

## Indicateur 1

```
//@version=2
```

```
//
```

```
study(title = "Scalping Swing Trading Tool R1-4 by JustUncleL", shorttitle =  
"SCALPSWING R1-4", overlay = true)
```

```
//
```

```
// Revision: 1
```

```
// Original Author: JustUncleL
```

```
//
```

```
// Description:
```

```
// This study project is a Scalping Swing trading Tool designed for a two pane  
TradingView chart layout:
```

```
// - the first pane set to 15min Time Frame;
```

```
// - the second pane set to 1min Time Frame(TF).
```

```
// The tools incorporates the majority of the indicators needed to analyse and scalp  
Trends for Swings,
```

```
// PullBacks and reversals on 15min charts and 1min charts. The set up optionally  
utilies Heikin Ashi
```

```
// candle charts.
```

```
//
```

```
// NOTE: A Pullback is synomous to Retracement, generally a Pullback refers to a  
large Retracement of 100pips
```

```
// or more. In the context of this Tool and any comments related to it, a Pullback will  
be the
```

```
// same as a Retracement.
```

```
//
```

```
// Incorporated within this tool are the following indicators:
```

```
// 1. The following EMAs:
```

```
// - Green = EMA89 (15min TF) = EMA75 (1min TF)
```

```
// - Blue = EMA200 (15min TF) = EMA180 (1min TF)
```

```
// - Black = EMA633 (15min TF) = EMA540 (1min TF)
```

```
// 2. The 10EMA (default) High/Low+Close Price Action Channel (PAC).
```

```
// 3. Fractals
```

```
// 4. HH, LH, LL, HL finder to help with drawing Trend lines and mini Trend Lines.
```

```
// 5. Coloured coded Bar high lighting based on the PAC:
```

```
// - blue = bar closed above PAC
```

```
// - red = bar closed below PAC
```

```
// - gray = bar closed inside PAC
```

```
// - red line = EMA36 of bar close
```

```
// 6. Optionally display Pivot points (disables Fractals automatically when selected).
```

```
// 7. Display EMA5-12 Channel
```

```
// 8. Display EMA12-36 Ribbon
```

```
// 9. Optionally display EMA36 and PAC instead of EMA12-36 Ribbon.
```

```
//
```

```
// Setup and hints:
```

```
//
```

```
// - Set to two pane TradingView chart, set first pane to 15Min and second to 1min.
```

```

// - Set the chart to Heikin Ashi Candles in both panes (optional).
// - I also add "SweetSpot Gold2" indicator to the chart as well to help with support
and resistance
// finding and shows where the important "00" and "0" lines are.
// - Use the EMA200 on the 15min pane as the anchor. So when prices above
EMA200 we only trade long (buy)
// and when prices below the EMA200 we only trade short (sell).
// - On the 15min chart draw any obvious Vertical Trend Lines (VTL), use Pivots
point as a guide.
// - On the 15min chart what we're looking for price to Pullback into the EMA5-12
Channel or EMA12-36 ribbon
// we Trendlines utilising the Pivot points or Fractals to guide your TL drawing.
// - On the 15min chart look for the trend to resume and break through the drawn TL.
The bar color needs to
// change back to the trend direction colour to confirm as a break.
// - Now this break can be traded as a 15min trade or now look to the 1min chart.
// - On the 1min chart draw any Pullback into any of the EMAs.
// - On the 1min chart look for the trend to resume and break through the drawn TL.
The bar color needs to
// change back to the trend direction colour to confirm as a break.
// - Now this break can be traded as a 1min trade.
// - So we are looking for continuation signals in terms of a strong, momentumdriven
pullbacks of the EMA36.
// - The other EMAs are there to check for other Pullbacks when EMA36 is broken.
// - There is also an option to select Pristine (ie Ideal) filtered Fractals, which look
like tents or V shape
// 5-candle patterns. These are actually used to calculate the Pivot points as well.
// - Other than the SweetSpot Gold2 indicator, you should not need any other
indicator to scalp
// for pullbacks.
//
// Revisions:
// R1 by JustUncleL
// - Original version.
//
// References:
// - [RS]Fractals V8 by RicardoSantos
// - Price Action Trading System v0.3 by JustUncleL
// - SweetSpot Gold2 R1 by JustUncleL
// - http://www.swing-trade-stocks.com/pullbacks.html
// - https://www.forexstrategiesresources.com/scalping-forex-strategies/106-1-min-scalping-with-34-exponential-moving-average-channel/
//
// === INPUTS ===
ShowPAC_ = input(true, title="Show Price Action Channel (PAC)")
scolor = input(true, title="Show coloured Bars close relative on PAC")
HiLoLen = input(10,minval=2,title="High Low PAC Length")
ShowEMA12_Channel= input(true)
ShowEMA36_Ribbon = input(true)

