move (no params needed)

SL won't move until the position is closed by TP. Be aware that if TP is 0, this Static method will only give losses since there is no TP.

FixPips: SL_trFixPips

SL is set such that it is never more than SL_trFixPips away at the open of each new bar.

ATR: SL_trATR

SL is set at $SL_trATR * ATR$ below the last bar low.

PrevBar: SL_trPrevBar

SL is set below the lowest bar low of the previous SL_trPrevBar bars.

Acc: SL_trTrigFac, SL_trAcc (recommended method)

SL is moved up when profit is larger than $SL_trTrigFac * (TP - OP)$. SL is then set using a polynomial with a degree controlled by SL_trAcc. Higher values of SL_trAcc result in more aggressive moves of SL toward the current price.

MA20: SL_trConvex01, SL_trTrigFac

If profit is greater than $SL_trTrigFac * (TP - OP)$, SL is set in between MA20 and last bar low, never higher than the last bar low. SL_trConvex01 controls how close to MA20 SL is set: 0 will be the closest to MA20.

ATR+MA20: SL_trATR, SL_trTrigFac

If profit is greater than $SL_trTrigFac * (TP - OP)$, it sets SL as the minimum between the last bar low and $MA20 - SL_trATR * ATR$.

B.E.+MA20: SL_trATR

If MA20 is greater than OP and the position is in profit, SL is set below the MA20 at a $SL_trATR * ATR$ distance.

B.E.+MA50: SL_trATR

It works the same as the previous method but uses MA50 instead.

StepPip: SL_trFixPips

When the position is in profit by $2 * SL_trFixPips$, SL is moved up SL_trFixPips, then, only when the close price moves up by SL_trFixPips, SL is moved SL_trFixPips. This is the step-wise version of the previous method.

DistPip: SL_trTrigFac, SL_trFixPips

When the position is in profit by $SL_trTrgFac \cdot (OP - SL)$, SL is moved up $SL_trFixPips$, then, only when the close price moves up by $SL_trFixPips$, SL is moved $SL_trFixPips$.

StepAtr: SL_trATR

When the position is in profit by $2 \cdot SL_trATR \cdot Atr[1]$, SL is moved up $SL_trATR \cdot Atr[1]$, then, only when the close price moves up by $SL_trATR \cdot Atr[1]$, SL is moved $SL_trATR \cdot Atr[1]$. Notice that the step $SL_trATR \cdot Atr[1]$ is not constant, but depends on the last candle ATR.

DistAtr: SL_trTrigFac, SL_trATR

When the position is in profit by $SL_trTrgFac \cdot Atr[1]$, SL is moved up $SL_trATR \cdot Atr[1]$, then, only when the close price moves up by $SL_trATR \cdot Atr[1]$, SL is moved $SL_trATR \cdot Atr[1]$. Notice that the step $SL_trATR \cdot Atr[1]$ is not constant, but depends on the last candle ATR.

StepMoney: SL_trFixMoney

When the position is in profit every $SL_trFixMoney$ in the account currency, SL is moved up $SL_trFixMoney$ in a step-wise manner.

DelCl: SL_trConvex01

SL is up $SL_trConvex01 \cdot (\text{difference in close price of the last two bars})$

MA20200: SL_trTrigFac, SL_trConvex01

When the position is in profit by an amount larger than $SL_trTrigFac \cdot (TP - OP)$, SL is set in between MA20 and MA200. $SL_trConvex01$ controls how close to MA20 the SL is set.

TPOP: SL_trConvex01

When the position is in profit by an amount larger than $SL_trConvex01 \cdot (TP - OP)$, SL is set as a convex combination of the previous SL and the last bar close. Lower values of $SL_trConvex01$ are farther away from the current price (more conservative SL).

FP+PB: SL_trFixPips, SL_trPrevBar

It compares SL computed with FixPips and PrevBar methods. It selects the closest SL to the current price.

BarCount: SL_trBarCountOp, SL_trFixPips

When the position is in profit, and the number of bars after opening the position reaches $SL_trBarCountOp$, the SL is set to $CP - SL_trFixPips$.

PullBack: SL_trConvex01 || SL_ClosePos

When the position is in profit, and the latest max that closes between the open bar and the current bar is overtaken by the current close, then two possible scenarios happen:

If $SL_ClosePos$ is FALSE, the SL is set between MA20 and the low of the last bar. $SL_trConvex01$ controls how close SL is to the MA20.

If $SL_ClosePos$ is TRUE, the position is closed immediately in profit.

O.V.20: *SL_trConvex01*

When the position is in profit and the previous bar has a body greater than $SL_trConvex01 * ATR$, SL is set to near the previous bar open, otherwise, SL is set to MA20. In other words, it mostly trails “O”n the “V”alue of MA20, but if there is a large body bar, the SL is moved near the open of that body, and stays there until the MA20 catches up.

O.V.40: *SL_trConvex01*

When the position is in profit and the previous bar has a body greater than $SL_trConvex01 * ATR$, SL is set to near the previous bar open, otherwise, SL is set to MA40. In other words, it mostly trails “O”n the “V”alue of MA40, but if there is a large body bar, the SL is moved near the open of that body, and stays there until the MA40 catches up.

Swing: *SL_trFixPips*

When the position is in profit, if there is a low fractal at the third previous bar, the SL is set to below the low of the third bar by a distance of $SL_trFixPips$ pips.

