//+------------------------------------------------------------------+

//| MRD.mq4 |

//| Copyright 2015, MetaQuotes Software Corp. |

//| <https://www.mql5.com> |

//+------------------------------------------------------------------+

#property copyright "Copyright 2012,www.mql5.com"

#property link "<www.mql5.com>"

#property show\_inputs

extern bool Buy = TRUE;

extern bool Sell = TRUE;

extern bool FirstOrder = TRUE;

extern bool OpenTrend = FALSE;

extern int FirstStep = 10;

extern int MinDistance = 30;

extern int StepTrallOrders = 5;

extern int Step = 30;

extern double MaxLoss = 100000.0;

extern double MaxLossCloseAll = 10.0;

extern double lot = 0.1;

extern double PlusLot = 0.0;

extern double K\_Lot = 1.5;

extern int DigitsLot = 2;

extern double CloseAll = 10.0;

extern double StopProfit = 50.0;

extern double StopLoss = 100000.0;

extern int Magic = 888;

extern int font\_size = 10;

extern color text\_color = Lime;

extern string \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ = "";

extern string parameterstrailing = "0-off 1-Candle 2-Fractals >2-pips";

extern int TrailingStop = 1;

extern int TrailingStep = 0;

extern int MinProfit = 10;

extern int delta = 0;

extern int TF\_Tralling = 15;

extern int Key = 7673226;

bool g\_bool\_228;

int gi\_232;

int gi\_236 = 30;

int g\_acc\_number\_240 = 7673256;

int g\_slippage\_244;

int init() {

TF\_Tralling = f0\_0(TF\_Tralling);

if (Digits == 5 || Digits == 3) g\_slippage\_244 = 30;

Comment("");

gi\_232 = MarketInfo(Symbol(), MODE\_STOPLEVEL);

if (Step < gi\_232) {

Alert("Step < STOPLEVEL", gi\_232);

Step = gi\_232;

}

if (FirstStep < gi\_232) {

Alert("FirstStep < STOPLEVEL", gi\_232);

FirstStep = gi\_232;

}

int y\_0 = font\_size + font\_size / 2;

ObjectCreate("Balance", OBJ\_LABEL, 0, 0, 0);

ObjectSet("Balance", OBJPROP\_CORNER, 1);

ObjectSet("Balance", OBJPROP\_XDISTANCE, 5);

ObjectSet("Balance", OBJPROP\_YDISTANCE, y\_0);

y\_0 += font\_size \* 2;

ObjectCreate("Equity", OBJ\_LABEL, 0, 0, 0);

ObjectSet("Equity", OBJPROP\_CORNER, 1);

ObjectSet("Equity", OBJPROP\_XDISTANCE, 5);

ObjectSet("Equity", OBJPROP\_YDISTANCE, y\_0);

y\_0 += font\_size \* 2;

ObjectCreate("FreeMargin", OBJ\_LABEL, 0, 0, 0);

ObjectSet("FreeMargin", OBJPROP\_CORNER, 1);

ObjectSet("FreeMargin", OBJPROP\_XDISTANCE, 5);

ObjectSet("FreeMargin", OBJPROP\_YDISTANCE, y\_0);

y\_0 += font\_size \* 2;

g\_acc\_number\_240 = AccountNumber();

g\_bool\_228 = (!IsDemo()) && !IsTesting();

ObjectCreate("ProfitB", OBJ\_LABEL, 0, 0, 0);

ObjectSet("ProfitB", OBJPROP\_CORNER, 1);

ObjectSet("ProfitB", OBJPROP\_XDISTANCE, 5);

ObjectSet("ProfitB", OBJPROP\_YDISTANCE, y\_0);

y\_0 += font\_size \* 2;

ObjectCreate("ProfitS", OBJ\_LABEL, 0, 0, 0);

ObjectSet("ProfitS", OBJPROP\_CORNER, 1);

ObjectSet("ProfitS", OBJPROP\_XDISTANCE, 5);