```

```

ShowPAC = HiLoLen>30 ? ShowEMA36_Ribbon? false: ShowPAC_:
ShowEMA12_Channel? false : ShowPAC_
ShowPivots = input(true)
ShowPivotLabels = input(true)
// Disable Fractal drawing if Pivots enabled
ShowHHLL_ = input(false)
ShowHHLL = ShowPivots?false:ShowHHLL_
ShowFractals_ = input(true)
ShowFractals = ShowPivots?false:ShowFractals_
ShowFractalLevels =input(false)
//ShowFractalLevels = ShowPivots?false:ShowFractalLevels_
filterBW_ = input(false, title="Filter for Pristine (Ideal) Fractals")
// Need Williams Filter for Pivots.
filterBW = ShowPivots?true:filterBW_

// --- CONSTANTS ---
DodgerBlue = #1E90FF
//
// --- SOURCES ---
close_ = security(ticker, period, close)
open_ = security(ticker, period, open)
high_ = security(ticker, period, high)
low_ = security(ticker, period, low)
//
// === /INPUTS ===

// ||--- Fractal Recognition Functions:
-----||
isRegularFractal(mode) =>
  ret = mode == 1 ? high_[5] < high_[4] and high_[4] < high_[3] and high_[3] >
high_[2] and high_[2] > high_[1] :
  mode == -1 ? low_[5] > low_[4] and low_[4] > low_[3] and low_[3] < low_[2]
and low_[2] < low_[1] : false

isBWFractal(mode) =>
  ret = mode == 1 ? high_[5] < high_[3] and high_[4] < high_[3] and high_[3] >
high_[2] and high_[3] > high_[1] :
  mode == -1 ? low_[5] > low_[3] and low_[4] > low_[3] and low_[3] < low_[2]
and low_[3] < low_[1] : false
// ||-----||

// MA Colour finder for EMA Ribbon plot.
maColor(maBase, ma, maRef) =>
  change(ma)>=0 and maBase>maRef ? DodgerBlue
:change(ma)<0 and maBase>maRef ? maroon
:change(ma)<=0 and maBase<maRef ? red
:change(ma)>=0 and maBase<maRef ? blue
:gray

// === /FUNCTIONS ===

```

```

// === SERIES SETUP ===

// Price action channel
pacC    = ema(close_,HiLoLen)
pacL    = ema(low_,HiLoLen)
pacU    = ema(high_,HiLoLen)

// All other EMAs
EMA05   = ema(close_, 05)
EMA11   = ema(close_, 11)
EMA12   = ema(close_, 12)
EMA15   = ema(close_, 15)
EMA18   = ema(close_, 18)
EMA21   = ema(close_, 21)
EMA24   = ema(close_, 24)
EMA27   = ema(close_, 27)
EMA30   = ema(close_, 30)
EMA33   = ema(close_, 33)
EMA36   = ema(close_, 36)
EMA75   = ema(close_, 75)
EMA89   = ema(close_, 89)
EMA180  = ema(close_, 180)
EMA200  = ema(close_, 200)
EMA540  = ema(close_, 540)
EMA633  = ema(close_, 633)

// === /SERIES ===

// === PLOTTING ===
//
// If selected, Plot the Price Action Channel (PAC) base on EMA high,low and close
L=plot(ShowPAC ?pacL:na, color=gray, linewidth=1, title="High PAC
EMA",transp=50)
U=plot(ShowPAC ?pacU:na, color=gray, linewidth=1, title="Low PAC
EMA",transp=50)
C=plot(ShowPAC ?pacC:na, color=lime, linewidth=1, title="Close PAC
EMA",transp=0)
fill(L,U, color=gray,transp=92,title="Fill HiLo PAC")

// Colour bars according to the close position relative to the PAC selected.
bColour  = close>=pacU? blue : close<=pacL? red : gray
barcolor(scolor?bColour:na, title = "Bar Colours")

// Draw the EMA12 ribbon
ema05=plot(ShowEMA12_Channel?EMA05:na,
color=blue,linewidth=1,transp=92,title="EMA05")
ema11=plot(ShowEMA12_Channel?EMA11:na,
color=blue,linewidth=1,transp=92,title="EMA11")
fill(ema05,ema11, color=blue,transp=92,title="Fill EMA5-12")