StepOSL: *SL_trTrigFac, SL_trConvex01*

When the position is in profit, and the $CP - OP$ is greater than $T * SL_trTrigFac * (OP - SL_{atEntry})$ the SL is moved up to be away from the CP an amount equal to $(1 - T * SL_trConvex01) * (OP - SL_{atEntry})$. The quantity “T” is the number of rounded times $CP - OP \geq SL_trTrigFac * (OP - SL_{atEntry})$. This means that as the position keeps growing in profit, the trailing SL gets tighter. The larger the $SL_trConvex01$, the closer (more aggressive) the SL is to the CP.

HL20: *SL_trConvex01, SL_trPrevBar*

When the position is in profit, it uses the candle low price to compute the trailing SL from a convex combination between the low price $SL_trPrevBar$ candles earlier, and the MA20. $SL_trConvex01$ is in the interval $[0,1]$. $SL_trPrevBar$ controls how close to MA20 the SL is.

HL40: *SL_trConvex01, SL_trPrevBar*

When the position is in profit, it uses the candle low price to compute the trailing SL from a convex combination between the low price $SL_trPrevBar$ candles earlier, and the MA40. $SL_trConvex01$ is in the interval $[0,1]$. $SL_trPrevBar$ controls how close to MA40 the SL is.

HA: *SL_trPrevBar*

When the position is in profit, it uses Heikin Ashi candles to compute the trailing SL from the low value $SL_trPrevBar$ candles earlier, current price, and previous SL.

CutLoss: *SL_trBarCountOp, SL_trConvex01*

When the position is losing money at the close of the latest candle, $SL_trConvex01$ (which is in this range: $[0,1]$) is mapped linearly to $p = [0.01, 0.1] * (OP - SL)$. If $Profit \leq -p$ and the number of candles since the open is greater or equal to $SL_trBarCountOp$, then SL is moved to $Close[latestCandle] - p/10$. This SL trailing strategy can be used to cut losses quick since the SL is set very tight to the close price of the previous candle. The smaller the two parameters are, the quicker the losses can be cut. Choose $SL_trConvex01$ based on your risk tolerance for losses relative to the total risk in the trade (i.e., $OP - SL$). When using this strategy, the win rate

will go down and the number of consecutive losing trades will go up, but it is still possible to have a profitable strategy a profit factor greater than 1.

Gap: *SL_trConvex01*

If the profit>0, and there is a gap up, SL is set to a convex combination of the previous SL and the last candle low. The lower *SL_trConvex01*, the more conservative the SL is. If the profit is less or equal to zero, SL is not moved.

Input Parameters for SL Trailing Methods

Only a maximum of two parameters are in play based on the SL-trailing method chosen above. When optimizing, use the values inside brackets [a,b,c]. The value inside vertical bars (like this: |x|) disables the corresponding parameter or method.

***SL_trConvex01* [0,0.1,1]**

Parameter used in several SL trailing methods. It should be in the interval [0,1]

***SL_trTrigFac* [0.2,0.2,1]**

Parameter used in several SL trailing methods.

***SL_trFixPips* [10,10,120]**

Parameter used in several SL trailing methods. Its unit is PIP.

***SL_trATR* [0.5,0.5,]**

Parameter used in several SL trailing methods. Its unit is a multiple of ATR.

***SL_trPrevBar* [3,3,10]**

Parameter used in several SL trailing methods. Its unit is number of bars in the past.

***SL_trAcc* [0.25,0.25,1.25]**

Parameter used in the Acc method only. It works along with *SL_trTrigFac* to trigger the SL move to Break Even. A value *SL_trTrigFac*=0.2 has a faster move than 0.8. *SL_trTrigFac*>=1 will disable this trailing method and it will behave as a static SL. If TP is not defined (*TP_opMult*=0), the trailing SL is also static. Therefore, if the Acc trailing method is used, we recommend to set *TP_opMult*>0. When the SL is static, the trailing is relying on other parameters in the next section of this document. *SL_trAcc* controls how tight the SL is at every new bar after the move to break even. Higher *SL_trAcc* will move SL tighter to the current price in a polynomial manner.

***SL_trFixMoney* [Currency amount]**

Parameter used in several SL trailing methods. Its unit is in the account currency (e.g., USD, EUR, JPY, GBP, etc.).

***SL_trBarCountOp* [3,1,5]**

Parameter used in several SL trailing methods. Its unit is number of bars. It is compared to the number of bars after the opening of the trade.

SL PARAMS TO AVOID BIG LOSSES

These techniques are performed before calling the SL trailing method chosen above. If these techniques are activated, they are designed to avoid big losses before being stopped out by the SL at the open. If you are too aggressive with these parameters you risk to be stopped-out very often.

SL_negGap: move SL when a negative Gap happens

In a BUY trade, Gap is defined as a three bar event where there is a distance (price gap) between the first bar HIGH and the third bar LOW.

In a SELL trade, Gap is defined as a three bar event where there is a distance (price gap) between the first bar LOW and the third bar HIGH.

If SL_negGap is true, move SL aggressively only if there was a gap in the opposite(negative) direction of the trade.

If SL_negGap is false, do nothing on negative gaps events.

SL_BADbigBar ATR factor |0|[2:1:5]

If there is one bar (or two consecutive bars) with big body size ($\text{body} > \text{SL_BADbigBar} * \text{ATR}$) against the position, the SL is set very tight, and continues to be tight from then on.

