

Better Volume Indicator - EasyLanguage - Version 19 August 2012

Inputs: LowVol(True), ClimaxUp(True), ClimaxDown(True), Churn(True), ClimaxChurn(False), LowVolColor(Yellow), ClimaxUpColor(Red), ClimaxDownColor(White), ChurnColor(Green), ClimaxChurnColor(Magenta), Color(Cyan), UseUpTicks(True), Use2Bars(True), ShowAvg(False), AvgColor(Red);

Variables: BarColor(Cyan), Lookback(20);

BarColor = Color;

If BarType > 1 then begin

 If C > O and Range <> 0 then Value1 = (Range/(2*Range+O-C))*V;

 If C < O and Range <> 0 then Value1 = ((Range+C-O)/(2*Range+C-O))*V;

 If C = O then Value1 = 0.5*V;

 Value2 = V-Value1;

End;

If BarType <= 1 and UseUpTicks = False then begin

 If C > O and Range <> 0 then Value1 = (Range/(2*Range+O-C))*Ticks;

 If C < O and Range <> 0 then Value1 = ((Range+C-O)/(2*Range+C-O))*Ticks;

 If C = O then Value1 = 0.5*Ticks;

 Value2 = Ticks-Value1;

End;

If BarType <= 1 and UseUpTicks then begin

 Value1 = UpTicks;

 Value2 = DownTicks;

End;

Value3 = AbsValue(Value1+Value2);

Value4 = Value1*Range;

Value5 = (Value1-Value2)*Range;

Value6 = Value2*Range;

Value7 = (Value2-Value1)*Range;

If Range <> 0 then begin

 Value8 = Value1/Range;

 Value9 = (Value1-Value2)/Range;

 Value10 = Value2/Range;

 Value11 = (Value2-Value1)/Range;

 Value12 = Value3/Range;

End;

If Use2Bars then begin

 Value13 = Value3+Value3[1];

 Value14 = (Value1+Value1[1])*(Highest(H,2)-Lowest(L,2));

 Value15 = (Value1+Value1[1]-Value2-Value2[1])*(Highest(H,2)-Lowest(L,2));

 Value16 = (Value2+Value2[1])*(Highest(H,2)-Lowest(L,2));

 Value17 = (Value2+Value2[1]-Value1-Value1[1])*(Highest(H,2)-Lowest(L,2));

 If Highest(H,2) <> Lowest(L,2) then begin

 Value18 = (Value1+Value1[1])/(Highest(H,2)-Lowest(L,2));

 Value19 = (Value1+Value1[1]-Value2-Value2[1])/(Highest(H,2)-Lowest(L,2));

 Value20 = (Value2+Value2[1])/(Highest(H,2)-Lowest(L,2));

 Value21 = (Value2+Value2[1]-Value1-Value1[1])/(Highest(H,2)-Lowest(L,2));

 Value22 = Value13/(Highest(H,2)-Lowest(L,2));

 End;