```

```

//
// If this is the 1min Time Frame select 15* EMAs
emaFast = isintraday? interval==1? EMA75 : EMA89 : EMA89
emaMedium = isintraday? interval==1? EMA180 : EMA200 : EMA200
emaSlow = isintraday? interval==1? EMA540 : EMA633 : EMA633

plot(emaFast, color=green,linewidth=3,transp=20,title="EMA fast")
plot(emaMedium, color=blue,linewidth=3,transp=20,title="EMA medium")
plot(emaSlow, color=black,linewidth=3,transp=20,title="EMA slow")

// Draw the EMA36 ribbon
plot( ShowEMA36_Ribbon?EMA12:na, color=maColor(EMA12,EMA12,EMA36),
style=line, title="MA12", linewidth=2,transp=20)
plot( ShowEMA36_Ribbon?EMA15:na, color=maColor(EMA12,EMA15,EMA36),
style=line, title="MA15", linewidth=1,transp=20)
plot( ShowEMA36_Ribbon?EMA18:na, color=maColor(EMA12,EMA18,EMA36),
style=line, title="MA18", linewidth=1,transp=20)
plot( ShowEMA36_Ribbon?EMA21:na, color=maColor(EMA12,EMA21,EMA36),
style=line, title="MA21", linewidth=1,transp=20)
plot( ShowEMA36_Ribbon?EMA24:na, color=maColor(EMA12,EMA24,EMA36),
style=line, title="MA24", linewidth=1,transp=20)
plot( ShowEMA36_Ribbon?EMA27:na, color=maColor(EMA12,EMA27,EMA36),
style=line, title="MA27", linewidth=1,transp=20)
plot( ShowEMA36_Ribbon?EMA30:na, color=maColor(EMA12,EMA30,EMA36),
style=line, title="MA30", linewidth=1,transp=20)
plot( ShowEMA36_Ribbon?EMA33:na, color=maColor(EMA12,EMA33,EMA36),
style=line, title="MA33", linewidth=1,transp=20)
plot( EMA36, color=ShowEMA36_Ribbon?maColor(EMA12,EMA36,EMA36):red,
style=line, title="MA36", linewidth=2,transp=20)
// === /SERIES ===

// === PLOTTING ===

// II--- Fractal Recognition:
// II-----||
filteredtopf = filterBW ? isRegularFractal(1) : isBWFractal(1)
filteredbotf = filterBW ? isRegularFractal(-1) : isBWFractal(-1)

plotshape(ShowFractals? filteredtopf :na, title='Filtered Top Fractals',
style=shape.triangledown, location=location.abovebar, color=red, offset=-3,transp=0)
plotshape(ShowFractals? filteredbotf :na, title='Filtered Bottom Fractals',
style=shape.triangleup, location=location.belowbar, color=lime, offset=-3,transp=0)

topfractals = filteredtopf ? high_[3] : topfractals[1]
botfractals = filteredbotf ? low_[3] : botfractals[1]

topfcolor = topfractals != topfractals[1] ? na : green
botfcolor = botfractals != botfractals[1] ? na : red

```

```

plot(ShowFractalLevels? topfractals : na, color=topfcolor, transp=0, linewidth=2)
plot(ShowFractalLevels? botfractals : na, color=botfcolor, transp=0, linewidth=2)

// ||-----||
// ||--- Higher Highs, Lower Highs, Higher Lows, Lower Lows
// -----||
higherhigh = filteredtopf == false ? false : ( valuewhen(filteredtopf == true, high_[3],
1) < valuewhen(filteredtopf == true, high_[3], 0) and
      (ShowPivots or valuewhen(filteredtopf == true,
high_[3], 2) < valuewhen(filteredtopf == true, high_[3], 0)))
lowerhigh = filteredtopf == false ? false : ( valuewhen(filteredtopf == true, high_[3], 1)
> valuewhen(filteredtopf == true, high_[3], 0) and
      (ShowPivots or valuewhen(filteredtopf == true, high_[3],
2) > valuewhen(filteredtopf == true, high_[3], 0)))
higherlow = filteredbotf == false ? false : ( valuewhen(filteredbotf == true, low_[3], 1)
< valuewhen(filteredbotf == true, low_[3], 0) and
      (ShowPivots or valuewhen(filteredbotf == true, low_[3],
2) < valuewhen(filteredbotf == true, low_[3], 0)))
lowerlow = filteredbotf == false ? false : ( valuewhen(filteredbotf == true, low_[3], 1) >
valuewhen(filteredbotf == true, low_[3], 0) and
      (ShowPivots or valuewhen(filteredbotf == true, low_[3],
2) > valuewhen(filteredbotf == true, low_[3], 0)))

// If selected show HH/LL on top/below candles.
plotshape(ShowHHLL ? higherhigh : na, title='HH', style=shape.square,
location=location.abovebar, color=maroon, text="[HH]", offset=-3,transp=0)
plotshape(ShowHHLL ? lowerhigh : na, title='LH', style=shape.square,
location=location.abovebar, color=maroon, text="[LH]", offset=-3,transp=0)
plotshape(ShowHHLL ? higherlow : na, title='HL', style=shape.square,
location=location.belowbar, color=green, text="[HL]", offset=-3,transp=0)
plotshape(ShowHHLL ? lowerlow : na, title='LL', style=shape.square,
location=location.belowbar, color=green, text="[LL]", offset=-3,transp=0)

// If selected display Pivot points
plotshape(ShowPivots and ShowPivotLabels? higherhigh : na, title='Higher High',
style=shape.cross, location=location.abovebar, color=maroon, text="[HH]\n[PVT]",
offset=-3,transp=0)
plotshape(ShowPivots and not ShowPivotLabels? higherhigh : na, title='Higher
High+', style=shape.cross, location=location.abovebar, color=maroon,
offset=-3,transp=0)
//
plotshape(ShowPivots and ShowPivotLabels? lowerhigh : na, title='Lower High',
style=shape.cross, location=location.abovebar, color=maroon, text="[LH]\n[PVT]",
offset=-3,transp=0)
plotshape(ShowPivots and not ShowPivotLabels? lowerhigh : na, title='Lower
High+', style=shape.cross, location=location.abovebar, color=maroon,
offset=-3,transp=0)
//
plotshape(ShowPivots and ShowPivotLabels? higherlow : na, title='Higher Low',

