```
// This Pine Script® code is subject to the terms of the Mozilla Public License 2.0 at https://mozilla.org/MPL/2.0/

//@version=6

indicator("MTF Key-Level FVG Delivery Framework", shorttitle="MTF FVG Delivery", overlay=true, max_labels_count=200)


// --- Key levels ---

usePriorDayLevels = input.bool(true, "Use prior-day high / low", tooltip="Use the previous daily candle high and low as PDH and PDL.", group="Key Levels")

useVAH = input.bool(false, "Use manual VAH", tooltip="Enable the entered value-area high. This indicator does not calculate volume-profile levels.", group="Key Levels")

vahInput = input.float(0.0, "VAH price", tooltip="Enter the value-area high price when manual VAH is enabled.", group="Key Levels")

useVAL = input.bool(false, "Use manual VAL", tooltip="Enable the entered value-area low. This indicator does not calculate volume-profile levels.", group="Key Levels")

valInput = input.float(0.0, "VAL price", tooltip="Enter the value-area low price when manual VAL is enabled.", group="Key Levels")

showKeyLevels = input.bool(false, "Plot active key levels", tooltip="Display all enabled key levels on the chart.", group="Key Levels")


// --- Timeframes ---

contextTfInput = input.timeframe("15", "Context timeframe", tooltip="Timeframe for key-level FVG delivery evidence.", group="Timeframes")

setupTfInput = input.timeframe("5", "Setup timeframe", tooltip="Timeframe for displacement, FVG delivery, and IFVG failure.", group="Timeframes")
```

// --- Filters ---

atrLengthInput = input.int(14, "ATR length", minval=2, maxval=100, tooltip="ATR lookback used to scale FVG and close-through filters.", group="Filters")

minContextFvgAtr = input.float(0.0, "15m minimum FVG size (ATR fraction)", minval=0.0, maxval=2.0, step=0.05, tooltip="Require the 15-minute FVG gap to be at least this fraction of 15-minute ATR; zero disables the filter.", group="Filters")

minContextCloseAtr = input.float(0.10, "15m minimum close beyond level (ATR fraction)", minval=0.0, maxval=2.0, step=0.05, tooltip="Require candle 3 to close this far beyond the touched level, measured in 15-minute ATR.", group="Filters")

minDisplacementCloseAtr = input.float(0.05, "5m displacement close-through (ATR fraction)", minval=0.0, maxval=2.0, step=0.05, tooltip="Require the displacement candle to close beyond the key level by this fraction of 5-minute ATR.", group="Filters")

minSetupFvgAtr = input.float(0.10, "5m minimum FVG size (ATR fraction)", minval=0.0, maxval=2.0, step=0.05, tooltip="Require the delivery FVG size to meet this fraction of 5-minute ATR.", group="Filters")

minIfvgCloseAtr = input.float(0.10, "5m IFVG close-through (ATR fraction)", minval=0.0, maxval=2.0, step=0.05, tooltip="Require an inversion candle to close beyond the opposite FVG boundary by this fraction of 5-minute ATR.", group="Filters")

// --- Sequence windows ---

maxSetupBarsFromDisplacement = input.int(5, "5m max bars: displacement to FVG", minval=1, maxval=30, tooltip="Maximum 5-minute bars between key-

level displacement and directional FVG completion.", group="Sequence Windows")

maxSetupBarsToInversion = input.int(5, "5m max bars: FVG to IFVG", minval=1, maxval=30, tooltip="Maximum 5-minute bars allowed for the delivery FVG to invert decisively.", group="Sequence Windows")

maxEntryBarsAfterIfvg = input.int(15, "1m max bars after IFVG", minval=3, maxval=120, tooltip="Maximum one-minute bars to wait for FVG-123 after the confirmed 5-minute IFVG.", group="Sequence Windows")

contextValidBars = input.int(4, "15m context validity (bars)", minval=1, maxval=30, tooltip="How long a 15-minute context signal remains valid for optional entry confirmation.", group="Sequence Windows")

requireContextForEntry = input.bool(false, "Require matching 15m context", tooltip="When enabled, a 1-minute entry requires a recent 15-minute delivery-evidence signal aligned with the IFVG.", group="Sequence Windows")