SL_ClosePos (reduce SL if opp entry is better)

This is used in some of the trailing SL methods above. Also, even when it is not used in the selected trailing SL method, if TRUE and the entry conditions are reversed, SL is reduced.

SL_closeX200 (reduce SL if cross MA200 & -trend)

If TRUE, and price crosses MA200 in reversing trend, SL is reduced.

SL_ConsOppBars |1|[1,2,11](reduce SL if conseq bars opp dir)

This is the number of consecutive opposite direction bars (opposite to the direction of the trade) that will trigger a tighter a reduction in SL if the trade is in negative profit. If set to 1, it is not an active method.

SL PARAMS TO PRESERVE GAINS

The following parameters are used to quickly preserve the gains of a trade. If you are too aggressive with these parameters, your wins will tend to be small, not reaching full TP potential.

SL_posGap: move SL when a positive Gap happens

In a BUY trade, Gap is defined as a three bar event where there is a distance (price gap) between the first bar HIGH and the third bar LOW.

In a SELL trade, Gap is defined as a three bar event where there is a distance (price gap) between the first bar LOW and the third bar HIGH.

If SL_posGap is true, move SL aggressively only if there was a gap in the same (positive) direction of the trade.

If SL_posGap is false, do nothing on positive gaps events.

SL_GOODBigBar ATR factor |0|[2:1:5]

If there is one bar (or two consecutive bars) with big body size ($\text{body} > \text{SL_GOODbigBar} * \text{ATR}$) in favor of the position, the SL is set very tight, and continues to be tight from then on.

SL_turbo01: fraction of TP-OP tight SL |0|[0:.2:1]

If the profit is greater than $(1 - \text{SL_turbo01}) * (\text{TP} - \text{OP})$, the SL is moved near the previous bar low, and continues doing so at every new bar. In other words, the higher this parameter, the quicker the SL will start moving. Once it starts moving, it continues to keep a tight SL every new bar. This is a good strategy for situations when the profit keeps improving slowly without any big detectable jump in price. When this parameter is 0 it is inactive. When TP is set to zero, this parameter is also inactive.

SL_fractal: fractal near TP |<0.5|[0.4,0.1,1]

If the profit is greater than zero, and there was a fractal structure near the TP, i.e., $\text{fractal_High} \geq \text{SL_fractal} * (\text{TP} - \text{OP})$, the SL is moved near the previous bar low, and continues doing so at every new bar. When SL_fractal is less than 0.5, or greater than 1, or TP is set to zero, this is inactive.

PARTIAL TAKE-PROFIT PARAMS <@each tick>

SL_Gain2BE: trigger to move to B.E. |0|[0.5,0.5,TP]

SL_Gain2CL: added trigger to close %vol |0|[0.5,0.5,TP]

SL_GainVpct: % vol to close (==0 -> move to break even)

These three parameters work together and work on every tick. If the position has a profit more than $SL_Gain2BE * (OP - SLatEntry)$, it will be moved SL to Break Even. If the position has a profit more than $(SL_Gain2BE + SL_Gain2CL) * (OP - SLatEntry)$, the position will be closed and another position is immediately opened with SL at the open price of the closed position, and with a lot size equal to $(100 - SL_GainVpct) / 100$ of the original position lot size. The TP price remains the same. This two operations are equivalent to “partially taking profit” the original lot size. It works in Netting accounts without violating any rule. If you want to use this feature, it is recommended to set SL_GainVpct at least 50%. If you want to move SL to break even without partially closing the position, use $SL_Gain2BE > 0$, $SL_Gain2CL = 0$.

PROP FIRM INFORMATION

Our EA is “proprietary firm” friendly. If you are using our EA to pass a Proprietary firm challenge, you will like these features. You need to provide three parameters:

Prop.Firm-like rules for daily & account DD

These rules are used when computing the daily and account draw-down limits. Each prop firm has its own rules regarding the daily draw-down. The current version of the EA supports the following “types” of rules:

FTMO-like rules (go to www.ftmo.com)

TheFundedTrader-like rules (go to www.thefundedtraderprogram.com)

NoRules

*****IMPORTANT:** You don’t need to use FTMO or TFT; you are simply stating that your prop firm uses rules “like” FTMO, or “like” TFT.

***** Note:** if you select “NoRules” the counter of violation of max daily losses will increase every tick that the daily loss is violated. This does not affect anything in the strategy, but you will see a large number in the Tester Summary printed in the Journal tab, and if you choose to constraint the number of max daily losses, it is likely this constraint will be violated.

Current GMT off set (eg, Prague=2 when DST, 1 otherwise)

This is the current GMT off set for the prop firm location. For instance, FTMO is in the Central European Time zone, which is GMT+1 when there is no Daylight Saving Time (DST) (~October to ~March), or it is GMT+2 when there is DST. So, depending on when in the year you are running this EA, you would enter 1, or 2 if you are using FTMO. In another example, TheFundedTrader is in the Easter Standard Time zone, which is GMT-5 when there is no DST, or GMT-4 when there is DST. So, you would enter -5 or -4 depending on when you are using this EA.

Roll over hour in PropFirm time (e.g., 17 for TheFundedTrader)

This is the time when the prop firm reset the day to start a new trading day. In the case of FTMO, they do it at CET 24 (that is at midnight), hence you enter 24 here. The prop firm TFT does it at EST 17 (that is at the 5pm), hence you enter 17 here. Your prop firm may have different roll over hour that you will find in the prop firm website, and enter here.