```

```

style=shape.cross, location=location.belowbar, color=green, text="[PVT]\n[HL]",
offset=-3,transp=0)
plotshape(ShowPivots and not ShowPivotLabels? higherlow : na, title='Higher
Low+', style=shape.cross, location=location.belowbar, color=green,
offset=-3,transp=0)
//
plotshape(ShowPivots and ShowPivotLabels? lowerlow : na, title='Lower Low',
style=shape.cross, location=location.belowbar, color=green, text="[PVT]\n[LL]",
offset=-3,transp=0)
plotshape(ShowPivots and not ShowPivotLabels? lowerlow : na, title='Lower Low+',
style=shape.cross, location=location.belowbar, color=green, offset=-3,transp=0)
//
// Number candles on Pivot patterns.
plotchar(ShowPivots ? filteredtopf: na, title='High 1u', location=location.abovebar,
color=maroon, char="1", offset=-5,transp=0)
plotchar(ShowPivots ? filteredtopf: na, title='High 2u', location=location.abovebar,
color=maroon, char="2", offset=-4,transp=0)
plotchar(ShowPivots ? filteredtopf: na, title='High 2d', location=location.abovebar,
color=maroon, char="2", offset=-2,transp=0)
plotchar(ShowPivots ? filteredtopf: na, title='High 1d', location=location.abovebar,
color=maroon, char="1", offset=-1,transp=0)
//
plotchar(ShowPivots ? filteredbotf: na, title='Low 1d', location=location.belowbar,
color=green, char="1", offset=-5,transp=0)
plotchar(ShowPivots ? filteredbotf: na, title='Low 2d', location=location.belowbar,
color=green, char="2", offset=-4,transp=0)
plotchar(ShowPivots ? filteredbotf: na, title='Low 2u', location=location.belowbar,
color=green, char="2", offset=-2,transp=0)
plotchar(ShowPivots ? filteredbotf: na, title='Low 1u', location=location.belowbar,
color=green, char="1", offset=-1,transp=0)

// === /PLOTTING ===

// === eof

```

-----

## Indicateur 2

```

study("Ersoy-Kesişme")
plot(close,title="Close")
short = sma(close,1)
long = sma(close,21)
plot(short, color = lime,title="Short",style = areabr)
plot(long, color = red,title="long",style =areabr)
plot(cross(short, long) ? long : na,color=blue,style = cross, linewidth =
3,title="Kesişme")
src = close, len = 1
out = sma(src, len)

```

```

out1 = security(tickerid, 'D', out)
plot(out1,color=blue,title="Günlük-Days",linewidth = 3)
out2 = security(tickerid, 'W', out)
plot(out2,color=red,title="Haftalık-Weekly",linewidth = 3)
out3 = security(tickerid, 'M', out)
plot(out3,color=lime,title="Aylık-Mountly",linewidth = 3)

```

-----

### Indicateur 3

```

//Created By ChrisMoody on 12-26-2014...V3 MAJOR Update on 1-05-2014
//01-05-2014 Major Updates Include:
//FILTERED ENTRIES --- AND AGGRESSIVE FILTERED ENTRIES - HIGHLIGHT
BARS AND ALERTS
//Ability to Change All Bars To Gray, and Plot Entries AND Highlight Bars That Match
The Williams Vix Fix
//Alerts Enabled for 4 Different Criteria
//Ability To Plot Alerts True/False Conditions on top of the WVF Histogram
//Or Get Rid Of the Histogram and just see True/False Alerts Conditions.

```

```

study(title="CM_Williams_Vix_Fix_V3_Ultimate_Filtered_Alerts",
shorttitle="CM_WVF_V3_Ult", overlay=false)
//Inputs Tab Criteria.
pd = input(22, title="LookBack Period Standard Deviation High")
bbl = input(20, title="Bollinger Band Length")
mult = input(2.0 , minval=1, maxval=5, title="Bollinger Band Standard Devaition
Up")
lb = input(50 , title="Look Back Period Percentile High")
ph = input(.85, title="Highest Percentile - 0.90=90%, 0.95=95%, 0.99=99%")
new = input(false, title="-----Highlight Bars Below Use Original Criteria-----" )
sbc = input(true, title="Show Highlight Bar if WVF WAS True and IS Now False")
sbcc = input(false, title="Show Highlight Bar if WVF IS True")
new2 = input(false, title="-----Highlight Bars Below Use FILTERED Criteria-----" )
sbcFilt = input(true, title="Show Highlight Bar For Filtered Entry")
sbcAggr = input(false, title="Show Highlight Bar For AGGRESSIVE Filtered Entry")
new3 = input(false, title="Check Below to turn All Bars Gray, Then Check the Boxes
Above, And your will have Same Colors As VixFix")
sgb = input(false, title="Check Box To Turn Bars Gray?")
//Criteria for Down Trend Definition for Filtered Pivots and Aggressive Filtered Pivots
ltLB = input(40, minval=25, maxval=99, title="Long-Term Look Back Current Bar Has
To Close Below This Value OR Medium Term--Default=40")
mtLB = input(14, minval=10, maxval=20, title="Medium-Term Look Back Current Bar
Has To Close Below This Value OR Long Term--Default=14")
str = input(3, minval=1, maxval=9, title="Entry Price Action Strength--Close > X Bars
Back---Default=3")
//Alerts Instructions and Options Below...Inputs Tab
new4 = input(false, title="-----Turn On/Off ALERTS
Below-----" )
new5 = input(false, title="----To Activate Alerts You HAVE To Check The Boxes Below