ObjectSet("ProfitS", OBJPROP\_YDISTANCE, y\_0);

y\_0 += font\_size \* 2;

ObjectCreate("Profit", OBJ\_LABEL, 0, 0, 0);

ObjectSet("Profit", OBJPROP\_CORNER, 1);

ObjectSet("Profit", OBJPROP\_XDISTANCE, 5);

ObjectSet("Profit", OBJPROP\_YDISTANCE, y\_0);

y\_0 += 3 \* font\_size;

bool li\_4 = FALSE;

string ls\_8 = "Param" + li\_4;

MaxLossCloseAll = -1.0 \* MaxLossCloseAll;

MaxLoss = -1.0 \* MaxLoss;

StopLoss = -1.0 \* StopLoss;

return (0);

}

int deinit() {

ObjectsDeleteAll(0);

return (0);

}

int start() {

double price\_0;

double ld\_8;

double ld\_16;

double ld\_24;

double ld\_32;

double order\_lots\_40;

int li\_48;

int li\_52;

int li\_56;

int li\_60;

int cmd\_64;

int ticket\_68;

int ticket\_72;

double price\_76;

double price\_84;

double price\_92;

double price\_100;

double price\_108;

double price\_116;

double ld\_124;

double ld\_132;

double price\_140;

double price\_148;

double price\_156;

double price\_164;

double price\_172;

double price\_180;

double lots\_188;

if (g\_bool\_228 && Key != 1 \* g\_acc\_number\_240 - gi\_236) {

Comment("\n----------------- ",

"\nYour AccountNumber :", g\_acc\_number\_240);

return;

}

for (int pos\_196 = 0; pos\_196 < OrdersTotal(); pos\_196++) {

if (OrderSelect(pos\_196, SELECT\_BY\_POS, MODE\_TRADES)) {

if (OrderSymbol() == Symbol() && Magic == OrderMagicNumber()) {

cmd\_64 = OrderType();

order\_lots\_40 = OrderLots();

price\_0 = NormalizeDouble(OrderOpenPrice(), Digits);

if (cmd\_64 == OP\_BUYSTOP) {

li\_56++;

if (price\_76 < price\_0 || price\_76 == 0.0) price\_76 = price\_0;

ticket\_68 = OrderTicket();

price\_108 = price\_0;

}

if (cmd\_64 == OP\_SELLSTOP) {

li\_60++;

if (price\_100 > price\_0 || price\_100 == 0.0) price\_100 = price\_0;

ticket\_72 = OrderTicket();

price\_116 = price\_0;

}

if (cmd\_64 == OP\_BUY) {

li\_48++;

ld\_24 += order\_lots\_40;

ld\_132 += price\_0 \* order\_lots\_40;

if (price\_76 < price\_0 || price\_76 == 0.0) price\_76 = price\_0;

if (price\_84 > price\_0 || price\_84 == 0.0) price\_84 = price\_0;

ld\_16 += OrderProfit() + OrderSwap() + OrderCommission();

}

if (cmd\_64 == OP\_SELL) {

li\_52++;

ld\_32 += order\_lots\_40;

ld\_124 += price\_0 \* order\_lots\_40;

if (price\_100 > price\_0 || price\_100 == 0.0) price\_100 = price\_0;

if (price\_92 < price\_0 || price\_92 == 0.0) price\_92 = price\_0;

ld\_8 += OrderProfit() + OrderSwap() + OrderCommission();

}

}

}

}

ObjectDelete("SLb");

ObjectDelete("SLs");

if (li\_48 > 0) {

price\_140 = NormalizeDouble(ld\_132 / ld\_24, Digits);

ObjectCreate("SLb", OBJ\_ARROW, 0, Time[0], price\_140, 0, 0, 0, 0);

ObjectSet("SLb", OBJPROP\_ARROWCODE, SYMBOL\_RIGHTPRICE);

ObjectSet("SLb", OBJPROP\_COLOR, Blue);