***** IMPORTANT:** make sure you change the parameter *Current GMT off* twice a year (first in the Spring, and second in the Fall)

RISK MANAGEMENT (RM) PARAMETERS

Risk Management (RM) method at the open

There are 8 different ways to set the risk for a trade:

Fixd % Free Margin (RM:1,2,6) (recommended for live trading)

This uses a fixed percent of the free margin to calculate the amount of money risked on each trade. It will use InitialRisk_pct (RM1), MaximumRisk_pct(RM2) and MaxLotSize(RM6) when computing the lot size. This is the recommended method when trading live.

Fixd % of Ini.Deposit (RM:0,1,2,6) (recommended for optimization)

This uses a fixed percent of the initial deposit to calculate the amount of money risked on each trade. It will use InitialRisk_pct, MaximumRisk_pct and MaxLotSize when computing the lot size. This is the recommended method when optimizing parameters.

Fixd % Allowed DailyLoss (RM:1,2,6)

This uses a fixed percent of the allowed daily loss to calculate the amount of money risked on each trade. For instance, if the max daily loss allowed is 5% of the initial deposit (which is set in RM8), the user may want to risk 10% of that amount on each trade, which is equivalent to 0.5% of the initial deposit ($10\% \times 5\% = 0.5\%$). InitialRisk_pct, MaximumRisk_pct and MaxLotSize are used when computing the lot size.

Lin Grow % FreeMarg (RM:1,2,6)

This uses a linear growth of risk (from RM1=InitialRisk_pct to RM2=MaximumRisk_pct) in percentage of the free margin. It will limit the lot size with RM6=MaxLotSize;

Cumltve Grow % FreeMarg (RM:1,2,5,6)

This uses a cumulative growth of risk (from RM1=InitialRisk_pct to RM2=MaximumRisk_pct) in percentage of the free margin using RM5=CumRedFactor as multiplier. The amount at risk can go up or down It will limit the lot size with RM6=MaxLotSize;

Fixd lot per order (RM:3,6,2)

This uses a fixed lot size of RM3=FixedLotSize for each trade. It will consider RM2=MaximumRisk_pct and RM6=MaxLotSize when computing the final lot size for the trade.

Fixd Lot/(10k FreeMarg) (RM:4,2,6)

This uses a fixed lot size of RM4=LotSizePer10k for each 10,000 units of the account currency in free margin. It will consider RM2=MaximumRisk_pct and RM6=MaxLotSize when computing the final lot size for the trade.

Fixd Currency Amount (RM:7,6)

This uses a fixed amount of account currency from the free margin defined by RM7=inpFixedMargin, and RM6=MaxLotSize when computing the final lot size for the trade. For instance, if your account is in USD, you could set it RM7=30 to risk 30 USD at every trade.

RM0: Init.Deposit to calculate max loss allowed

This is the initial deposit in the account. This is used to compute the maximum daily and account loss allowed. During live trading, this quantity will differ from the account balance and equity.

*****IMPORTANT:** When the EA is loaded to the chart, it checks that this Init.Deposit is greater than the minimum balance, or equity, allowed which is determined by parameters RM10 and RM11 below. If this check fails, the EA is removed automatically from the chart. The parameters RM1,..., RM7 used in the risk management methods above are variables that could be optimized, but we recommend to keep them constant. They are:

RM1: Initial % Risk (per trade)

RM2: Max % Risk (per trade)

RM3: Fixed Lot Size (per trade)

RM4: Lot Size Per every 10k of free margin

RM5: Cum.mult.factor of gain % [0.1,1]

RM6: Max Lot Size per trade [>0]

RM7: Fixed Currency amount risked (per trade)

OTHER RISK PARAMETERS (FIXED)

RM8: Prop.Firm MaxDailyLoss in % of Ini.Dep.

This is used to calculate the max daily loss allowed, as a percentage of the initial deposit. This does not change when the balance or the equity changes. Every new day (based on server time) the lowest equity allowed changes as per the prop firm rules. See Note2 below

RM9: Safety factor for DailyLoss (RM8) [0.5,1]

This multiplies RM8 to have a buffer of safety. For instance, if RM8=5%, RM9=0.9, and Initial Deposit=10,000, then max daily loss limit is $0.05 \times 0.9 \times 10,000 = 450$, every new day. See Note2 below

RM10: Prop.Firm MaxAccountLoss in % of Ini.Dep.

This is used to calculate the max account loss allowed, as a percentage of the initial deposit. This does not change when the balance or the equity changes. See Note2 below.

RM11: Safety factor for Acct.Loss (RM10) [0.5,1]

This multiplies RM10 to have a buffer of safety for the account. For instance, if RM10=10%, RM11=0.9, and Initial Deposit=10,000, then max account loss limit is $0.10 \times 0.9 \times 10,000 = 900$. This means that the account equity cannot go below $10000 - 900 = 9100$ any time, any day. See Note2 below

RM12: Use RM8, RM10 when computing lot size

This is a boolean parameter (True/False). When TRUE, the lot size is checked against RM8 and RM10. When FALSE, the lot size calculation ignores RM8 and RM10 limits. However, margin limits are always checked.

RM13: Safety factor for MARGIN_SO_CALL [1,5]

Lot sizes are always checked against the MARGIN_SO_CALL level value for your account. As a safety, we check against $RM13 \times MARGIN_SO_CALL$ to diminish the chances of a margin call. RM13 must be larger than 1. If a proposed trade violates the safe margin level, lot size is reduced automatically.

