

# PA\_SUPER\_ROBOT

## USER MANUAL



# 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:

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

CP=current price

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.

## PA\_SUPER\_ROBOT STRATEGY PARAMETERS

*ip1\_d [0,0.5,3]*

*p2\_d [0.25,0.25,2]*

*p1\_e [2,1,6]*

These parameters controls the **PA\_SUPER\_ROBOT** entry strategy. The larger they are, the less entries opportunities will be. User may play with different ranges and steps in the optimization, but the range values shown above will work in most Forex symbols.

### Note:

**PA\_SUPER\_ROBOT** allows trading with non-Forex symbols, such as ETFs, indices, equities, but be aware that **PA\_SUPER\_ROBOT** was not tested for non-Forex symbols. Use **PA\_SUPER\_ROBOT** 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 used. 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  $TP\_opNbars \leq 0$ , 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.

#### *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.

#### *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.

#### *TPopGap: (TP\_opGapFrac)*

After finding the latest gap in price, it sets the TP to a convex combination between the low and high of the gaped candle. If no gap is found, the TP is set to the high of the previous candle minus 200 points. TP\_opGapFrac is the convex combination weight. If near 1, it sets the TP closer to the gaped candle high. It then compares the TP computed with the highest price in the previous TP\_opNbars bars, and selects the lower of the two values.

#### *TPopFractal: (TP\_opAtr)*

If  $TP\_opAtr > 0$ , then TP is computed by adding  $TP\_opAtr * ATR$  to OP, but it is not lower than the most recent fractal High price. 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.

## 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\_opGapFrac*

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

## SL <TRAILING> PARAMETERS

### *SL Trailing Method (at each new bar)*

There are several method 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.

#### *Static: SL won't move*

SL won't move until the position is closed by TP. Be aware that if  $TP=0$ , this Static method will only gives 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\_trAccBE, SL\_trAcc (recommended method)*

SL is moved up when profit is larger than  $SL\_trAccBE * (TP - OP)$ . 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  $SL\_trTrgFac*(OP-SL)$ , 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  $2*SL\_trATR*Atr[1]$ , SL is moved up  $SL\_trATR*Atr[1]$ , then, only when the close price moves up by  $SL\_trATR*Atr[1]$ , SL is moved  $SL\_trATR*Atr[1]$ . Notice that the step  $SL\_trATR*Atr[1]$  is not constant, but depends on the last candle ATR.

*DistAtr: SL\_trTrigFac,SL\_trATR*

When the position is in profit by  $SL\_trTrgFac*Atr[1]$ , SL is moved up  $SL\_trATR*Atr[1]$ , then, only when the close price moves up by  $SL\_trATR*Atr[1]$ , SL is moved  $SL\_trATR*Atr[1]$ . Notice that the step  $SL\_trATR*Atr[1]$  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\_trTrigFac*

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

*MA20200:SL\_trTrigFac,SL\_trConvex01*

When the position is in profit by an amount larger than  $SL\_trTrigFac*(TP-OP)$ , SL is set in between MA20 and MA200.  $SL\_trConvex01$  controls how close to MA20 the SL is set.

*Percent: SL\_trPricePct*

When the position is in profit by an amount larger than  $2*SL\_trPricePct*(TP-OP)/100$ , SL is set  $SL\_trPricePct*(TP-OP)/100$  away from the close price.

*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  $T * SL\_trTrigFac * (OP - SLatEntry)$  the SL is moved up to be away from the CP an amount equal to  $(1 - T * SL\_trConvex01) * (OP - SLatEntry)$ . The quantity “T” is the number of rounded times  $CP - OP \geq SL\_trTrigFac * (OP - SLatEntry)$ . 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.

## 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|[2,1,10](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 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\_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  $\text{SL\_Gain2BE} * (\text{OP} - \text{SLatEntry})$ , it will be moved SL to Break Even. If the position has a profit more than  $(\text{SL\_Gain2BE} + \text{SL\_Gain2CL}) * (\text{OP} - \text{SLatEntry})$ , 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  $(100 - \text{SL\_GainVpct}) / 100$  of the original position lot size. The TP price remains the same. This two operations are equivalent to “taking profit and partially closing” the original lot size. It works in Netting accounts without violating any rule. 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  $\text{SL\_Gain2BE} > 0$ ,  $\text{SL\_Gain2CL} = 0$ .

## 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](http://www.ftmo.com))

TheFundedTrader-like rules (go to [www.thefundedtraderprogram.com](http://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

### *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.

### *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 % Ini. Dep. (RM: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.