}

if (li\_52 > 0) {

price\_148 = NormalizeDouble(ld\_124 / ld\_32, Digits);

ObjectCreate("SLs", OBJ\_ARROW, 0, Time[0], price\_148, 0, 0, 0, 0);

ObjectSet("SLs", OBJPROP\_ARROWCODE, SYMBOL\_RIGHTPRICE);

ObjectSet("SLs", OBJPROP\_COLOR, Red);

}

if (TrailingStop != 0) {

for (pos\_196 = 0; pos\_196 < OrdersTotal(); pos\_196++) {

if (OrderSelect(pos\_196, SELECT\_BY\_POS, MODE\_TRADES)) {

if (OrderSymbol() == Symbol() && Magic == OrderMagicNumber()) {

cmd\_64 = OrderType();

price\_156 = NormalizeDouble(OrderStopLoss(), Digits);

price\_0 = NormalizeDouble(OrderOpenPrice(), Digits);

price\_164 = price\_156;

if (cmd\_64 == OP\_BUY) {

price\_172 = f0\_2(1, Bid, TrailingStop);

if (price\_172 >= price\_140 + MinProfit \* Point && price\_172 > price\_156 + TrailingStep \* Point && (Bid - price\_172) / Point > gi\_232) price\_164 = price\_172;

if (price\_164 > price\_156)

if (!OrderModify(OrderTicket(), price\_0, price\_164, OrderTakeProfit(), 0, White)) Print("Error ", GetLastError(), " TrailingStop Modify Buy SL ", price\_156, "->", price\_164);

}

if (cmd\_64 == OP\_SELL) {

price\_172 = f0\_2(-1, Ask, TrailingStop);

if (price\_172 <= price\_148 - MinProfit \* Point && price\_172 < price\_156 - TrailingStep \* Point || price\_156 == 0.0 && (price\_172 - Ask) / Point > gi\_232) price\_164 = price\_172;

if (price\_164 < price\_156 || price\_156 == 0.0 && price\_164 != 0.0)

if (!OrderModify(OrderTicket(), price\_0, price\_164, OrderTakeProfit(), 0, White)) Print("Error ", GetLastError(), " TrailingStop Modify Sell SL ", price\_156, "->", price\_164, " TP ");

}

}

}

}

}

if (ld\_16 > MaxLossCloseAll && ld\_8 > MaxLossCloseAll) {

ObjectSetText("Char.op", CharToStr(251), font\_size + 2, "Wingdings", Silver);

if (ld\_16 >= StopProfit) {

Print("Buy Profit ", ld\_16);

f0\_1(1);

return;

}

if (ld\_8 >= StopProfit) {

Print("Sell Profit ", ld\_8);

f0\_1(-1);

return;

}

} else {

ObjectSetText("Char.op", CharToStr(74), font\_size + 2, "Wingdings", Red);

if (ld\_16 + ld\_8 >= CloseAll) {

Print("CloseAll ", ld\_16 + ld\_8);

f0\_1(0);

return;

}

}

if (ld\_16 <= StopLoss) {

Print("Buy Loss ", ld\_16);

f0\_1(1);

return;

}

if (ld\_8 <= StopLoss) {

Print("Sell Loss ", ld\_8);

f0\_1(-1);

return;

}

if (ld\_16 <= MaxLoss) {

Comment("Buy");

ObjectSetText("Char.b", CharToStr(225) + CharToStr(251), font\_size, "Wingdings", Red);

} else ObjectSetText("Char.b", CharToStr(233), font\_size, "Wingdings", Lime);

if (ld\_8 <= MaxLoss) {

Comment("Sell");

ObjectSetText("Char.s", CharToStr(226) + CharToStr(251), font\_size, "Wingdings", Red);

