

BullCharts charting software

BullScript — worked example — PART2

Adding prompts for MA periods in the Simple custom MA indicator ribbon

Overview

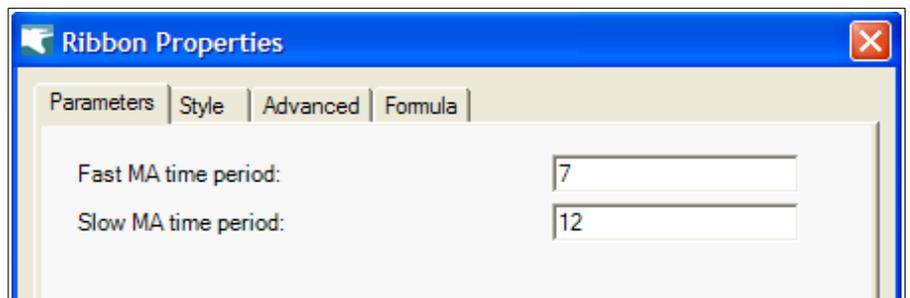
This is PART 2 of a worked example. In Part 1 we showed step-by-step how to create a ribbon indicator related to the Moving Average Cross-over chart indicator. An example of this is shown in Figure 1 below. For more details about that, refer to the earlier document.



Figure 1: Showing the new ribbon indicator

In this PART 2 we will add some extra script to the indicator so as to prompt for the number of MA periods that we want to be plotted. This is because in our own testing of strategy ideas we often want to test out variations of parameters to see what seems to work best for us. So prompting for the MA periods will easily allow us to quickly review different values on the chart.

Robert Brain
BullCharts User Group
convenor

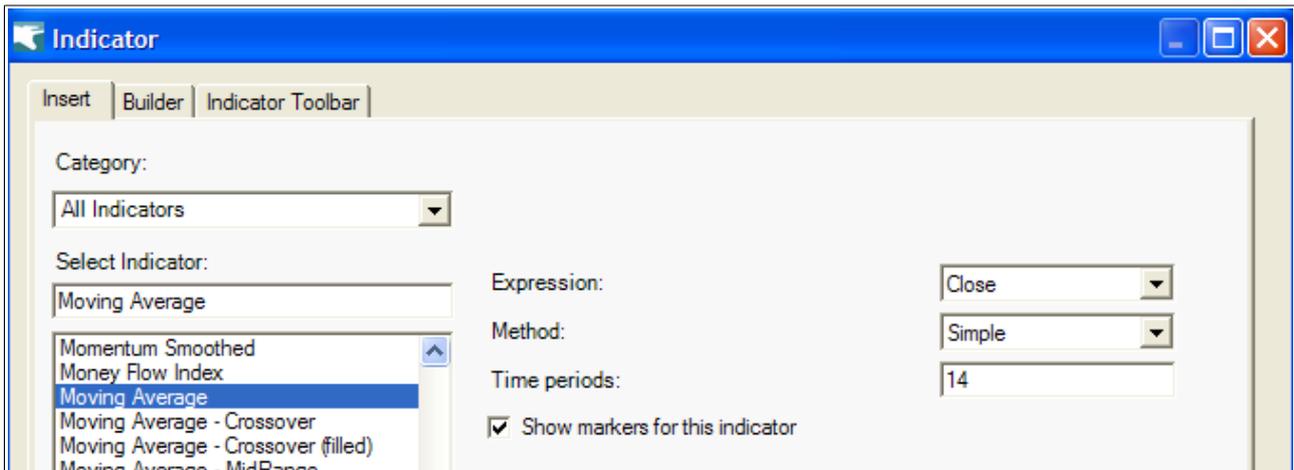


The BullScript code

For the original script that we created in Part 1, see the Appendix section on page 4 below.

Adding the prompts for time periods

Our objective here is to add some code to this script so as to produce an on-screen prompt like the one for the Moving Average (MA) indicator. The screen shot here shows three prompts, but at this stage we only want to prompt for the “Time periods” value. Let’s assume that we will leave the MA “expression” based on Close prices, and the MA “Method” will be Simple.



The script formula to achieve this in this Moving Average indicator dialogue box is:

```
n := input("Time periods",14,1);
```

So all we need to do is drop something similar to this into our existing script. But to facilitate this we need to revise the script to include some **SCRIPT VARIABLES**. So, let’s introduce two variables to store the number of time periods for the slow moving average and the fast moving average as follows. Let’s call these variables:

```
SlowMA  
FastMA
```

Then the following existing formula

```
is7GT12 := MA(C,7,E) > MA(C,12,E);
```

can be revised as follows (noting the differences):

```
is_FAST_GT_SLOW := MA(C,FastMA,E) > MA(C,SlowMA,E);
```

Note that “is_FAST_GT_SLOW” is just one **VARIABLE** name.