End;

```

Condition1 = Value3 = Lowest(Value3,Lookback);
Condition2 = Value4 = Highest(Value4,Lookback) and C > O;
Condition3 = Value5 = Highest(Value5,Lookback) and C > O;
Condition4 = Value6 = Highest(Value6,Lookback) and C < O;
Condition5 = Value7 = Highest(Value7,Lookback) and C < O;
Condition6 = Value8 = Lowest(Value8,Lookback) and C < O;
Condition7 = Value9 = Lowest(Value9,Lookback) and C < O;
Condition8 = Value10 = Lowest(Value10,Lookback) and C > O;
Condition9 = Value11 = Lowest(Value11,Lookback) and C > O;
Condition10 = Value12 = Highest(Value12,Lookback);
If Use2Bars then begin
    Condition11 = Value13 = Lowest(Value13,Lookback);
    Condition12 = Value14 = Highest(Value14,Lookback) and C > O and C[1] > O[1];
    Condition13 = Value15 = Highest(Value15,Lookback) and C > O and C[1] > O[1];
    Condition14 = Value16 = Highest(Value16,Lookback) and C < O and C[1] < O[1];
    Condition15 = Value17 = Highest(Value17,Lookback) and C < O and C[1] < O[1];
    Condition16 = Value18 = Lowest(Value18,Lookback) and C < O and C[1] < O[1];
    Condition17 = Value19 = Lowest(Value19,Lookback) and C < O and C[1] < O[1];
    Condition18 = Value20 = Lowest(Value20,Lookback) and C > O and C[1] > O[1];
    Condition19 = Value21 = Lowest(Value21,Lookback) and C > O and C[1] > O[1];
    Condition20 = Value22 = Highest(Value22,Lookback);
End;

If BarType > 1 then begin
    If LowVol and (Condition1 or Condition11) then BarColor = LowVolColor;
    If ClimaxUp and (Condition2 or Condition3 or Condition8 or Condition9 or Condition12 or
Condition13 or Condition18 or Condition19) then BarColor = ClimaxUpColor;
    If ClimaxDown and (Condition4 or Condition5 or Condition6 or Condition7 or Condition14
or Condition15 or Condition16 or Condition17) then BarColor = ClimaxDownColor;
    If Churn and (Condition10 or Condition20) then BarColor = ChurnColor;
    If ClimaxChurn and (Condition10 or Condition20) and (Condition2 or Condition3 or
Condition4 or Condition5 or Condition6 or Condition7 or Condition8 or Condition9 or
Condition12 or Condition13 or Condition14 or Condition15 or Condition16 or
Condition17 or Condition18 or Condition19) then BarColor = ClimaxChurnColor;
End;

If BarType <= 1 then begin
    If LowVol and (Condition1 or (Condition11 and D=D[1])) then BarColor = LowVolColor;
    If ClimaxUp and (Condition2 or Condition3 or Condition8 or Condition9 or ((Condition12
or Condition13 or Condition18 or Condition19) and D=D[1])) then BarColor = ClimaxUpColor;
    If ClimaxDown and (Condition4 or Condition5 or Condition6 or Condition7 or
((Condition14 or Condition15 or Condition16 or Condition17) and D=D[1])) then BarColor =
ClimaxDownColor;
    If Churn and (Condition10 or (Condition20 and D=D[1])) then BarColor = ChurnColor;
    If ClimaxChurn and (Condition10 or (Condition20 and D=D[1])) and (Condition2 or
Condition3 or Condition4 or Condition5 or Condition6 or Condition7 or Condition8 or Condition9
or
((Condition12 or Condition13 or Condition14 or Condition15 or Condition16 or
Condition17 or Condition18 or Condition19) and D=D[1])) then BarColor = ClimaxChurnColor;
End;

Plot1(Value3,"Volume",BarColor);

```

```
If ShowAvg then Plot2(Average(Value3,200),"Avg",AvgColor);
```

```
{ Change Log: }
```

```
{23 November 2007 - Added LowChurn colored volume bars      }  
{19 April 2008 - Got rid of LowChurn and replaced with ClimaxDown }  
{19 April 2008 - Added open & close conditions with ClimaxUp and ClimaxDown }  
{19 April 2008 - Added different calculations for tick/intra-day charts  }  
{26 June 2008 - Added ability to turn average volume line on and off    }  
{5 July 2008 - Changed Lookback from a variable to an input      }  
{13 July 2008 - Added 2 bar climax, churn and low volume conditions    }  
{4 September 2008 - Changed daily bars calculation to match tick/intra-day }  
{25 January 2009 - Added condition total volume (Value3) could not be -ve }  
{19 August 2012 - Changed calculation for Intra-day + UseUpTicks = False }
```

Better Volume PaintBar - EasyLanguage - Version 19 August 2012

Inputs: LowVol(True), ClimaxUp(True), ClimaxDown(True), Churn(True), ClimaxChurn(False),
LowVolColor(Yellow), ClimaxUpColor(Red), ClimaxDownColor(White), ChurnColor(Blue),
ClimaxChurnColor(Magenta), Color(Green), UseUpTicks(True), Use2Bars(True);
Variables: BarColor(Green), Lookback(20);

BarColor = Color;

If BarType > 1 then begin

 If C > O and Range <> 0 then Value1 = (Range/(2*Range+O-C))*V;

 If C < O and Range <> 0 then Value1 = ((Range+C-O)/(2*Range+C-O))*V;

 If C = O then Value1 = 0.5*V;

 Value2 = V-Value1;

End;

If BarType <= 1 and UseUpTicks = False then begin

 If C > O and Range <> 0 then Value1 = (Range/(2*Range+O-C))*Ticks;

 If C < O and Range <> 0 then Value1 = ((Range+C-O)/(2*Range+C-O))*Ticks;

 If C = O then Value1 = 0.5*Ticks;

 Value2 = Ticks-Value1;

End;

If BarType <= 1 and UseUpTicks then begin

 Value1 = UpTicks;

 Value2 = DownTicks;