// --- Display ---

showContextMarkers = input.bool(true, "Show 15m context markers", tooltip="Mark confirmed 15-minute delivery-evidence signals.", group="Display")

showIfvgMarkers = input.bool(true, "Show 5m IFVG markers", tooltip="Mark confirmed 5-minute inversion signals.", group="Display")

showEntryMarkers = input.bool(true, "Show 1m FVG-123 entry markers", tooltip="Mark a confirmed chart-timeframe FVG after a 5-minute IFVG; use a 1-minute chart for intended timing.", group="Display")

// --- Active key-level series ---

priorDayHigh = request.security(syminfo.tickerid, "D", high[1], gaps=barmerge.gaps_off, lookahead=barmerge.lookahead_on)

```
priorDayLow = request.security(syminfo.tickerid, "D", low[1],
gaps=barmerge.gaps_off, lookahead=barmerge.lookahead_on)

pdh = usePriorDayLevels ? priorDayHigh : na

pdl = usePriorDayLevels ? priorDayLow : na

vah = useVAH and vahInput > 0 ? vahInput : na

val = useVAL and valInput > 0 ? valInput : na


plot(showKeyLevels ? pdh : na, "PDH", color=color.new(#f23645, 25),
linewidth=1, style=plot.style_linebr)

plot(showKeyLevels ? pdl : na, "PDL", color=color.new(#089981, 25),
linewidth=1, style=plot.style_linebr)

plot(showKeyLevels ? vah : na, "VAH", color=color.new(#f59e0b, 20),
linewidth=1, style=plot.style_linebr)

plot(showKeyLevels ? val : na, "VAL", color=color.new(#5b9cf6, 20),
linewidth=1, style=plot.style_linebr)


// --- 15-minute key-level FVG delivery evidence ---

f_touchesPrevTwo(_level) =>

   not na(_level) and ((high[2] >= _level and low[2] <= _level) or (high[1] >=
_level and low[1] <= _level))


f_contextAtLevel(_level, _gapSize, _atr, _bullish) =>

   touched = f_touchesPrevTwo(_level)

   closeDistance = _bullish ? close - _level : _level - close
```

```
    touched and not na(_level) and closeDistance >= _atr * minContextCloseAtr
and _gapSize >= _atr * minContextFvgAtr


f_context15(_pdh, _pdl, _vah, _val) =>
    atr15 = ta.atr(atrLengthInput)
    bullGap = low > high[2]
    bearGap = high < low[2]
    bullGapSize = low - high[2]
    bearGapSize = low[2] - high
    bullEvidence = bullGap and (f_contextAtLevel(_pdl, bullGapSize, atr15, true)
or f_contextAtLevel(_vah, bullGapSize, atr15, true))
    bearEvidence = bearGap and (f_contextAtLevel(_pdh, bearGapSize, atr15,
false) or f_contextAtLevel(_val, bearGapSize, atr15, false))
    [bullEvidence[1], bearEvidence[1]]


[ctxBullConfirmed, ctxBearConfirmed] = request.security(syminfo.tickerid,
contextTfInput, f_context15(pdh, pdl, vah, val), gaps=barmerge.gaps_off,
lookahead=barmerge.lookahead_on)
newContextPeriod = ta.change(time(contextTfInput)) != 0
contextBullEvent = newContextPeriod and ctxBullConfirmed
contextBearEvent = newContextPeriod and ctxBearConfirmed


// --- 5-minute displacement -> FVG -> IFVG state machine ---
f_bullCrossAt(_level, _atr) =>
```

```
    not na(_level) and close[1] <= _level and close > _level + _atr *
minDisplacementCloseAtr


f_bearCrossAt(_level, _atr) =>
    not na(_level) and close[1] >= _level and close < _level - _atr *
minDisplacementCloseAtr


f_setup5(_pdh, _pdl, _vah, _val) =>
    atr5 = ta.atr(atrLengthInput)
    bullGapNow = low > high[2]
    bearGapNow = high < low[2]
    bullGapSizeNow = low - high[2]
    bearGapSizeNow = low[2] - high
    bullDisplacement = f_bullCrossAt(_pdh, atr5) or f_bullCrossAt(_pdl, atr5) or
f_bullCrossAt(_vah, atr5) or f_bullCrossAt(_val, atr5)
    bearDisplacement = f_bearCrossAt(_pdh, atr5) or f_bearCrossAt(_pdl, atr5)
