//@version=5

indicator(title="(5. MS) Market Structure Break & Order Block", overlay=true, max\_lines\_count=500, max\_bars\_back=4900, max\_boxes\_count=500)

settings = "Settings"

zigzag\_len = input.int(9, "ZigZag Length", group=settings)

show\_zigzag = input.bool(true, "Show Zigzag", group=settings)

fib\_factor = input.float(0.33, "Fib Factor for breakout confirmation", 0, 1, 0.01, group=settings)

var float[] high\_points\_arr = array.new\_float(5)

var int[] high\_index\_arr = array.new\_int(5)

var float[] low\_points\_arr = array.new\_float(5)

var int[] low\_index\_arr = array.new\_int(5)

var box[] bu\_ob\_boxes = array.new\_box(5)

var box[] be\_ob\_boxes = array.new\_box(5)

var box[] bu\_bb\_boxes = array.new\_box(5)

var box[] be\_bb\_boxes = array.new\_box(5)

f\_get\_high(ind) =>

    [array.get(high\_points\_arr, array.size(high\_points\_arr) - 1 - ind), array.get(high\_index\_arr, array.size(high\_index\_arr) - 1 - ind)]

f\_get\_low(ind) =>

    [array.get(low\_points\_arr, array.size(low\_points\_arr) - 1 - ind), array.get(low\_index\_arr, array.size(low\_index\_arr) - 1 - ind)]

f\_main1() =>

    to\_up = high >= ta.highest(zigzag\_len)

    to\_down = low <= ta.lowest(zigzag\_len)

    trend = 1

    trend := nz(trend[1], 1)

    trend := trend == 1 and to\_down ? -1 : trend == -1 and to\_up ? 1 : trend

    last\_trend\_up\_since = ta.barssince(to\_up[1])

    low\_val = ta.lowest(nz(last\_trend\_up\_since > 0 ? last\_trend\_up\_since : 1, 1))

    low\_index = bar\_index - ta.barssince(low\_val == low)

    last\_trend\_down\_since = ta.barssince(to\_down[1])

    high\_val = ta.highest(nz(last\_trend\_down\_since > 0 ? last\_trend\_down\_since : 1, 1))

    high\_index = bar\_index - ta.barssince(high\_val == high)

    if ta.change(trend) != 0

        if trend == 1

            array.push(low\_points\_arr, low\_val)

            array.push(low\_index\_arr, low\_index)

        if trend == -1

            array.push(high\_points\_arr, high\_val)

            array.push(high\_index\_arr, high\_index)

[trend, h0i, h0, l0i, l0, market, h1i, h1, l1i, l1, bu\_ob\_since\_high, bu\_ob\_since\_low, be\_ob\_since\_high, be\_ob\_since\_low] = request.security(syminfo.tickerid, tf, f\_main1())

[alert\_market, be\_bb\_since\_high, be\_bb\_since\_low, bu\_bb\_since\_high, bu\_bb\_since\_low, bu\_ob\_index, bu\_bb\_index, be\_ob\_index, be\_bb\_index] = request.security(syminfo.tickerid, tf, f\_main2())

if ta.change(trend) != 0 and show\_zigzag

if trend == 1

line.new(h0i, h0, l0i, l0)

if trend == -1

line.new(l0i, l0, h0i, h0)

if ta.change(market) != 0

if market == 1

line.new(h1i, h1, h0i, h1, color=color.green, width=2)

label.new(int(math.avg(h1i, l0i)), h1, "MSB", color=color.new(color.black, 100), style=label.style\_label\_down, textcolor=color.green, size=size.small)

bu\_ob = box.new(bu\_ob\_index, bu\_ob\_since\_high, bar\_index + 10, bu\_ob\_since\_low, bgcolor=color.green, border\_color=color.green, text="Bu-OB", text\_color=color.green, text\_halign=text.align\_right, text\_size=size.tiny)

bu\_bb = box.new(bu\_bb\_index, bu\_bb\_since\_high, bar\_index + 10, bu\_bb\_since\_low, bgcolor=color.green, border\_color=color.green, text=l0 < l1 ? "Bu-BB" : "Bu-MB", text\_color=color.green, text\_halign=text.align\_right, text\_size=size.tiny)

array.push(bu\_ob\_boxes, bu\_ob)

array.push(bu\_bb\_boxes, bu\_bb)

if market == -1

line.new(l1i, l1, l0i, l1, color=color.red, width=2)

label.new(int(math.avg(l1i, h0i)), l1, "MSB", color=color.new(color.black, 100), style=label.style\_label\_up, textcolor=color.red, size=size.small)

be\_ob = box.new(be\_ob\_index, be\_ob\_since\_high, bar\_index + 10, be\_ob\_since\_low, bgcolor=color.red, border\_color=color.red, text="Be-OB", text\_color=color.red, text\_halign=text.align\_right, text\_size=size.tiny)

be\_bb = box.new(be\_bb\_index, be\_bb\_since\_high, bar\_index + 10, be\_bb\_since\_low, bgcolor=color.red, border\_color=color.red, text=h0 > h1 ? "Be-BB" : "Be-MB", text\_color=color.red, text\_halign=text.align\_right, text\_size=size.tiny)

array.push(be\_ob\_boxes, be\_ob)

array.push(be\_bb\_boxes, be\_bb)

for bull\_ob in bu\_ob\_boxes

bottom = box.get\_bottom(bull\_ob)

if close < bottom

box.delete(bull\_ob)

else if array.size(bu\_ob\_boxes) == 5

box.delete(array.shift(bu\_ob\_boxes))

else

box.set\_right(bull\_ob, bar\_index + 10)

for bear\_ob in be\_ob\_boxes

top = box.get\_top(bear\_ob)

if close > top

box.delete(bear\_ob)

else if array.size(be\_ob\_boxes) == 5

box.delete(array.shift(be\_ob\_boxes))

else

box.set\_right(bear\_ob, bar\_index + 10)

for bear\_bb in be\_bb\_boxes

top = box.get\_top(bear\_bb)

if close > top

box.delete(bear\_bb)

else if array.size(be\_bb\_boxes) == 5

box.delete(array.shift(be\_bb\_boxes))

else

box.set\_right(bear\_bb, bar\_index + 10)

for bull\_bb in bu\_bb\_boxes

bottom = box.get\_bottom(bull\_bb)

if close < bottom

box.delete(bull\_bb)

else if array.size(bu\_bb\_boxes) == 5

box.delete(array.shift(bu\_bb\_boxes))

else

box.set\_right(bull\_bb, bar\_index + 10)

if ta.change(alert\_market) != 0

alert("MSB", alert.freq\_once\_per\_bar)

save('plot.png')