End;

Value3 = AbsValue(Value1+Value2);

Value4 = Value1*Range;

Value5 = (Value1-Value2)*Range;

Value6 = Value2*Range;

Value7 = (Value2-Value1)*Range;

If Range <> 0 then begin

 Value8 = Value1/Range;

 Value9 = (Value1-Value2)/Range;

 Value10 = Value2/Range;

 Value11 = (Value2-Value1)/Range;

 Value12 = Value3/Range;

End;

If Use2Bars then begin

 Value13 = Value3+Value3[1];

 Value14 = (Value1+Value1[1])*(Highest(H,2)-Lowest(L,2));

 Value15 = (Value1+Value1[1]-Value2-Value2[1])*(Highest(H,2)-Lowest(L,2));

 Value16 = (Value2+Value2[1])*(Highest(H,2)-Lowest(L,2));

 Value17 = (Value2+Value2[1]-Value1-Value1[1])*(Highest(H,2)-Lowest(L,2));

 If Highest(H,2) <> Lowest(L,2) then begin

 Value18 = (Value1+Value1[1])/(Highest(H,2)-Lowest(L,2));

 Value19 = (Value1+Value1[1]-Value2-Value2[1])/(Highest(H,2)-Lowest(L,2));

 Value20 = (Value2+Value2[1])/(Highest(H,2)-Lowest(L,2));

 Value21 = (Value2+Value2[1]-Value1-Value1[1])/(Highest(H,2)-Lowest(L,2));

 Value22 = Value13/(Highest(H,2)-Lowest(L,2));

 End;

End;

```

Condition1 = Value3 = Lowest(Value3,Lookback);
Condition2 = Value4 = Highest(Value4,Lookback) and C > O;
Condition3 = Value5 = Highest(Value5,Lookback) and C > O;
Condition4 = Value6 = Highest(Value6,Lookback) and C < O;
Condition5 = Value7 = Highest(Value7,Lookback) and C < O;
Condition6 = Value8 = Lowest(Value8,Lookback) and C < O;
Condition7 = Value9 = Lowest(Value9,Lookback) and C < O;
Condition8 = Value10 = Lowest(Value10,Lookback) and C > O;
Condition9 = Value11 = Lowest(Value11,Lookback) and C > O;
Condition10 = Value12 = Highest(Value12,Lookback);
If Use2Bars then begin
    Condition11 = Value13 = Lowest(Value13,Lookback);
    Condition12 = Value14 = Highest(Value14,Lookback) and C > O and C[1] > O[1];
    Condition13 = Value15 = Highest(Value15,Lookback) and C > O and C[1] > O[1];
    Condition14 = Value16 = Highest(Value16,Lookback) and C < O and C[1] < O[1];
    Condition15 = Value17 = Highest(Value17,Lookback) and C < O and C[1] < O[1];
    Condition16 = Value18 = Lowest(Value18,Lookback) and C < O and C[1] < O[1];
    Condition17 = Value19 = Lowest(Value19,Lookback) and C < O and C[1] < O[1];
    Condition18 = Value20 = Lowest(Value20,Lookback) and C > O and C[1] > O[1];
    Condition19 = Value21 = Lowest(Value21,Lookback) and C > O and C[1] > O[1];
    Condition20 = Value22 = Highest(Value22,Lookback);
End;

If BarType > 1 then begin
    If LowVol and (Condition1 or Condition11) then BarColor = LowVolColor;
    If ClimaxUp and (Condition2 or Condition3 or Condition8 or Condition9 or Condition12 or
Condition13 or Condition18 or Condition19) then BarColor = ClimaxUpColor;
    If ClimaxDown and (Condition4 or Condition5 or Condition6 or Condition7 or Condition14
or Condition15 or Condition16 or Condition17) then BarColor = ClimaxDownColor;
    If Churn and (Condition10 or Condition20) then BarColor = ChurnColor;
    If ClimaxChurn and (Condition10 or Condition20) and (Condition2 or Condition3 or
Condition4 or Condition5 or Condition6 or Condition7 or Condition8 or Condition9 or
Condition12 or Condition13 or Condition14 or Condition15 or Condition16 or
Condition17 or Condition18 or Condition19) then BarColor = ClimaxChurnColor;
End;

If BarType <= 1 then begin
    If LowVol and (Condition1 or (Condition11 and D=D[1])) then BarColor = LowVolColor;
    If ClimaxUp and (Condition2 or Condition3 or Condition8 or Condition9 or ((Condition12
or Condition13 or Condition18 or Condition19) and D=D[1])) then BarColor = ClimaxUpColor;
    If ClimaxDown and (Condition4 or Condition5 or Condition6 or Condition7 or
((Condition14 or Condition15 or Condition16 or Condition17) and D=D[1])) then BarColor =
ClimaxDownColor;
    If Churn and (Condition10 or (Condition20 and D=D[1])) then BarColor = ChurnColor;
    If ClimaxChurn and (Condition10 or (Condition20 and D=D[1])) and (Condition2 or
Condition3 or Condition4 or Condition5 or Condition6 or Condition7 or Condition8 or Condition9
or
((Condition12 or Condition13 or Condition14 or Condition15 or Condition16 or
Condition17 or Condition18 or Condition19) and D=D[1])) then BarColor = ClimaxChurnColor;
End;

If BarColor <> Color then PlotPaintBar(H,L,O,C,"BetterVol",BarColor);