Then we just need to add in the prompt to get the values for **SlowMA** and **FastMA**. It might look like this:

```
FastMA:= input("Fast MA time period",7,1);  
SlowMA:= input("Slow MA time period",12,1);
```

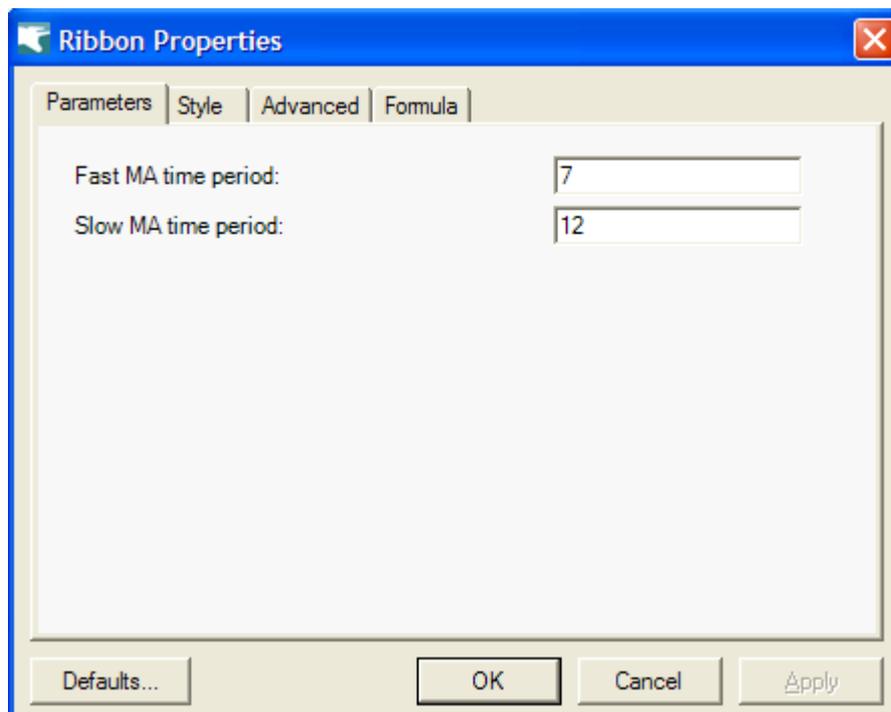
Note that the two values shown here (7 and 12) are specified here as the default values if no others are entered. The value “1” included is the minimum valid number that is allowed to be entered. See the BullScript help for the INPUT function for more details.

The resulting updated script

The updated script which now prompts for the two MA period values is shown here (which you can copy and paste into your own BullCharts indicator builder script dialogue).

```
[NAME = STATUS; TARGET = RIBBON ; author = Bleakley, Des ] ;
FastMA:= input("Fast MA time period",7,1) ;
SlowMA:= input("Slow MA time period",12,1) ;
is_FAST_GT_SLOW := MA(C,FastMA,E) > MA(C,SlowMA,E) ;
[color = RGB(200,255,200) ; name = Good ;
  fillstyle = small grid ] ;
is_FAST_GT_SLOW ;
[color = RGB(255,200,160) ; name = Bad ;
  fillstyle = small grid] ;
NOT is_FAST_GT_SLOW ;
```

The “Insert indicator “ dialogue box which prompts for the two periods is as shown in the following screen shot.



The original script from Part 1 of this exercise along with explanatory comments is in the Appendix section on the next page.

APPENDIX

The original BullScript

Following is the original BullScript that was used for this particular ribbon. Refer to the explanatory comments below, referring to each "Line No."

The list of BullScript statements

Line No.	script statements
1	<code>[NAME = STATUS; TARGET = RIBBON ; author = Bleakley, Des];</code>
2	<code>is7GT12 := MA(C,7,E) > MA(C,12,E) ;</code>
3	<code>{The next statement contains three ATTRIBUTES to be applied to the price PLOT.}</code>
4	<code>[color = RGB(200,255,200); name = Good 7 on 12; fillstyle = small grid];</code>
5	<code>is7GT12;</code>
6	<code>{This statement "plots" the values that were calculated above.}</code>
7	<code>[color = RGB(255,200,160); name = Bad 7 on 12; fillstyle = small grid];</code>
8	<code>NOT is7GT12;</code>
9	<code>{This statement "plots" the opposite of the values.}</code>

1. The line number 1 is simply a statement containing three ATTRIBUTES (more detail later).
2. Line 2 is very important. It uses the FUNCTION: "MA(C,7,E)". This refers to the Moving Average (MA) of the security being plotted, based on Close prices, 7-periods, and the Exponential MA. And remember that this FUNCTION refers to all of the Moving Average values for the security being plotted, and not just the value for only one day, or a week. It compares one moving average "MA(C,7,E)" to another "MA(C,12,E)", if the value of the first one "is greater than" the value of the second one, then this comparison evaluates to be "true", and the VARIABLE at the left of this equation (the FORMULA) takes on the value one, otherwise it is set to zero. Remember that the two characters := are the ASSIGNMENT OPERATOR (a fancy "equals" sign). The comparison on the right of the OPERATOR is an EXPRESSION.
3. {Anything inside curly braces is a COMMENT.}
4. Line 4 specifies some ATTRIBUTES for the plot which is about to be generated.
5. This statement simply "plots" the values that were calculated using the FORMULA above.
6. { (another comment) }
7. Line 7 specifies some ATTRIBUTES for the second part of the plot which is about to be generated.
8. This statement specifies what is to be plotted when the values are zero.