The parameters  $RM1, \dots, RM7$  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  $0.05 * 0.9 * 10,000 = 450$ , 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  $0.10 * 0.9 * 10,000 = 900$ . 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 checked against  $RM13 * MARGIN\_SO\_CALL$  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.

**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)**

When trading to pass a prop firm challenge, there is always a target to achieve. This target is computed as  $RM14 * InitialDeposit / 100$ .

**RM16: Stop EA if target achieved (live trading)**

In live trading only, if RM16 is TRUE and if the equity achieves the target given by RM15, the EA closes all trades and stops trading. If RM16 is FALSE, RM15 has no effect, and the EA keep running. This feature is useful when trying to pass Prop. Firms challenges.

**RM17: Req.Margin must be less than Balance/2**

If RM17 is true, the lot size calculation makes sure that the required margin is less than half of the account balance; If RM17 is false, this check is not performed. In either case, the require margin level is always checked in the calculation of the lot size.

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

$$Margin = Lots * Contract\_Size / Leverage * Margin\_Rate$$

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 four order types you can select from:

Market  
Stop  
Limit

We recommend Market or stop orders 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 where 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 have open positions simultaneously. That number is given in this parameter.

### *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.

## 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 four 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 four option when the position is losing at the beginning of the news window:

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, but the following news are also considered as high importance regardless of the calendar importance tag: 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 within 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 withing the news window which will trigger the news action to close the position if so requested.

## 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. 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 before 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*

## 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* [ $\geq 1$ ]

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: <a href="https://www.mql5.com/en/articles/14365">https://www.mql5.com/en/articles/14365</a>	
- 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

<b>- Hard Constraints:</b>	
<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:

<b>----- Misc Optimization Params -----</b>	
<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 =====									
--- PASR on XAUUSD - 2025.03.21 22:56:57 ----									
C100000	1.00		Init.Dep.	TestLength(Yrs)					
C37865	38.0	%		MetaTrader Profit, Ann.Ret.%					
C33858	34.0	%		Prof(inc Expenses), Ann.Ret.%					
C-357.1	(-0.4%)			C0.0(0.0%)		minEqty-IniDep	minBal-IniDep		
4.7%	C5361	C5361		mxBalDDrel%	mxBalDDrel	mxBalDD			
5.9%	C6886	C6886		mxEqDDrel%	mxEqDDrel	mxEqDD			
5.50	2.72	5.60		RecovFactor	PrfFct	ShrpeR			
62%	24/39	15		WinRate%	Wins/TotTrd	#Losses			
1.70	C971			RRR	Exp.PayOff				
C2493.9	C-1466.0			AvgWinTrade	AvgLossTrade				
C987.9	+/-	C2183.5		Test Trades: Mean +/-	Std				
C4007.0	C-2058.4			BestTrade	WorstTrade				
2	3			#AvgLossTrd	#consqLossTrd				
C-2015	0.0	C1998		mxDalyLoss	#mxDalyLoss/Yr	avgRisk/trade			
0.00	1.73	43.45	%		RoRA%	Sortino	Prof/Risk%		
0.92	102906	821.48	2.48		LR:corr	intercept	slope	std	
0.65	1.05	49%		AvgVol	LargestVol	GoodPred%			
1:33	766%	C-664.7		AccLevrge	MinMargLvl%	TotSwap			
C7359.07				maxSimultPositionsMargin					
39	325.1	488.2	189.9		SLOpen_pip:	n	avg	max	min
C37836.0	+/-	C8349.6		MCPredict	Trades: Mean +/-	Std			
Objective Functions:									
Value	Target	Weight	Contribution%						
3.40e+01	100.00	100.00	74.5%		MAX_AnnRetPct				
6.15e+01	100.00	10.00	13.5%		MAX_WinRatePct				
5.50e+00	10.00	10.00	12.1%		MAX_RecovFact				
Constraints: (Actual vs Bound)									
Pass	:	MaxDailyLoss	=	-2.02%	vs	-5.00%			
Pass	:	MaxLossTrade	=	-2.06%	vs	-5.00%			
Pass	:	MaxAccLoss	=	-0.36%	vs	-10.00%			
Pass	:	WinRate	=	62%	vs	50%			
Pass	:	nbrConsecLossTrades	=	3	vs	5			
Pass	:	RecovFactor	=	5.50	vs	4.00			
Pass	:	# MaxDaiLossPerYear	=	0.00	vs	0.00			
Pass	:	Trades/Week	=	0.8	vs	0.5			
Pass	:	Eqty_DrawDown	=	5.95%	vs	10.00%			
Pass	:	MC(Pred/Std)	=	4.53	vs	3.00			

## 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		