```

```
{ Change Log: }
{23 November 2007 - Added LowChurn colored volume bars      }
{28 March 2008 - Added ability to turn on and off different colored bars }
{28 March 2008 - Got rid of redundant code calculations      }
{19 April 2008 - Got rid of LowChurn and replaced with ClimaxDown }
{19 April 2008 - Added open & close conditions with ClimaxUp and ClimaxDown }
{19 April 2008 - Added different calculations for tick/intra-day charts }
{13 July 2008 - Added 2 bar climax, churn and low volume conditions }
{4 September 2008 - Changed daily bars calculation to match tick/intra-day }
{25 January 2009 - Added condition total volume (Value3) could not be -ve }
{3 July 2009 - Allowed the default bar coloring to be changed }
{19 August 2012 - Changed calculation for Intra-day + UseUpTicks = False }
```

Better Volume - ProRealTime/ChartNet/CMC - Version Dec. 2009 - Dutchy

```
// Better Volume Indicator
// Original idea by emini-watch.com
// Source : http://emini-watch.com/free-stuff/volume-indicator/ AND
http://emini-watch.com/category/volume-indicator/

// Rewrite for ProRealTime / ChartNet / CMC by Dutchy, march 2009
// Code is provided as public domain, no warranty.
// You can find Dutchy on following fora
// http://www.aktienboard.com/forum/f29/prorealtime-cmc-script-programmierung-t94783/
// http://www.pro-at.com/forums-bourse/sujet-Forums-Partenaires-Forum-ProRealTime-92.html
// Peace

// Better Volume Indicator - Summary
// ClimaxUp (Red) : Top, Start Up Trend, Down Trend Continue
// ClimaxDown (White): Bottom, Up Trend Continue, Start Down Trend
// LowVolume (Yellow): Bottom, Top, Up Trend Continue
// Churn (Green): Bottom, Top, Down Trend Continue
// ClimaxChurn (Magenta): Top, Down Trend Continue

// Parameter: History = Boolean
// Parameter: Use2Bars = Boolean
// Parameter: Lookback = 20

ONCE Condition1 = 0
ONCE Condition2 = 0
ONCE Condition3 = 0
ONCE Condition4 = 0
ONCE Condition5 = 0
ONCE Condition6 = 0
ONCE Condition7 = 0
ONCE Condition8 = 0
ONCE Condition9 = 0
ONCE Condition10 = 0
ONCE Condition11 = 0
ONCE Condition12 = 0
ONCE Condition13 = 0
ONCE Condition14 = 0
ONCE Condition15 = 0
ONCE Condition16 = 0
ONCE Condition17 = 0
ONCE Condition18 = 0
ONCE Condition19 = 0
ONCE Condition20 = 0

VolValue = Volume
LowVolValue = VolValue
ClimaxUpValue = VolValue
ClimaxDownValue = VolValue
ChurnValue = VolValue
ClimaxChurnValue = VolValue
StopVolValue = VolValue
```

```
TrampolineValue = VolValue
AVVolume = Average[Lookback*2](VolValue)
```

```
IF History = 0 THEN
  BI = 400
ELSIF History = 1 THEN
  BI = Lookback
ENDIF
```

```
IF BARINDEX > BI AND Volume <> 0 THEN
```

```
  IF Close > Open THEN
    Value1 = Volume * (Range / (2 * Range + Open - Close))
  ELSIF Close < Open THEN
    Value1 = Volume * ((Range + Close - Open) / (2 * Range + Close - Open))
  ENDIF
  IF Close = Open THEN
    Value1 = 0.5 * Volume
  ENDIF
  Value2 = Volume - Value1
```

```
  Value3 = Value1 + Value2
  Value4 = Value1 * Range
  Value5 = (Value1 - Value2) * Range
  Value6 = Value2 * Range
  Value7 = (Value2 - Value1) * Range
  IF Range <> 0 THEN