RM14: Nbr Simult. symbols in trading (Tester only)

It is likely that you will use this EA in multiple symbols simultaneously in a single MT5 terminal. Since this EA only runs on one symbol at a time, the Tester –during optimization-- can only compute the free margin due the symbol being simulated. In order to simulate the presence of more symbols being active in the live trading, the EA assumes –during the Tester simulation-- that there are RM14 symbols running simultaneously, each one with the same amount of margin required. A reasonable value could be RM14 between 2 and 4, but the user is free to set it outside this range. If RM14=1, then the simulation assumes 100% of the account balance is available for margin.

RM15: % Ini.Dep. target (closeALL, stop EA) |0|>0]

When trading to pass a prop firm challenge, there is always a target to achieve. This target is computed as $RM15 * RM0 / 100$ (RM0 is the initial deposit entered in the input set file). If you are using a prop firm and RM15>0 and the equity has achieved the target, all orders and positions will be closed (no matter what MAGIC number they have) and the EA will stop trading. This is done to avoid momentarily passing the challenge, then reversing. This feature only applies in live trading.

RM16: Req.Margin must be less than Balance/2 (boolean)

If RM16 is true, the lot size calculation makes sure that the required margin is less than half of the account balance; If RM16 is false, this check is not performed. In either case, the require margin level is always checked in the calculation of the lot size. If RM16 is true, the Lot size is automatically adjusted to comply with this constraint.

*****Note1:** Keep in mind that the margin required in Forex is computed as

$$\text{Margin} = \text{Lots} * \text{Contract_Size} / \text{Leverage} * \text{Margin_Rate}$$

therefore, using large lots (i.e., large RM1 with large RM6), will require to reserve large margins for the trade, leaving less balance available for other trades in parallel.

*****Note2:** If there are several symbols using this EA simultaneously, the max loss allowed (daily or account) must be the same in all symbols. That is: RM8*RM9 and RM10*RM11 must be the same in all symbols. If they are different from symbol to symbol, then the EA behavior is likely to be incorrect.

MISCELLANEOUS PARAMETERS

Order Type

There are three order types you can select from:

Market
Stop
Limit

We recommend Market or Stop orders. Each order type uses different parameters shown in parenthesis:

Market order (instant execution, no params)

Stop order (NBACK, WBSO)

Limit order (WBLO,nBackLO,atrLO)

NBACK: # bars back high/low for Stop Orders

When using stop orders the EA set the BUY price just above the high of this # of previous bars. A value of 1 is common.

WBSO: Max # waiting bars for Stop Orders

The EA will wait up to this many bars to trigger the stop order; after that, the order is canceled.

WBLO: Max # waiting bars for Limit Orders

The EA will wait up to this many bars to trigger the limit order; after that, the order is canceled.

nBackLO: for Limit Orders

atrLO: for Limit Orders

In order to set the limit price, the EA will use lows of nBackLO bars in the past and also Ask-atrLO*ATR of current bar.

Max # Open Symbols [0][1:1:n] (Live trading)

The user can run this EA in as many symbols as desired, but because of the margin requirements, there may be cases in which the lot size becomes too small when too many symbols are open (depending on your risk in each trade). One way to avoid very small lot sizes, is to limit the number of symbols that can open positions simultaneously. That number is given in this parameter. If this parameter is 0, there is no limits of how many symbols can be trading simultaneously.

Slippage (points) for closing positions [2,20]

This is the slippage allowed when closing trades.

Waiting time (minutes) before week starts

The user may want to start trading some minutes after the opening of the week (5pm EST). This parameter define the number of minutes to wait before trades are allowed. If the time frame is larger than this waiting time, a full time frame candle goes without trades.

Round trip commission per lot (\$/lot) (estimate)

This is an estimated value used in the Tester when optimization is performed. It is ignored during live trading.

Allow trades to BUY, or SELL, or Both

The user can force the EA to send only BUY orders, or only SELL orders, or Both kinds of orders. We recommend to allow Both kinds of orders.

Display Permitted daily losses on chart

When TRUE, the permitted daily loss is displayed in the chart, and updated on every tick. This takes cpu time, and is not recommended to be used for all open charts in the MT5 terminal.

Magic number. If 0, autogenerate MAGIC number

The user can prescribe a MAGIC number here. If this is set to zero, a unique MAGIC number is generated by the EA for each symbol in the MT5 terminal. We recommend to set this to 0. The auto-generated MAGIC number will be unique depending on the symbol, time frame, and other values.

NEWS HANDLING PARAMETERS (Live trading only)

The logic behind this section is that users can prevent (if so desired) to open trades during a window around an important news (as some prop firms require). They can also manage any position already open when the news window begins. The user may want to move SL before the news starts in case there is a big move in price caused by the important news. The user may want to close open positions as well.

Minutes before News to take actions

Set the number of minutes before the news when the EA should start taking news action. This is the beginning of the news window. News actions are considered at every tick.

Minutes after News to restart trading

Set the number of minutes after the news when the EA should end taking news action. This is the end of the news window. News actions are considered at every tick.

Action for profitable open positions

If there is an open position in profit when the news window is beginning, the user have five options to handle the position:

Close it

Keep open, same SL

Keep open, move SL to Break Even

Keep open, move SL to $NAF \times ATR$

Keep open, move SL halfway

NAF is an abbreviation for News Action Factor.

Action for losing open positions

As in the case above, the user has the same five option when the position is losing at the beginning of the news window. The last three options may result in very tight SL.