```

```

For Any Alert Criteria You Want----")
new6 = input(false, title="----You Can Un Check The Box BELOW To Turn Off the
WVF Histogram And Just See True/False Alert Criteria----")
swvf = input(true, title="Show Williams Vix Fix Histogram, Uncheck to Turn Off!")
sa1 = input(false, title="Show Alert WVF = True?")
sa2 = input(false, title="Show Alert WVF Was True Now False?")
sa3 = input(false, title="Show Alert WVF Filtered?")
sa4 = input(false, title="Show Alert WVF AGGRESSIVE Filter?")

//Williams Vix Fix Formula
wvf = ((highest(close, pd)-low)/(highest(close, pd)))*100
sDev = mult * stdev(wvf, bbl)
midLine = sma(wvf, bbl)
lowerBand = midLine - sDev
upperBand = midLine + sDev
rangeHigh = (highest(wvf, lb)) * ph

//Filtered Bar Criteria
upRange = low > low[1] and close > high[1]
upRange_Aggr = close > close[1] and close > open[1]
//Filtered Criteria
filtered = ((wvf[1] >= upperBand[1] or wvf[1] >= rangeHigh[1]) and (wvf < upperBand
and wvf < rangeHigh))
filtered_Aggr = (wvf[1] >= upperBand[1] or wvf[1] >= rangeHigh[1]) and not (wvf <
upperBand and wvf < rangeHigh)

//Alerts Criteria
alert1 = wvf >= upperBand or wvf >= rangeHigh ? 1 : 0
alert2 = (wvf[1] >= upperBand[1] or wvf[1] >= rangeHigh[1]) and (wvf < upperBand
and wvf < rangeHigh) ? 1 : 0
alert3 = upRange and close > close[str] and (close < close[ltLB] or close <
close[mtLB]) and filtered ? 1 : 0
alert4 = upRange_Aggr and close > close[str] and (close < close[ltLB] or close <
close[mtLB]) and filtered_Aggr ? 1 : 0

//Highlight Bar Criteria
barcolor(sbcAggr and alert4 ? orange : na)
barcolor(sbcFilt and alert3 ? fuchsia : na)
barcolor(sbc and ((wvf[1] >= upperBand[1] or wvf[1] >= rangeHigh[1]) and (wvf <
upperBand and wvf < rangeHigh)) ? aqua : na)
barcolor(sbcc and (wvf >= upperBand or wvf >= rangeHigh) ? lime : na)
barcolor(sgb and close ? gray : na)

//Coloring Criteria of Williams Vix Fix
col = wvf >= upperBand or wvf >= rangeHigh ? lime : gray

//Plots for Williams Vix Fix Histogram and Alerts
plot(swvf and wvf ? wvf * -1 : na, title="Williams Vix Fix", style=columns, linewidth =
4, color=col)
plot(sa1 and alert1 ? alert1 : 0, title="Alert If WVF = True", style=line, linewidth=2,

```

```

color=lime)
plot(sa2 and alert2 ? alert2 : 0, title="Alert If WVF Was True Now False", style=line,
linewidth=2, color=aqua)
plot(sa3 and alert3 ? alert3 : 0, title="Alert Filtered Entry", style=line, linewidth=2,
color=fuchsia)
plot(sa4 and alert4 ? alert4 : 0, title="Alert Aggressive Filtered Entry", style=line,
linewidth=2, color=orange)