    Value8 = Value1 / Range
    Value9 = (Value1 - Value2) / Range
    Value10 = Value2 / Range
    Value11 = (Value2 - Value1) / Range
    Value12 = Value3 / Range
  ENDIF
```

```
  Value13 = Value3 + Value3[1]
  Value14 = (Value1 + Value1[1]) * (Highest[2](High) - Lowest[2](Low))
  Value15 = (Value1 + Value1[1] - Value2 - Value2[1]) * (Highest[2](High) - Lowest[2](Low))
  Value16 = (Value2 + Value2[1]) * (Highest[2](High) - Lowest[2](Low))
  Value17 = (Value2 + Value2[1] - Value1 - Value1[1]) * (Highest[2](High) - Lowest[2](Low))
  IF Highest[2](High) <> Lowest[2](Low) THEN
    Value18 = (Value1 + Value1[1]) / (Highest[2](High) - Lowest[2](Low))
  ENDIF
  Value19 = (Value1 + Value1[1] - Value2 - Value2[1]) / (Highest[2](High) - Lowest[2](Low))
  Value20 = (Value2 + Value2[1]) / (Highest[2](High) - Lowest[2](Low))
  Value21 = (Value2 + Value2[1] - Value1 - Value1[1]) / (Highest[2](High) - Lowest[2](Low))
  Value22 = Value13 / (Highest[2](High) - Lowest[2](Low))
```

```
  IF Use2Bars = 0 THEN
    Condition1 = Value3 = Lowest[Lookback](Value3)
    Condition2 = Value4 = Highest[Lookback](Value4) AND Close > Open
    Condition3 = Value5 = Highest[Lookback](Value5) AND Close > Open
    Condition4 = Value6 = Highest[Lookback](Value6) AND Close < Open
    Condition5 = Value7 = Highest[Lookback](Value7) AND Close < Open
```



```

Condition6 = Value8 = Lowest[Lookback](Value8) AND Close < Open
Condition7 = Value9 = Lowest[Lookback](Value9) AND Close < Open
Condition8 = Value10 = Lowest[Lookback](Value10) AND Close > Open
Condition9 = Value11 = Lowest[Lookback](Value11) AND Close > Open
Condition10 = Value12 = Highest[Lookback](Value12)
ELSIF Use2Bars = 1 THEN
Condition11 = Value13 = Lowest[Lookback](Value13)
Condition12 = Value14 = Highest[Lookback](Value14) AND Close > Open AND Close[1] >
Open[1]
Condition13 = Value15 = Highest[Lookback](Value15) AND Close > Open AND Close[1] >
Open[1]
Condition14 = Value16 = Highest[Lookback](Value16) AND Close < Open AND Close[1] <
Open[1]
Condition15 = Value17 = Highest[Lookback](Value17) AND Close < Open AND Close[1] <
Open[1]
Condition16 = Value18 = Lowest[Lookback](Value18) AND Close < Open AND Close[1] <
Open[1]
Condition17 = Value19 = Lowest[Lookback](Value19) AND Close < Open AND Close[1] <
Open[1]
Condition18 = Value20 = Lowest[Lookback](Value20) AND Close > Open AND Close[1] >
Open[1]
Condition19 = Value21 = Lowest[Lookback](Value21) AND Close > Open AND Close[1] >
Open[1]
Condition20 = Value22 = Highest[Lookback](Value22)
ENDIF

// *****

IF (Condition1 or Condition11) THEN // Yellow
LowVol = LowVolValue
ELSE
LowVol = 0
ENDIF

IF (Condition2 or Condition3 or Condition8 or Condition9 or Condition12 or Condition13 or
Condition18 or Condition19) THEN // Red
ClimaxUp = ClimaxUpValue
IF LowVol <> 0 THEN
ClimaxUp = ClimaxUpValue / 2
ENDIF
ELSE
ClimaxUp = 0
ENDIF

IF (Condition4 or Condition5 or Condition6 or Condition7 or Condition14 or Condition15 or
Condition16 or Condition17) THEN // White
ClimaxDown = ClimaxDownValue
IF LowVol <> 0 OR ClimaxUp <> 0 THEN
ClimaxDown = ClimaxDownValue
ENDIF
ELSE
ClimaxDown = 0
ENDIF