Close it

Keep open, same SL

Keep open, move SL to Break Even

Keep open, move SL to $NAF \times ATR$

Keep open, move SL halfway

NAF: News Atr Factor to tight SL

NAF is used if the user prefer the last option to move SL to $NAF \times ATR$ away from the current price. The smaller NAF is, the tighter the SL, and the more likely that a small movement in the price will stop-out the position. The last option, namely, “move SL halfway” will move the current SL to the middle between the current SL and the current price.

Reset SL back to original (after news)

When TRUE, and when SL has been moved based on the options above, the SL is reset after the news window is done. The SL is moved back to the same value as before the news window began. When FALSE, SL is not moved back to its original value.

Close pending Orders before News

When TRUE any order pending is close before entering the news window. There are prop firm that allow orders during the news, in which case FALSE is the proper value.

Note 1: the News calendar is read using the built-in function `MqlCalendarEvent` from Metatrader 5. These are the same news you can see in the “Calendar” tab. No other news feed is used.

Note 2: Only “high” importance news are considered, but the following news are also considered as high importance regardless of the calendar says: CPI, PPI, and Employment for the USD symbols; and CPI, PPI for GBP symbols.

Note 3: If you request to close a profitable position during the news window, this action will only happen if the position is profitable inside the news window. For instance, the position might have been a losing trade when the news windows starts (in which case the news action is determined by your choice of losing positions), then the position turns profitable inside the news window which will trigger the news action to close the position if so requested.

Note 4: Keep in mind that important news (or events during a weekend) may create gaps in the price which might jump over the Stop Loss price. In other words, there is no guarantee that a stop loss price would trigger a closing-position order.

WEEKEND HANDLING PARAMETERS

Handling weekends is done with the same philosophy of handling news. The user may want to keep positions open, or close them, or move the SL before the weekend starts (5pm EST). Variables are used in the same way as in the news handling section above. There is no option to restore the SL to its original value after the weekend.

Minutes before Friday 5pm EST to take actions

Weekend action for profitable open positions

Weekend action for losing open positions

Close pending Orders before weekend

*** Note: While optimizing input parameters, if you select to close winning positions on Friday 5 EST, you will see that Friday will become the most profitable day of the week.

Weekend action for profitable open positions

If there is an open position in profit when the weekend starts, the user have four options to handle the position.

Close it

Keep open, same SL

Keep open, move SL to Break Even

Keep open, move SL to $WAF \times ATR$

Keep open, move SL halfway

WAF is an abbreviation for Weekend Action Factor.

Weekend Action for losing open positions

As in the case above, the user has the same four option when the position is losing at the beginning of the weekend. The last three options may result in very tight SL.

Close it

Keep open, same SL

Keep open, move SL to Break Even

Keep open, move SL to $WAF \times ATR$

Keep open, move SL halfway

WAF: Weekend Atr Factor to tight SL

WAF is used if the user prefer the last option to move SL to $WAF \times ATR$ away from the current price. The smaller WAF is, the tighter the SL, and the more likely that a small movement in the price will stop-out the position. The last option, namely, “move SL halfway” will move the current SL to the middle between the current SL and the current price.

Note 1: Keep in mind that important events during a weekend (e.g., a war between countries, the death of a president, etc.) may create gaps in the price which might jump over the Stop Loss price. In other words, there is no guarantee that a stop loss price would trigger a closing-position order.

WITHDRAWAL SIMULATIONS (for the Tester only)

MetaTrader5 allows to simulate withdrawals during a tester simulation. These are the variables to simulate them.

Wmoney [0|[0,Ini.Dep.]

This is the amount in account currency that will be withdrawn (simulated during the testing).

Wmult [≥ 1]

When the account balance is greater than the Initial balance + $Wmult * Wmoney$, a withdraw in the amount of $Wmoney$ will be executed in the tester simulation. For instance, if $Wmoney=500$, $Wmult=2$, Initial Deposit=10000, then there will be a withdrawal of 500 every time the account balance reaches $10000+2*500$ or 11000; after the withdraw, the balance goes to 10500.

CUSTOM MAX OPTIMIZATION

If the user wants to use this section of the inputs, we direct you to the article in

<https://www.mql5.com/en/articles/14365>

titled “A Generic Optimization Formulation (GOF) to Implement Custom Max with Constraints”
In that article you will learn how to use GOF.

The default set file --provided along with this User Manual-- has the optimization settings we use to optimize parameters every week or every two weeks. The user may try to set the optimization in any way preferred. This is why this is called a “Generic” optimization formulation.

Optimization Problem

The following screenshots show the setup of the optimization problem using GOF. User is welcome to experiment once you get familiar with it.

There are three objectives:

maximize the Annual Return %, with a target of 100% and a weight of 100.

maximize the win rate, with a target of 100%, and a weight of 10.

maximize the recovery factor, with a target of 10, and a weight of 10.

