

回測—1120321 裕隆

The screenshot shows a trading strategy configuration window with the following sections:

- 腳本設定 (Script Settings):**
 - 腳本: 移動停利
 - 停利(點): 2
 - 停利回跌(點): 0.5
 - 停損(點): 0.7
- 執行商品類型 (Execution Commodity Type):**
 - 指定商品: 裕隆(2201)
- 執行頻率 (Execution Frequency):**
 - 頻率: 1分鐘 (原始值)
 - 逐筆洗價: 啟用
- 策略部位計算起點 (Strategy Position Calculation Start Point):**
 - 不指定
 - 資料讀取筆數: 100
- 交易帳號 (Trading Account):**
 - 已啟用
 - 證-模擬交易(內建)-01
 - 現股 / 債券
 - ROD / IOC
 - 只作多 / 只作空 / 無限制
 - 單邊交易費用: 0.2 %
 - 交易回報等待時間: 20 秒
- 策略部位 (Strategy Position):**
 - 部位: 依腳本計算
 - 計算起點: 不指定
 - Table with columns: 商品名稱, 部位, 策略均價. Content: 依腳本計算

{多單移動停利(點)設定停損點(如果不設定的話, 請把 loss_point 設定成 0), 以及停利點, 跟回跌點數
價格下跌到停損時出場價格上漲到停利點後啟動移動停利, 如果價格繼續上漲, 則繼續持有, 如果價格回檔超過回跌點數時, 則停利出場}

```
input: profit_point(10, "停利(點)");
```

```
input: profit_drawback_point(5, "停利回跌(點)");
```

```
input: loss_point(10, "停損(點)");
```

```
var: long_condition(false);           { 進場買進作多 }
```

{範例:均線穿越時買進 1 張以成交價為基礎, 設定固定停損以及移動停利}

```
if profit_point = 0 then raiseruntimeerror("請設定停利(點)");
```

```
if profit_drawback_point = 0 then raiseruntimeerror("請設定停利回跌(點)");
```

```
if profit_drawback_point > profit_point then raiseruntimeerror("停利(點)需大於停利回跌(點)");
```

```
long_condition = Average(Close, 5) cross over Average(Close, 20);
```

```
if Position = 0 and long_condition then begin
```

```
    SetPosition(1);           { 買進 1 張 }
```

```
end;
```

```
if Position = 1 and Filled = 1 then begin
```

```
    var: intrabarpersist max_profit_point(0);   { 啟動停利後最大獲利點 }
```

```
if loss_point > 0 and Close <= FilledAvgPrice - loss_point then begin
    { 停損 }
    SetPosition(0);
    max_profit_point = 0;
end else begin
{ 判斷是否要啟動停利 }
    if max_profit_point = 0 and Close >= FilledAvgPrice + profit_point then begin
        max_profit_point = Close;
    end;

    if max_profit_point <> 0 then begin
        if Close <= max_profit_point - profit_drawback_point then begin
            { 停利 }
            SetPosition(0);
            max_profit_point = 0;
        end else if Close > max_profit_point then begin
            { 移動最大獲利點 }
            max_profit_point = Close;
        end;
    end;
end;
end;
end;
```

```

//適合於 1 分鐘頻率
if barFreq <> "Min" or barInterval <> 1 then raiseRunTimeError
("本腳本只適合於 1 分鐘線");
variable: barcount(0),H15(0),L15(0);
if date <> date[1] then barcount=1 else barcount+=1;
if barcount=15 then
begin
H15=highest(high,15);
L15=lowest(low,15);

end;
condition1=false;

if close>H15 then
begin
condition1=true;
end;

```

{多單移動停利(點)設定停損點(如果不設定的話, 請把 loss_point 設定成 0), 以及停利點, 跟回跌點數
價格下跌到停損時出場
價格上漲到停利點後啟動移動停利, 如果價格繼續上漲, 則繼續持有, 如果價格回檔超過回跌點
數時, 則停利出場}

```

input: profit_point(10, "停利(點)");
input: profit_drawback_point(5, "停利回跌(點)");
input: loss_point(10, "停損(點)");

var: long_condition(false);          { 進場買進作多 }

```

{ 範例:均線穿越時買進 1 張
以成交價為基礎, 設定固定停損以及移動停利}

```

if profit_point = 0 then raiseruntimeerror("請設定停利(點)");
if profit_drawback_point = 0 then raiseruntimeerror("請設定停利回跌(點)");
if profit_drawback_point > profit_point then raiseruntimeerror("停利(點)需大於停利回跌(點)");

long_condition = Average(Close, 5) cross over Average(Close, 20);

```

```
if Position = 0 and barcount >= 16 and condition1
  then begin
    SetPosition(1);          { 買進 1 張 }
  end;
end;
```

```
if Position = 1 and Filled = 1 then begin
  var: intrabarpersist max_profit_point(0);  { 啟動停利後最大獲利點 }

  if loss_point > 0 and Close <= FilledAvgPrice - loss_point then begin
    { 停損 }
    SetPosition(0);
    max_profit_point = 0;

  end else begin

    { 判斷是否要啟動停利 }
    if max_profit_point = 0 and Close >= FilledAvgPrice + profit_point then begin
      max_profit_point = Close;
    end;

    if max_profit_point <> 0 then begin
      if Close <= max_profit_point - profit_drawback_point then begin
        { 停利 }
        SetPosition(0);
        max_profit_point = 0;
      end else if Close > max_profit_point then begin
        { 移動最大獲利點 }
        max_profit_point = Close;
      end;
    end;
  end;
end;

end;
```