```

```

IF (Condition10 or Condition20) THEN // Green
  Churn = ChurnValue
  IF LowVol <> 0 OR ClimaxUp <> 0 OR ClimaxDown <> 0 THEN
    Churn = ChurnValue / 2
  ENDIF
ELSE
  Churn = 0
ENDIF

//IF (Condition10 or Condition20) AND (Condition2 or Condition3 or Condition4 or Condition5
or Condition6 or Condition7 or Condition8 or Condition9 or Condition12 or Condition13 or
Condition14 or Condition15 or Condition16 or Condition17 or Condition18 or Condition19) THEN
  IF Churn <> 0 AND (ClimaxUp <> 0 OR ClimaxDown <> 0) THEN // Magenta
    ClimaxChurn = ClimaxChurnValue
  ELSE
    ClimaxChurn = 0
  ENDIF

//StopVolume
LocalClosingPosition = 1 - (High - Close) / (High - Low)
IF Volume > Volume[1] AND Range < Range[1] AND ((High > High[1] AND
LocalClosingPosition < 0.4) OR (Low < Low[1] AND LocalClosingPosition > 0.6)) THEN
  StopVol = StopVolValue * 1 / 3 // HigherVolume & LowerRange @ HigherHigh or
LowerLow - DBlue
ELSE
  StopVol = 0
ENDIF

//Trampoline
FOR i = 1 TO 2 // Close together
  IF ((ClimaxChurn[i] > 0 OR ClimaxUp[i] > 0) AND (ClimaxChurn > 0 OR ClimaxUp > 0)) AND
((Close[i] > Open[i] AND Close < Open) OR (Close[i] < Open[i] AND Close > Open)) THEN
    Trampoline = TrampolineValue * 1 / 4
    BREAK
  ELSE
    Trampoline = 0
  ENDIF
NEXT

ENDIF

RETURN Volume COLOURED (0, 200, 200) AS "Volume (LBlue)", LowVol COLOURED (255,
255, 0) AS "LowVol (Yellow Histo): Bottom & Top & UpTrend Cont", ClimaxUp COLOURED (255,
0, 0) AS "Climax Up (Red Histo): Start UpTrend & Top & DownTrend Cont", ClimaxDown
COLOURED (255, 255, 255) AS "ClimaxDown (White Histo): Bottom & UpTrend Cont & Start
Down Trend", Churn COLOURED (0, 155, 50) AS "Churn = HighVol@LowRange (Green Histo):
Bottom & Top & Down Trend Cont", ClimaxChurn COLOURED (255, 0, 255) AS "ClimaxChurn
(Magenta Histo): Top & DownTrend Cont: Brearish", Trampoline COLOURED (0, 0, 255) AS
"Trampoline (DBLue Histo): Reversal", StopVol COLOURED (0, 0, 0) AS "StopVol (Black Histo):
Profit Taking", AVVolume COLOURED (255, 0, 0) AS "AVVolume (Red Line)"

```

Better Volume - MC.NET - Jeff Ellestad

The code is posted and discussed in this thread:

<http://www.multicharts.com/discussion/viewtopic.php?f=20&t=11254>

Jeff Ellestad can be reached via the MultiCharts forum and has already provided a function version of the code and an update to the original indicator.

Better Volume & Better X-Trend - TradingView - rimko

The code is posted and discussed in this thread:

<https://prod-frontend-proxy.tradingview.com/script/HjLUwa5t-Better-X-Trend-Volume/>

Disclaimer: Futures and options trading have large potential rewards, but also large potential risk. You must be aware of the risks and be willing to accept them in order to invest in the futures and options markets. Don't trade with money you can't afford to lose. This is neither a solicitation nor an offer to buy or sell futures, stocks or options. No representation is being made that any account will or is likely to achieve profits or losses similar to those discussed here. The past performance of any trading system or methodology is not necessarily indicative of future results. If you decide to invest real money, all trading decisions should be your own.

CFTC Required Disclaimer: Hypothetical or simulated performance results have certain inherent limitations. Unlike an actual performance record, simulated results do not represent actual trading. Also, since the trades have not actually been executed, the results may have under- or over-compensated for the impact, if any, of certain market factors, such as lack of liquidity. Simulated trading programs in general are also subject to the fact that they are designed with the benefit of hindsight. No representation is being made that any account will or is likely to achieve profits or losses similar to those shown.