or f_bearCrossAt(_vah, atr5) or f_bearCrossAt(_val, atr5)
    bullLevel = f_bullCrossAt(_pdh, atr5) ? _pdh : f_bullCrossAt(_pdl, atr5) ? _pdl
: f_bullCrossAt(_vah, atr5) ? _vah : _val
    bearLevel = f_bearCrossAt(_pdh, atr5) ? _pdh : f_bearCrossAt(_pdl, atr5) ?
_pdl : f_bearCrossAt(_vah, atr5) ? _vah : _val


    var int displacementDir = 0
    var int displacementBar = na
    var float displacementLevel = na
```

```
var bool waitingForFvg = false
var int fvgDir = 0
var float fvgTop = na
var float fvgBottom = na
var int fvgBar = na
var bool waitingForIfvg = false
var bool longIfvgPulse = false
var bool shortIfvgPulse = false
longIfvgPulse := false
shortIfvgPulse := false

if bullDisplacement or bearDisplacement
   displacementDir := bullDisplacement ? 1 : -1
   displacementBar := bar_index
   displacementLevel := bullDisplacement ? bullLevel : bearLevel
   waitingForFvg := true
   waitingForIfvg := false
   fvgDir := 0
   fvgTop := na
   fvgBottom := na
   fvgBar := na
```

```
    if waitingForFvg and bar_index - displacementBar >
maxSetupBarsFromDisplacement
        waitingForFvg := false

    if waitingForFvg and displacementDir == 1 and bullGapNow and close >
displacementLevel and bullGapSizeNow >= atr5 * minSetupFvgAtr
        fvgDir := 1
        fvgTop := low
        fvgBottom := high[2]
        fvgBar := bar_index
        waitingForFvg := false
        waitingForIfvg := true
    else if waitingForFvg and displacementDir == -1 and bearGapNow and close
< displacementLevel and bearGapSizeNow >= atr5 * minSetupFvgAtr
        fvgDir := -1
        fvgTop := low[2]
        fvgBottom := high
        fvgBar := bar_index
        waitingForFvg := false
        waitingForIfvg := true

    if waitingForIfvg and bar_index - fvgBar > maxSetupBarsToInversion
        waitingForIfvg := false
```

```
    if waitingForIfvg and bar_index > fvgBar and fvgDir == 1 and close < fvgBottom - atr5 * minIfvgCloseAtr
        shortIfvgPulse := true
        waitingForIfvg := false
    else if waitingForIfvg and bar_index > fvgBar and fvgDir == -1 and close > fvgTop + atr5 * minIfvgCloseAtr
        longIfvgPulse := true
        waitingForIfvg := false

    [longIfvgPulse[1], shortIfvgPulse[1]]

[ifvgLongRaw, ifvgShortRaw] = request.security(syminfo.tickerid, setupTfInput, f_setup5(pdh, pdl, vah, val), gaps=barmerge.gaps_off, lookahead=barmerge.lookahead_on)

newSetupPeriod = ta.change(time(setupTfInput)) != 0

ifvgLongEvent = newSetupPeriod and ifvgLongRaw

ifvgShortEvent = newSetupPeriod and ifvgShortRaw

// Restore the earlier compact marker style: small directional arrows with a 15m tag.

plotshape(showContextMarkers and contextBullEvent, title="15m Bullish Delivery Evidence", style=shape.triangleup, location=location.belowbar, color=#089981, size=size.small, text="15m", textcolor=#089981)

plotshape(showContextMarkers and contextBearEvent, title="15m Bearish Delivery Evidence", style=shape.triangledown, location=location.abovebar, color=#f23645, size=size.small, text="15m", textcolor=#f23645)
```

alertcondition(contextBullEvent, "15m Bullish Delivery Evidence", "Confirmed 15-minute bullish key-level FVG delivery evidence.")

alertcondition(contextBearEvent, "15m Bearish Delivery Evidence", "Confirmed 15-minute bearish key-level FVG delivery evidence.")

alertcondition(contextBullEvent or contextBearEvent, "15m Key-Level FVG Delivery Evidence", "Confirmed 15-minute key-level FVG delivery evidence in either direction.")