```

-----

#### Indicateur 4

```

//
// @author LazyBear
//
// List of my public indicators: http://bit.ly/1LQaPK8
// List of my app-store indicators: http://blog.tradingview.com/?p=970
//
study("Ehlers Center of Gravity Oscillator [LazyBear]", shorttitle="ECGO_LB",
overlay=false, precision=3)
src=input(hl2, title="Source")
length=input(10, title="Length", minval=1, maxval=100)
fr=input(true, title="Fill Osc/Trigger region")
nm0=0, dm0=0
nm1=1<=length ? (nm0+(1 + 0) * src[0]) : nm0, dm1=1<=length ? dm0+src[0] : dm0
nm2=2<=length ? (nm1+(1 + 1) * src[1]) : nm1, dm2=2<=length ? dm1+src[1] : dm1
nm3=3<=length ? (nm2+(1 + 2) * src[2]) : nm2, dm3=3<=length ? dm2+src[2] : dm2
nm4=4<=length ? (nm3+(1 + 3) * src[3]) : nm3, dm4=4<=length ? dm3+src[3] : dm3
nm5=5<=length ? (nm4+(1 + 4) * src[4]) : nm4, dm5=5<=length ? dm4+src[4] : dm4
nm6=6<=length ? (nm5+(1 + 5) * src[5]) : nm5, dm6=6<=length ? dm5+src[5] : dm5
nm7=7<=length ? (nm6+(1 + 6) * src[6]) : nm6, dm7=7<=length ? dm6+src[6] : dm6
nm8=8<=length ? (nm7+(1 + 7) * src[7]) : nm7, dm8=8<=length ? dm7+src[7] : dm7
nm9=9<=length ? (nm8+(1 + 8) * src[8]) : nm8, dm9=9<=length ? dm8+src[8] : dm8
nm10=10<=length ? (nm9+(1 + 9) * src[9]) : nm9, dm10=10<=length ? dm9+src[9] :
dm9
nm11=11<=length ? (nm10+(1 + 10) * src[10]) : nm10, dm11=11<=length ?
dm10+src[10] : dm10
nm12=12<=length ? (nm11+(1 + 11) * src[11]) : nm11, dm12=12<=length ?
dm11+src[11] : dm11
nm13=13<=length ? (nm12+(1 + 12) * src[12]) : nm12, dm13=13<=length ?
dm12+src[12] : dm12
nm14=14<=length ? (nm13+(1 + 13) * src[13]) : nm13, dm14=14<=length ?
dm13+src[13] : dm13
nm15=15<=length ? (nm14+(1 + 14) * src[14]) : nm14, dm15=15<=length ?
dm14+src[14] : dm14
nm16=16<=length ? (nm15+(1 + 15) * src[15]) : nm15, dm16=16<=length ?
dm15+src[15] : dm15
nm17=17<=length ? (nm16+(1 + 16) * src[16]) : nm16, dm17=17<=length ?
dm16+src[16] : dm16
nm18=18<=length ? (nm17+(1 + 17) * src[17]) : nm17, dm18=18<=length ?

```

dm17+src[17] : dm17  
nm19=19<=length ? (nm18+(1 + 18) \* src[18]) : nm18, dm19=19<=length ?  
dm18+src[18] : dm18  
nm20=20<=length ? (nm19+(1 + 19) \* src[19]) : nm19, dm20=20<=length ?  
dm19+src[19] : dm19  
nm21=21<=length ? (nm20+(1 + 20) \* src[20]) : nm20, dm21=21<=length ?  
dm20+src[20] : dm20  
nm22=22<=length ? (nm21+(1 + 21) \* src[21]) : nm21, dm22=22<=length ?  
dm21+src[21] : dm21  
nm23=23<=length ? (nm22+(1 + 22) \* src[22]) : nm22, dm23=23<=length ?  
dm22+src[22] : dm22  
nm24=24<=length ? (nm23+(1 + 23) \* src[23]) : nm23, dm24=24<=length ?  
dm23+src[23] : dm23  
nm25=25<=length ? (nm24+(1 + 24) \* src[24]) : nm24, dm25=25<=length ?  
dm24+src[24] : dm24  
nm26=26<=length ? (nm25+(1 + 25) \* src[25]) : nm25, dm26=26<=length ?  
dm25+src[25] : dm25  
nm27=27<=length ? (nm26+(1 + 26) \* src[26]) : nm26, dm27=27<=length ?  
dm26+src[26] : dm26  
nm28=28<=length ? (nm27+(1 + 27) \* src[27]) : nm27, dm28=28<=length ?  
dm27+src[27] : dm27  
nm29=29<=length ? (nm28+(1 + 28) \* src[28]) : nm28, dm29=29<=length ?  
dm28+src[28] : dm28  
nm30=30<=length ? (nm29+(1 + 29) \* src[29]) : nm29, dm30=30<=length ?  
dm29+src[29] : dm29  
nm31=31<=length ? (nm30+(1 + 30) \* src[30]) : nm30, dm31=31<=length ?  
dm30+src[30] : dm30  
nm32=32<=length ? (nm31+(1 + 31) \* src[31]) : nm31, dm32=32<=length ?  
dm31+src[31] : dm31  
nm33=33<=length ? (nm32+(1 + 32) \* src[32]) : nm32, dm33=33<=length ?  
dm32+src[32] : dm32  
nm34=34<=length ? (nm33+(1 + 33) \* src[33]) : nm33, dm34=34<=length ?  
dm33+src[33] : dm33  
nm35=35<=length ? (nm34+(1 + 34) \* src[34]) : nm34, dm35=35<=length ?  
dm34+src[34] : dm34  
nm36=36<=length ? (nm35+(1 + 35) \* src[35]) : nm35, dm36=36<=length ?  
dm35+src[35] : dm35  
nm37=37<=length ? (nm36+(1 + 36) \* src[36]) : nm36, dm37=37<=length ?  
dm36+src[36] : dm36  
nm38=38<=length ? (nm37+(1 + 37) \* src[37]) : nm37, dm38=38<=length ?  
dm37+src[37] : dm37  
nm39=39<=length ? (nm38+(1 + 38) \* src[38]) : nm38, dm39=39<=length ?  
dm38+src[38] : dm38  
nm40=40<=length ? (nm39+(1 + 39) \* src[39]) : nm39, dm40=40<=length ?  
dm39+src[39] : dm39  
nm41=41<=length ? (nm40+(1 + 40) \* src[40]) : nm40, dm41=41<=length ?  
dm40+src[40] : dm40  
nm42=42<=length ? (nm41+(1 + 41) \* src[41]) : nm41, dm42=42<=length ?  
dm41+src[41] : dm41  
nm43=43<=length ? (nm42+(1 + 42) \* src[42]) : nm42, dm43=43<=length ?

