

```

//@version=6
indicator("Multi-Exchange OI Volume Analysis", overlay=false,
format=format.volume)

// ===== INPUTS =====
string tfOI = input.timeframe("", "OI Timeframe", tooltip="Пусто = текущий
таймфрейм графика")

// Визуализация
bool showCandles = input.bool(true, "Показать свечи OI",
group="Отображение")
bool showCloseLine = input.bool(true, "Показать линию OI",
group="Отображение")
bool showDelta = input.bool(true, "Показать дельту OI",
group="Отображение")
bool showMA = input.bool(true, "Показать МА", group="Отображение")
bool showSignals = input.bool(true, "Показать сигналы входа",
group="Отображение")
int maLength = input.int(20, "Период МА", minval=1,
group="Отображение")

// Выбор бирж
bool useBinance = input.bool(true, "Binance", group="Биржи")
bool useBybit = input.bool(true, "Bybit", group="Биржи")
bool useOKX = input.bool(true, "OKX", group="Биржи")
bool useHyper = input.bool(true, "Hyperliquid", group="Биржи")

// Параметры сигналов
float deltaThreshold = input.float(10.0, "Порог дельты для сигнала (%)",
minval=0.1, step=0.5, group="Сигналы", tooltip="Минимальное изменение OI в %
для генерации сигнала")
int volumeMA = input.int(14, "Период МА для объема", minval=1,
group="Сигналы")
bool filterByPrice = input.bool(true, "Фильтр по движению цены",
group="Сигналы", tooltip="Учитывать направление цены при генерации сигнала")

// Цвета
color colorBullCandle = input.color(color.new(#00BCD4, 0), "Бычья свеча OI",
group="Цвета")
color colorBearCandle = input.color(color.new(#FF5252, 0), "Медвежья свеча
OI", group="Цвета")
color colorDeltaPos = input.color(color.new(#4CAF50, 30), "Дельта +",
group="Цвета")
color colorDeltaNeg = input.color(color.new(#F44336, 30), "Дельта -",
group="Цвета")

```

```

color colorOILine      = input.color(color.new(#2196F3, 0), "Линия OI",
group="Цвета")
color colorMA          = input.color(color.new(#FF9800, 0), "MA",
group="Цвета")
color colorBuySignal   = input.color(color.new(#00E676, 0), "Сигнал LONG",
group="Цвета")
color colorSellSignal  = input.color(color.new(#FF1744, 0), "Сигнал SHORT",
group="Цвета")

// ===== HELPERS =====
nz0(float x) => na(x) ? 0.0 : x
isValid(float x) => not na(x)

// ===== TIMEFRAME =====
string tf = (tfOI == "" ? timeframe.period : tfOI)

// ===== BUILD SYMBOLS =====
string base = syminfo.basecurrency

// Формат: EXCHANGE:BASEUSDT.P_OI (для перпетуальных фьючерсов)
string symBIN = "BINANCE:" + base + "USDT.P_OI"
string symBYB = "BYBIT:" + base + "USDT.P_OI"
string symOKX = "OKX:" + base + "USDT.P_OI"
string symHYP = "HYPERLIQUID:" + base + "USDT.P_OI"

// ===== REQUEST OI DATA =====
// Binance
float oBIN = useBinance ? request.security(symBIN, tf, open,
ignore_invalid_symbol=true) : na
float hBIN = useBinance ? request.security(symBIN, tf, high,
ignore_invalid_symbol=true) : na
float lBIN = useBinance ? request.security(symBIN, tf, low,
ignore_invalid_symbol=true) : na
float cBIN = useBinance ? request.security(symBIN, tf, close,
ignore_invalid_symbol=true) : na

// Bybit
float oBYB = useBybit ? request.security(symBYB, tf, open,
ignore_invalid_symbol=true) : na
float hBYB = useBybit ? request.security(symBYB, tf, high,
ignore_invalid_symbol=true) : na
float lBYB = useBybit ? request.security(symBYB, tf, low,
ignore_invalid_symbol=true) : na
float cBYB = useBybit ? request.security(symBYB, tf, close,
ignore_invalid_symbol=true) : na

