

<https://www.garot.com/trading/>

01 – Introduction

This is the informal video blog.

In part 3 of this VLOG series, we will build upon the Expert Advisor (EA) code from part 2.

I call this part: Placing Pending Orders

(page 2)

In this VLOG we will:

- Check for an open signal
- Place two pending orders: STOP BUY and STOP SELL
- Draw lines on the Chart to visualize the orders

Now let's take a look at the additions to the code.

02 – Code View in WinMerge

To show the additions to the code from version 02 to 03, I will do a quick walk-thru summary using WinMerge. After this summary, I will go through the added code in more detail in MetaEditor.

As I said, I am using the WinMerge application, which is Open Source software under the GNU General Public License.

The left side shows version 02, and the right side shows 03. The differences are highlighted. Added code is shown by gray emptiness on the left and highlighted yellow on the right.

- We immediately see a new input variable and a constant.
- In the OnTick() event, we add a section that determines if a position already exists.
- CheckOpenSignal is the “open logic,” to determine the conditions when orders should be placed.
- DrawLine() is used to specify the trigger price and SL for each open order.
- IsPendingBuySignal() and IsPendingSellSignal() define the ENTRY for the strategy. These are the specific conditions for when either pending order is placed.
- OrderStop() places the STOP BUY and STOP SELL pending orders.
- SelectPosition() simply determines if a position is already open or not.

03 – Code View in MetaEditor

I'm back in MetaEditor.

We've added an input for the ATR Factor, which I will describe soon.

The constant TRADE_VOLUME sets a fixed lot size of 0.10 for the orders placed. I prefer to use a small, fixed lot size when developing strategies. This allows me to better compare when a strategy is better or worse than another strategy. You don't want too many variables when testing strategies.

Let's take a look at the code addition to the OnTick() event.

Here we determine if a position is already open. If it is open, we have a placeholder to manage a trailing stop (this will be covered in the next video). If no position exists, we check for an open signal. Let's take a look at the code to determine if a position is open.

Essentially, SelectPosition() is a simple wrapper around the native MQL5 function PositionSelect(), which determines if a position for a given symbol exists. Our wrapper function adds another check:

If a position for the given symbol exists, is it "one of ours?"

By this is meant that an existing position may have been opened manually or by a different EA. Our EA is only concerned with positions opened by itself. This is where we use the "magic number," which should be different for every EA you write. Any well-written EA will have this "no-clobber" functionality.

Now let's take a look at CheckForOpenSignal() function. This function checks for both a pending buy and pending sell signal each time it is called, which is every bar. If we get either signal, a STOP order is placed. We then draw lines on the chart to indicate the trigger price and SL for the STOP orders.

Let's take a look at IsPendingBuySignal() to see the specific conditions where a BUY STOP is placed.

This function defines the ENTRY of the strategy. The first line determines if the current filled bar is a Doji or not. If it is not a Doji, the function returns. If it is a Doji, we determine the trigger price and SL for the order. For this strategy, the trigger price is simply the HIGH of the Doji. For the SL, I start with the LOW of the Doji, but then adjust it using the ATR times a scaling factor. Obviously, if the scaling factor is zero, the SL is simply the LOW of the Doji.

Remember that the ATR Factor (SL_ATR_FACTOR) is an input parameter, which means I can vary it using the optimizer in the MT5 Strategy Tester.

Now let's look at OrderStop(), which places the actual order. We pass in all the necessary parameters to place the order. For the purposes of this EA, it is assumed that there will be no TP set; and pending orders are valid until they are closed. We can easily see that the function we have written is simply a wrapper to the BuyStop() and SellStop() methods of the CTrade class.

Note that this function is very primitive. For example:

1. It has no checks for invalid passed arguments
2. It does not ensure that the TP and SL meet minimum broker server stops levels or freeze levels
3. No check for the amount of free margin

Also, we assume the default "filling type," and the expiration is hard-coded to GTC. So there is much room for improvement to this function.

Also note that, at this time, we don't delete existing pending orders when an order is filled. Typically, I would want to do that. We'll see the ramifications of this when we run the Strategy Tester.

04 – Running In The Tester

Now I am back in MetaTrader 5. I wish to open the Strategy Tester by pressing CTRL+R.

I want to Visualize, and I will use similar parameters that I used in the previous video.

After hitting the [Start] button, we got some actual trades!

If I scroll to the bottom of the [History] tab, I see that we are not profitable. This is actually to be

expected for two reasons.

1. Our only EXIT at this point is a SL.
2. When a pending order is filled, we don't delete the other pending order.

We can see how this 2nd issue is a problem by looking at the very first trade. Double click to jump to this trade on the chart.

We see the expected yellow arrow to show us a Doji. So far so good.

We see our Target Prices shown by two dotted LightBlue lines, and our SL's are depicted as two dotted Brown lines. Excellent.

On the very next candle, we see that the BUY order is converted to a market order by the little blue arrow. (Note that our EA didn't draw this arrow. The blue and red arrows are drawn automatically on the charts by MT5.)

Then we see that prices go against us until the other pending order's Trigger Price is hit.

WE DON'T WANT THIS!

So, in the next video, we will look at fixing these two described issues.

05 – Wrapping up

(read from slide)

Final Notes (not in video)

- If you have any suggestions for improvement to this script or this video series, do let me know. Programming is a bit of an art form, and each programmer will do it their own way.