dm42+src[42] : dm42  
nm44=44<=length ? (nm43+(1 + 43) \* src[43]) : nm43, dm44=44<=length ?  
dm43+src[43] : dm43  
nm45=45<=length ? (nm44+(1 + 44) \* src[44]) : nm44, dm45=45<=length ?  
dm44+src[44] : dm44  
nm46=46<=length ? (nm45+(1 + 45) \* src[45]) : nm45, dm46=46<=length ?  
dm45+src[45] : dm45  
nm47=47<=length ? (nm46+(1 + 46) \* src[46]) : nm46, dm47=47<=length ?  
dm46+src[46] : dm46  
nm48=48<=length ? (nm47+(1 + 47) \* src[47]) : nm47, dm48=48<=length ?  
dm47+src[47] : dm47  
nm49=49<=length ? (nm48+(1 + 48) \* src[48]) : nm48, dm49=49<=length ?  
dm48+src[48] : dm48  
nm50=50<=length ? (nm49+(1 + 49) \* src[49]) : nm49, dm50=50<=length ?  
dm49+src[49] : dm49  
nm51=51<=length ? (nm50+(1 + 50) \* src[50]) : nm50, dm51=51<=length ?  
dm50+src[50] : dm50  
nm52=52<=length ? (nm51+(1 + 51) \* src[51]) : nm51, dm52=52<=length ?  
dm51+src[51] : dm51  
nm53=53<=length ? (nm52+(1 + 52) \* src[52]) : nm52, dm53=53<=length ?  
dm52+src[52] : dm52  
nm54=54<=length ? (nm53+(1 + 53) \* src[53]) : nm53, dm54=54<=length ?  
dm53+src[53] : dm53  
nm55=55<=length ? (nm54+(1 + 54) \* src[54]) : nm54, dm55=55<=length ?  
dm54+src[54] : dm54  
nm56=56<=length ? (nm55+(1 + 55) \* src[55]) : nm55, dm56=56<=length ?  
dm55+src[55] : dm55  
nm57=57<=length ? (nm56+(1 + 56) \* src[56]) : nm56, dm57=57<=length ?  
dm56+src[56] : dm56  
nm58=58<=length ? (nm57+(1 + 57) \* src[57]) : nm57, dm58=58<=length ?  
dm57+src[57] : dm57  
nm59=59<=length ? (nm58+(1 + 58) \* src[58]) : nm58, dm59=59<=length ?  
dm58+src[58] : dm58  
nm60=60<=length ? (nm59+(1 + 59) \* src[59]) : nm59, dm60=60<=length ?  
dm59+src[59] : dm59  
nm61=61<=length ? (nm60+(1 + 60) \* src[60]) : nm60, dm61=61<=length ?  
dm60+src[60] : dm60  
nm62=62<=length ? (nm61+(1 + 61) \* src[61]) : nm61, dm62=62<=length ?  
dm61+src[61] : dm61  
nm63=63<=length ? (nm62+(1 + 62) \* src[62]) : nm62, dm63=63<=length ?  
dm62+src[62] : dm62  
nm64=64<=length ? (nm63+(1 + 63) \* src[63]) : nm63, dm64=64<=length ?  
dm63+src[63] : dm63  
nm65=65<=length ? (nm64+(1 + 64) \* src[64]) : nm64, dm65=65<=length ?  
dm64+src[64] : dm64  
nm66=66<=length ? (nm65+(1 + 65) \* src[65]) : nm65, dm66=66<=length ?  
dm65+src[65] : dm65  
nm67=67<=length ? (nm66+(1 + 66) \* src[66]) : nm66, dm67=67<=length ?  
dm66+src[66] : dm66  
nm68=68<=length ? (nm67+(1 + 67) \* src[67]) : nm67, dm68=68<=length ?