===== Custom Max Optimization =====	
Read this article: https://www.mql5.com/en/articles/14365	
- Build Custom Objective to Maximize:	
<input checked="" type="checkbox"/> Select Objective Function to Maximize 1:	1] Annual Return %
<input checked="" type="checkbox"/> Target 1	100
<input checked="" type="checkbox"/> Weight 1	100
<input checked="" type="checkbox"/> Select Objective Function to Maximize 2:	10] Win Rate %
<input checked="" type="checkbox"/> Target 2	100
<input checked="" type="checkbox"/> Weight 2	10
<input checked="" type="checkbox"/> Select Objective Function to Maximize 3:	6] Recovery Factor
<input checked="" type="checkbox"/> Target 3	10
<input checked="" type="checkbox"/> Weight 3	10
<input checked="" type="checkbox"/> Select Objective Function to Maximize 4:	0] None
<input checked="" type="checkbox"/> Target 4	50
<input checked="" type="checkbox"/> Weight 4	25
<input checked="" type="checkbox"/> Select Objective Function to Maximize 5:	0] None
<input checked="" type="checkbox"/> Target 5	1
<input checked="" type="checkbox"/> Weight 5	1

There are seven hard constraints:

- Daily Loss as % of Initial Deposit must be less or equal to 5%
- Account Loss as a % of Initial Deposit must be less or equal to 10%
- Win rate % greater or equal to 50%
- Consecutive losing trades to be less or equal to 5
- Recovery factor to be more or equal to 4
- Number of Maximum Daily Loss events per year to be less or equal to 0.

- Number of trades per week to be greater or equal to 0.5
- Equity Drawdown less than 10%
- MCPredProf/MCPredStd greater than 3

- Hard Constraints:		
<input checked="" type="checkbox"/> if false, all constraints are ignored		true
<input checked="" type="checkbox"/> Select Constraint Function 1:		1] Daily Loss % InitDep (RM8 recommended)
<input checked="" type="checkbox"/> Type 1		<= Less or equal to
<input checked="" type="checkbox"/> Bound Value 1		5
<input checked="" type="checkbox"/> Select Constraint Function 2:		2] Accnt Loss % InitDep (RM10 recommended)
<input checked="" type="checkbox"/> Type 2		<= Less or equal to
<input checked="" type="checkbox"/> Bound Value 2		10
<input checked="" type="checkbox"/> Select Constraint Function 3:		5] Win Rate %
<input checked="" type="checkbox"/> Type 3		>= Greater or equal to
<input checked="" type="checkbox"/> Bound Value 3		50
<input checked="" type="checkbox"/> Select Constraint Function 4:		4] Consecutive losing trades
<input checked="" type="checkbox"/> Type 4		<= Less or equal to
<input checked="" type="checkbox"/> Bound Value 4		5
<input checked="" type="checkbox"/> Select Constraint Function 5:		7] Recov Factor
<input checked="" type="checkbox"/> Type 5		>= Greater or equal to
<input checked="" type="checkbox"/> Bound Value 5		4
<input checked="" type="checkbox"/> Select Constraint Function 6:		18] # MaxDailyLoss Events/Year
<input checked="" type="checkbox"/> Type 6		<= Less or equal to
<input checked="" type="checkbox"/> Bound Value 6		0
<input checked="" type="checkbox"/> Select Constraint Function 7:		6] # trades/week
<input checked="" type="checkbox"/> Type 7		>= Greater or equal to
<input checked="" type="checkbox"/> Bound Value 7		0.5
<input checked="" type="checkbox"/> Select Constraint Function 8:		3] Equity DrawDown %
<input checked="" type="checkbox"/> Type 8		<= Less or equal to
<input checked="" type="checkbox"/> Bound Value 8		10
<input checked="" type="checkbox"/> Select Constraint Function 9:		21] MCPredProf/MCPredStd
<input checked="" type="checkbox"/> Type 9		>= Greater or equal to
<input checked="" type="checkbox"/> Bound Value 9		3
<input checked="" type="checkbox"/> Select Constraint Function 10:		0] None
<input checked="" type="checkbox"/> Type 10		<= Less or equal to
<input checked="" type="checkbox"/> Bound Value 10		10

The Misc Optimization Params are:

----- Misc Optimization Params -----					
<input checked="" type="checkbox"/> Choose Result-column's decimals	WinRate %				
<input checked="" type="checkbox"/> Choose capital method for Risk of Ruin	Loss Allowed by Prop...				
<input checked="" type="checkbox"/> Custom Value for Risk of Ruin (if needed)	0				
<input checked="" type="checkbox"/> Draw summary on chart	false				
<input checked="" type="checkbox"/> Print summary on journal	true				
<input checked="" type="checkbox"/> Subtract Largest Profit from Netprofit	true				
<input checked="" type="checkbox"/> Add Largest Loss to Net profit	false				
<input checked="" type="checkbox"/> Multiplier for Objectives (k_o)	100000				
<input checked="" type="checkbox"/> Multiplier for Penalties (k_p)	100				

OnTester Summary Explained

The summary printed in the Journal tab has many of the same variables in the Backtest tab, plus some more. Here we explain those additional variables that are not in the Backtest tab. For more information please read the GOF article mentioned above.

- TestLength(Yrs): the length of the test measured in years
- Prof (inc Expenses): profit minus commissions, swaps, and largest win (if requested).
- P/R*100: total profit divided by the sum of all risks taken, times 100.
- LR:corr intercept slope std: linear regression output including correlation coefficient, intercept, slope and standard deviation
- RRR: Reward to Risk ratio
- RoRA %: risk of ruin the account
- Sortino: the Sortino ratio
- # maxDailyLoss/Year: number of times that the max daily loss was reached during the test, per year.
- GoodPred %: percent of trades where the first bar went in the winning direction
- TotSwap: total amount spent in Swap fees
- max Simult. Positions Margin: largest amount hold in margin due to the simultaneous positions in parallel.
- Slopen_pip: n avg max min: n=the number of times a SL was set in an entry (this include orders not opened); avg: average of pip per trade; max,min: largest and smallest number of pips of SL at entry.
- MCPredProf, MCPredStd are the Monte Carlo simulation predicted mean and standard deviation.