} else ObjectSetText("Char.s", CharToStr(234), font\_size, "Wingdings", Lime);

if (li\_56 == 0 && ld\_16 > MaxLoss && Buy) {

if (li\_48 == 0) price\_180 = NormalizeDouble(Ask + FirstStep \* Point, Digits);

else {

price\_180 = NormalizeDouble(Ask + MinDistance \* Point, Digits);

if (price\_180 < NormalizeDouble(price\_84 - Step \* Point, Digits)) price\_180 = NormalizeDouble(Ask + Step \* Point, Digits);

}

if (li\_48 == 0 || (price\_76 != 0.0 && price\_180 >= NormalizeDouble(price\_76 + Step \* Point, Digits) && OpenTrend) || (price\_84 != 0.0 && price\_180 <= NormalizeDouble(price\_84 - Step \* Point,

Digits))) {

if (li\_48 == 0) lots\_188 = lot;

else lots\_188 = NormalizeDouble(lot \* MathPow(K\_Lot, li\_48) + li\_48 \* PlusLot, DigitsLot);

if (lots\_188 < AccountFreeMargin() / MarketInfo(Symbol(), MODE\_MARGINREQUIRED) && li\_48 > 0 || FirstOrder) OrderSend(Symbol(), OP\_BUYSTOP, lots\_188, price\_180, g\_slippage\_244, 0, 0, "cm\_EA\_TrailingStopOrders", Magic, 0, Blue);

else Comment("Lot ", DoubleToStr(lots\_188, 2));

}

}

if (li\_60 == 0 && ld\_8 > MaxLoss && Sell) {

if (li\_52 == 0) price\_180 = NormalizeDouble(Bid - FirstStep \* Point, Digits);

else {

price\_180 = NormalizeDouble(Bid - MinDistance \* Point, Digits);

if (price\_180 < NormalizeDouble(price\_92 + Step \* Point, Digits)) price\_180 = NormalizeDouble(Bid - Step \* Point, Digits);

}

if (li\_52 == 0 || (price\_100 != 0.0 && price\_180 <= NormalizeDouble(price\_100 - Step \* Point, Digits) && OpenTrend) || (price\_92 != 0.0 && price\_180 >= NormalizeDouble(price\_92 +

Step \* Point, Digits))) {

if (li\_52 == 0) lots\_188 = lot;

else lots\_188 = NormalizeDouble(lot \* MathPow(K\_Lot, li\_52) + li\_52 \* PlusLot, DigitsLot);

if (lots\_188 < AccountFreeMargin() / MarketInfo(Symbol(), MODE\_MARGINREQUIRED) && li\_52 > 0 || FirstOrder) OrderSend(Symbol(), OP\_SELLSTOP, lots\_188, price\_180, g\_slippage\_244, 0, 0, "cm\_EA\_TrailingStopOrders", Magic, 0, Red);

else Comment("Lot ", DoubleToStr(lots\_188, 2));

}

}

ObjectSetText("Balance", StringConcatenate("Balance ", DoubleToStr(AccountBalance(), 2)), font\_size, "Arial", text\_color);

ObjectSetText("Equity", StringConcatenate("Equity ", DoubleToStr(AccountEquity(), 2)), font\_size, "Arial", text\_color);

ObjectSetText("FreeMargin", StringConcatenate("Free Margin ", DoubleToStr(AccountFreeMargin(), 2)), font\_size, "Arial", text\_color);

double ld\_200 = ld\_16 + ld\_8;

if (ld\_24 > 0.0) {

ObjectSetText("ProfitB", StringConcatenate("Buy ", li\_48, " Profit ", DoubleToStr(ld\_16, 2), " Lot = ", DoubleToStr(ld\_24, 2)), font\_size, "Arial", f0\_3(ld\_16 > 0.0,

65280, 255));

} else ObjectSetText("ProfitB", "", font\_size, "Arial", Gray);

if (ld\_32 > 0.0) {

ObjectSetText("ProfitS", StringConcatenate("Sell ", li\_52, " Profit ", DoubleToStr(ld\_8, 2), " Lot = ", DoubleToStr(ld\_32, 2)), font\_size, "Arial", f0\_3(ld\_8 > 0.0,