dm67+src[67] : dm67  
nm69=69<=length ? (nm68+(1 + 68) \* src[68]) : nm68, dm69=69<=length ?  
dm68+src[68] : dm68  
nm70=70<=length ? (nm69+(1 + 69) \* src[69]) : nm69, dm70=70<=length ?  
dm69+src[69] : dm69  
nm71=71<=length ? (nm70+(1 + 70) \* src[70]) : nm70, dm71=71<=length ?  
dm70+src[70] : dm70  
nm72=72<=length ? (nm71+(1 + 71) \* src[71]) : nm71, dm72=72<=length ?  
dm71+src[71] : dm71  
nm73=73<=length ? (nm72+(1 + 72) \* src[72]) : nm72, dm73=73<=length ?  
dm72+src[72] : dm72  
nm74=74<=length ? (nm73+(1 + 73) \* src[73]) : nm73, dm74=74<=length ?  
dm73+src[73] : dm73  
nm75=75<=length ? (nm74+(1 + 74) \* src[74]) : nm74, dm75=75<=length ?  
dm74+src[74] : dm74  
nm76=76<=length ? (nm75+(1 + 75) \* src[75]) : nm75, dm76=76<=length ?  
dm75+src[75] : dm75  
nm77=77<=length ? (nm76+(1 + 76) \* src[76]) : nm76, dm77=77<=length ?  
dm76+src[76] : dm76  
nm78=78<=length ? (nm77+(1 + 77) \* src[77]) : nm77, dm78=78<=length ?  
dm77+src[77] : dm77  
nm79=79<=length ? (nm78+(1 + 78) \* src[78]) : nm78, dm79=79<=length ?  
dm78+src[78] : dm78  
nm80=80<=length ? (nm79+(1 + 79) \* src[79]) : nm79, dm80=80<=length ?  
dm79+src[79] : dm79  
nm81=81<=length ? (nm80+(1 + 80) \* src[80]) : nm80, dm81=81<=length ?  
dm80+src[80] : dm80  
nm82=82<=length ? (nm81+(1 + 81) \* src[81]) : nm81, dm82=82<=length ?  
dm81+src[81] : dm81  
nm83=83<=length ? (nm82+(1 + 82) \* src[82]) : nm82, dm83=83<=length ?  
dm82+src[82] : dm82  
nm84=84<=length ? (nm83+(1 + 83) \* src[83]) : nm83, dm84=84<=length ?  
dm83+src[83] : dm83  
nm85=85<=length ? (nm84+(1 + 84) \* src[84]) : nm84, dm85=85<=length ?  
dm84+src[84] : dm84  
nm86=86<=length ? (nm85+(1 + 85) \* src[85]) : nm85, dm86=86<=length ?  
dm85+src[85] : dm85  
nm87=87<=length ? (nm86+(1 + 86) \* src[86]) : nm86, dm87=87<=length ?  
dm86+src[86] : dm86  
nm88=88<=length ? (nm87+(1 + 87) \* src[87]) : nm87, dm88=88<=length ?  
dm87+src[87] : dm87  
nm89=89<=length ? (nm88+(1 + 88) \* src[88]) : nm88, dm89=89<=length ?  
dm88+src[88] : dm88  
nm90=90<=length ? (nm89+(1 + 89) \* src[89]) : nm89, dm90=90<=length ?  
dm89+src[89] : dm89  
nm91=91<=length ? (nm90+(1 + 90) \* src[90]) : nm90, dm91=91<=length ?  
dm90+src[90] : dm90  
nm92=92<=length ? (nm91+(1 + 91) \* src[91]) : nm91, dm92=92<=length ?  
dm91+src[91] : dm91  
nm93=93<=length ? (nm92+(1 + 92) \* src[92]) : nm92, dm93=93<=length ?

```

dm92+src[92] : dm92
nm94=94<=length ? (nm93+(1 + 93) * src[93]) : nm93, dm94=94<=length ?
dm93+src[93] : dm93
nm95=95<=length ? (nm94+(1 + 94) * src[94]) : nm94, dm95=95<=length ?
dm94+src[94] : dm94
nm96=96<=length ? (nm95+(1 + 95) * src[95]) : nm95, dm96=96<=length ?
dm95+src[95] : dm95
nm97=97<=length ? (nm96+(1 + 96) * src[96]) : nm96, dm97=97<=length ?
dm96+src[96] : dm96
nm98=98<=length ? (nm97+(1 + 97) * src[97]) : nm97, dm98=98<=length ?
dm97+src[97] : dm97
nm99=99<=length ? (nm98+(1 + 98) * src[98]) : nm98, dm99=99<=length ?
dm98+src[98] : dm98
nm100=100<=length ? (nm99+(1 + 99) * src[99]) : nm99, dm100=100<=length ?
dm99+src[99] : dm99
nm=nm100, dm=dm100
cg = iff(dm != 0, -nm/dm + (length + 1) / 2.0, 0)
t=cg[1]
plot(0, title="ZeroLine", color=gray)
duml=plot(fr?(cg>t?cg:t):na, style=circles, linewidth=0, color=gray, title="Dummy")
cmil=plot(cg, title="EhlersCGO", color=red)
tl=plot(t, title="Trigger",color=green)
fill(cmil, duml, color=red, transp=50, title="NegativeFill")
fill(tl, duml, color=lime, transp=50, title="PositiveFill")
ebc=input(false, title="Color bars?")
bc=ebc?(cg>0? (cg>t?lime:(cg==t?gray:green)) : (cg<t?red:orange)):na
barcolor(bc)

```