```

```

// OKX
float oOKX = useOKX ? request.security(symOKX, tf, open,
ignore_invalid_symbol=true) : na
float hOKX = useOKX ? request.security(symOKX, tf, high,
ignore_invalid_symbol=true) : na
float lOKX = useOKX ? request.security(symOKX, tf, low,
ignore_invalid_symbol=true) : na
float cOKX = useOKX ? request.security(symOKX, tf, close,
ignore_invalid_symbol=true) : na

// Hyperliquid
float oHYP = useHyper ? request.security(symHYP, tf, open,
ignore_invalid_symbol=true) : na
float hHYP = useHyper ? request.security(symHYP, tf, high,
ignore_invalid_symbol=true) : na
float lHYP = useHyper ? request.security(symHYP, tf, low,
ignore_invalid_symbol=true) : na
float cHYP = useHyper ? request.security(symHYP, tf, close,
ignore_invalid_symbol=true) : na

// ===== AGGREGATE DATA =====
float oAgg = nz0(oBIN) + nz0(oBYB) + nz0(oOKX) + nz0(oHYP)
float hAgg = nz0(hBIN) + nz0(hBYB) + nz0(hOKX) + nz0(hHYP)
float lAgg = nz0(lBIN) + nz0(lBYB) + nz0(lOKX) + nz0(lHYP)
float cAgg = nz0(cBIN) + nz0(cBYB) + nz0(cOKX) + nz0(cHYP)

// Проверка наличия данных
bool hasAgg = isValid(cBIN) or isValid(cBYB) or isValid(cOKX) or
isValid(cHYP)

// ===== FALLBACK TO LOCAL OI =====
string localSymbol = syminfo.tickerid + "_OI"
float oLocal = request.security(localSymbol, tf, open,
ignore_invalid_symbol=true)
float hLocal = request.security(localSymbol, tf, high,
ignore_invalid_symbol=true)
float lLocal = request.security(localSymbol, tf, low,
ignore_invalid_symbol=true)
float cLocal = request.security(localSymbol, tf, close,
ignore_invalid_symbol=true)

// Финальные значения OI
float o = hasAgg ? oAgg : oLocal
float h = hasAgg ? hAgg : hLocal

```

```

float l = hasAgg ? lAgg : lLocal
float c = hasAgg ? cAgg : cLocal

// ===== CALCULATIONS =====
// Дельта OI (абсолютное изменение)
float delta = c - nz0(c[1])

// Процентное изменение OI
float deltaPct = nz0(c[1]) != 0 ? (delta / nz0(c[1])) * 100 : 0

// Скользящая средняя для OI
float maOI = ta.sma(c, maLength)

// Скользящая средняя для дельты (сглаживание)
float deltaMA = ta.sma(math.abs(delta), volumeMA)

// Получаем цену для фильтрации
float priceClose = request.security(syminfo.tickerid, tf, close)
float priceOpen = request.security(syminfo.tickerid, tf, open)
bool priceUp = priceClose > priceOpen
bool priceDown = priceClose < priceOpen

// ===== SIGNAL GENERATION =====
// Значительный приток OI (большая положительная дельта)
bool bigOIIncrease = deltaPct > deltaThreshold and delta > deltaMA

// Значительный отток OI (большая отрицательная дельта)
bool bigOIDecrease = deltaPct < -deltaThreshold and math.abs(delta) > deltaMA

// Сигналы
bool longSignal = false
bool shortSignal = false

if showSignals
    // LONG: Большой приток OI + цена растет (если фильтр включен)
    longSignal := bigOIIncrease and (not filterByPrice or priceUp) and c >
maOI

    // SHORT: Большой отток OI + цена падает (если фильтр включен)
    shortSignal := bigOIDecrease and (not filterByPrice or priceDown) and c <
maOI

// ===== PLOTTING =====
// Цвет свечей OI
color candleColor = (c >= o) ? colorBullCandle : colorBearCandle

```

```

// Свечи OI
plotcandle(showCandles ? o : na,
           showCandles ? h : na,
           showCandles ? l : na,
           showCandles ? c : na,
           title="OI Candles",
           color=candleColor,
           wickcolor=candleColor,
           bordercolor=candleColor)

// Линия OI
plot(showCloseLine ? c : na,
     title="OI Close",
     color=colorOIline,
     linewidth=2)

// Скользящая средняя OI
plot(showMA ? maOI : na,
     title="OI MA",
     color=colorMA,
     linewidth=2,
     style=plot.style_line)

// Дельта OI (гистограмма)
color deltaColor = delta >= 0 ? colorDeltaPos : colorDeltaNeg
plot(showDelta ? delta : na,
     title="OI Delta",
     style=plot.style_columns,
     color=deltaColor)

// Нулевая линия
hline(0, "Zero Line", color=color.new(color.gray, 60),
     linestyle=hline.style_dashed)

// Сигналы входа
plotshape(showSignals and longSignal,
         title="LONG Signal",
         style=shape.triangleup,
         location=location.bottom,
         color=colorBuySignal,
         size=size.small,
         text="LONG")

plotshape(showSignals and shortSignal,