===== OnTester SUMMARY =====									
--- BO on XAUUSD - 2025.05.08 23:58:56 ----									
C100000	1.00		Init.Dep.	TestLength(Yrs)					
C103870	103.9	%		MetaTrader Profit, Ann.Ret.%					
C98290	98.3	%		Prof(inc Expenses), Ann.Ret.%					
C-380.6	(-0.4%)			C0.0(0.0%)		minEqty-IniDep	minBal-IniDep		
4.2%	C6689	C6689		mxBalDDrel%	mxBalDDrel	mxBalDD			
5.5%	C5921	C7382		mxEqDDrel%	mxEqDDrel	mxEqDD			
14.07	2.99	11.09		RecovFactor	PrfFct	ShrpeR			
64%	58/90	32		WinRate%	Wins/TotTrd	#Losses			
1.65	0.71	C1154		RRR Expectancy	Exp.PayOff				
C2692.1	C-1633.5			AvgWinTrade	AvgLossTrade				
C1159.2	+/-	C2483.9		Test Trades: Mean +/- Std					
C5580.1	C-2097.1			BestTrade	WorstTrade				
1	3		#AvgLossTrd	#conseqLossTrd					
C-4206	0.0	C1862		mxDalyLoss	#mxDalyLoss/Yr	avgRisk/trade			
0.00	1.70	58.66	%		RoRA%	Sortino	Prof/Risk%		
0.98	96521	1200.97	3.27		LR:corr	intercept	slope	std	
1.49	3.82	54%		AvgVol	LargestVol	GoodPred%			
1:33	194%	C-454.7		AccLevrge	MinMargLvl%	TotSwap			
C31192.50				maxSimultPositionsMargin					
90	181.9	691.2	51.9		SLopen_pip:	n	avg	max	min
C103737.4	+/-	C22075.5		MCPredict Trades: Mean +/- Std					
---- Objective Functions ----									
Value	Target	Weight	Contribution%						
9.83e+01	100.00	100.00	82.7%		MAX_AnnRetPct				
6.44e+01	100.00	10.00	5.4%		MAX_WinRatePct				
1.41e+01	10.00	10.00	11.8%		MAX_RecovFact				
----- Constraints: (Actual vs Bound) -----									
Pass	:	MaxDailyLoss	=	-4.21%	vs	-5.00%			
Pass	:	MaxLossTrade	=	-2.10%	vs	-5.00%			
Pass	:	MaxAccLoss	=	-0.38%	vs	-10.00%			
Pass	:	WinRate	=	64%	vs	55%			
Pass	:	nbrConsecLossTrades	=	3	vs	5			
Pass	:	RecovFactor	=	14.07	vs	3.00			
Pass	:	# MaxDaiLossPerYear	=	0.00	vs	0.00			
Pass	:	Trades/Week	=	1.7	vs	0.5			
Pass	:	maxEqtyDD/IniDep	=	7.38%	vs	10.00%			
Pass	:	MCPred(Prof/Std)	=	4.70	vs	2.00			
Pass	:	Expectancy	=	0.71	vs	0.25			

GLOBAL VARIABLES

Metatrader 5 has a feature to store permanent global variables. They can be seen by pressing F3 key on the keyboard, or going to Tools → Global Variables. This EA uses global variables to preserve some values in case there is a crash (power off, OS freezes, MT5 freezes, etc). When you restart the MT5 terminal, the stored global variables are read and used as if there was no crash.

One global variable users should be aware of is called *accountNbr_ResumeTradingOnDay*, where *accountNumber* is, obviously, the account number. In MT5, days are numbered starting from 0=Sunday to 6=Saturday. The EA won't trade anything if the actual day of the week is less than the global variable *accountNbr_ResumeTradingOnDay*.

The default value of *accountNbr_ResumeTradingOnDay* is zero, but there are two situations in which they may not be zero:

1) The Max daily loss limit is very close; the EA will close all trades, and stop trading. There will be a *Sleep()* command that will make the EA wait until the next day when trades can resume. In this case *accountNbr_ResumeTradingOnDay* is not zero temporarily until the EA wakes up from the *Sleep()* command.

2) The Max account loss limit is very close; the EA will close all trades, stop trading, remove the EA from the chart.

Global variables can be changed manually (via F3 key), but user must be aware of what the consequences are of changing them.

Global Variables				
Variable	Value	Time		
1/2 5019462724_ResumeTradingOnDay	0.0	2025.03.07 11:59		
1/2 GBPJPY_H2_5019492844_CurrTicket	-999.0	2025.03.07 12:33		
1/2 GBPJPY_H2_5019492844_g_slAtOpen	-999.0	2025.03.07 11:00		
1/2 USDJPY_H2_5019480834_CurrTicket	-999.0	2025.03.07 12:00		
1/2 USDJPY_H2_5019480834_g_slAtOpen	-999.0	2025.03.07 11:59		
1/2 USDJPY_H2_5019492964_CurrTicket	-999.0	2025.03.07 12:33		
1/2 XAUUSD_H3_5019496074_CurrTicket	-999.0	2025.03.07 12:33		
1/2 XAUUSD_H3_5019496074_g_slAtOpen	-999.0	2025.03.07 11:00		