```

```

        title="SHORT Signal",
        style=shape.triangledown,
        location=location.top,
        color=colorSellSignal,
        size=size.small,
        text="SHORT")

// ===== INFO TABLE =====
var table infoTable = table.new(position.top_right, 2, 9,
                                bgcolor=color.new(#000000, 90),
                                frame_color=color.new(#424242, 30),
                                frame_width=2,
                                border_width=1,
                                border_color=color.new(#424242, 50))

if barstate.islast and not na(c)
    // Заголовок
    table.cell(infoTable, 0, 0, " Metric",
               text_color=color.new(color.white, 0),
               text_size=size.normal,
               bgcolor=color.new(#1976D2, 20))
    table.cell(infoTable, 1, 0, "Value",
               text_color=color.new(color.white, 0),
               text_size=size.normal,
               bgcolor=color.new(#1976D2, 20))

    // Текущий OI
    table.cell(infoTable, 0, 1, "Current OI",
               text_color=color.new(color.silver, 0),
               text_size=size.small)
    table.cell(infoTable, 1, 1, str.tostring(c, format.volume),
               text_color=color.new(color.white, 0),
               text_size=size.small)

    // Дельта абсолютная
    string deltaText = delta >= 0 ? "+" + str.tostring(delta, format.volume)
: str.tostring(delta, format.volume)
    color deltaTextColor = delta >= 0 ? #00E676 : #FF1744
    table.cell(infoTable, 0, 2, "Delta (abs)",
               text_color=color.new(color.silver, 0),
               text_size=size.small)
    table.cell(infoTable, 1, 2, deltaText,
               text_color=deltaTextColor,
               text_size=size.small)

```

```

// Дельта процентная
string pctText = deltaPct >= 0 ? "+" + str.toString(deltaPct, "#.##") +
"%" : str.toString(deltaPct, "#.##") + "%"
color pctColor = deltaPct >= 0 ? #00E676 : #FF1744
table.cell(infoTable, 0, 3, "Delta (%)",
            text_color=color.new(color.silver, 0),
            text_size=size.small)
table.cell(infoTable, 1, 3, pctText,
            text_color=pctColor,
            text_size=size.small)

// MA OI
table.cell(infoTable, 0, 4, "MA(" + str.toString(maLength) + ")",
            text_color=color.new(color.silver, 0),
            text_size=size.small)
table.cell(infoTable, 1, 4, str.toString(maOI, format.volume),
            text_color=color.new(#FF9800, 0),
            text_size=size.small)

// Тренд OI
string oiTrend = c > maOI ? "↑ РОСТ" : "↓ ПАДЕНИЕ"
color trendColor = c > maOI ? #00E676 : #FF1744
table.cell(infoTable, 0, 5, "OI Trend",
            text_color=color.new(color.silver, 0),
            text_size=size.small)
table.cell(infoTable, 1, 5, oiTrend,
            text_color=trendColor,
            text_size=size.small)

// Сигнал
string signalText = longSignal ? "● LONG" : shortSignal ? "● SHORT" :
"● HET"
color signalColor = longSignal ? #00E676 : shortSignal ? #FF1744 :
color.gray
table.cell(infoTable, 0, 6, "Signal",
            text_color=color.new(color.silver, 0),
            text_size=size.small)
table.cell(infoTable, 1, 6, signalText,
            text_color=signalColor,
            text_size=size.small)

// Активные биржи
string exchanges = ""
exchanges += useBinance ? "BIN " : ""
exchanges += useBybit ? "BYB " : ""

```

```

exchanges += useOKX ? "OKX " : ""
exchanges += useHyper ? "HYP" : ""
table.cell(infoTable, 0, 7, "Exchanges",
           text_color=color.new(color.silver, 0),
           text_size=size.small)
table.cell(infoTable, 1, 7, exchanges,
           text_color=color.new(#FFC107, 0),
           text_size=size.small)

// Таймфрейм
table.cell(infoTable, 0, 8, "Timeframe",
           text_color=color.new(color.silver, 0),
           text_size=size.small)
table.cell(infoTable, 1, 8, tf,
           text_color=color.new(color.white, 0),
           text_size=size.small)

// ===== ALERTS =====
alertcondition(longSignal, title="LONG Signal", message="🟢 LONG:
Значительный приток OI обнаружен на {{ticker}}")
alertcondition(shortSignal, title="SHORT Signal", message="🔴 SHORT:
Значительный отток OI обнаружен на {{ticker}}")

```