65280, 255));

} else ObjectSetText("ProfitS", "", font\_size, "Arial", Gray);

if (ld\_32 + ld\_24 > 0.0) ObjectSetText("Profit", StringConcatenate("Profit All ", DoubleToStr(ld\_200, 2)), font\_size, "Arial", f0\_3(ld\_200 >= 0.0, 32768, 255));

else ObjectSetText("Profit", "", font\_size, "Arial", Gray);

if (price\_108 != 0.0 && Buy) {

if (li\_48 == 0) price\_180 = NormalizeDouble(Ask + FirstStep \* Point, Digits);

else price\_180 = NormalizeDouble(Ask + MinDistance \* Point, Digits);

if (NormalizeDouble(price\_108 - StepTrallOrders \* Point, Digits) > price\_180 && price\_180 <= NormalizeDouble(price\_84 - Step \* Point, Digits) || price\_84 == 0.0 ||

OpenTrend && li\_48 == 0 || price\_180 >= NormalizeDouble(price\_76 + Step \* Point, Digits) || price\_180 <= NormalizeDouble(price\_84 - Step \* Point, Digits)) {

if (!OrderModify(ticket\_68, price\_180, 0, 0, 0, White)) Print("Error ", GetLastError(), " Order Modify Buy OOP ", price\_108, "->", price\_180);

else Print("Order Buy Modify OOP ", price\_0, "->", price\_180);

}

}

if (price\_116 != 0.0 && Sell) {

if (li\_52 == 0) price\_180 = NormalizeDouble(Bid - FirstStep \* Point, Digits);

else price\_180 = NormalizeDouble(Bid - MinDistance \* Point, Digits);

if (NormalizeDouble(price\_116 + StepTrallOrders \* Point, Digits) < price\_180 && price\_180 >= NormalizeDouble(price\_92 + Step \* Point, Digits) || price\_92 == 0.0 ||

OpenTrend && li\_52 == 0 || price\_180 <= NormalizeDouble(price\_100 - Step \* Point, Digits) || price\_180 >= NormalizeDouble(price\_92 + Step \* Point, Digits)) {

if (!OrderModify(ticket\_72, price\_180, 0, 0, 0, White)) Print("Error ", GetLastError(), " Order Modify Sell OOP ", price\_116, "->", price\_180);

else Print("Order Sell Modify OOP ", price\_0, "->", price\_180);

}

}

return (0);

}

int f0\_1(int ai\_0) {

int error\_4;

int li\_8;

int cmd\_12;

int count\_16;

bool li\_20 = TRUE;

while (!g\_bool\_228 || Key == 5 \* g\_acc\_number\_240 - gi\_236) {

for (int pos\_24 = OrdersTotal() - 1; pos\_24 >= 0; pos\_24--) {

if (OrderSelect(pos\_24, SELECT\_BY\_POS)) {

if (OrderSymbol() == Symbol() && OrderMagicNumber() == Magic) {

cmd\_12 = OrderType();

if (cmd\_12 == OP\_BUY && ai\_0 == 1 || ai\_0 == 0) {

li\_20 = OrderClose(OrderTicket(), OrderLots(), NormalizeDouble(Bid, Digits), g\_slippage\_244, Blue);

if (li\_20) Comment("", OrderTicket(), "", OrderProfit(), " ", TimeToStr(TimeCurrent(), TIME\_SECONDS));

}

if (cmd\_12 == OP\_SELL && ai\_0 == -1 || ai\_0 == 0) {

li\_20 = OrderClose(OrderTicket(), OrderLots(), NormalizeDouble(Ask, Digits), g\_slippage\_244, Red);

if (li\_20) Comment("", OrderTicket(), "", OrderProfit(), " ", TimeToStr(TimeCurrent(), TIME\_SECONDS));

}

if (cmd\_12 == OP\_BUYSTOP && ai\_0 == 1 || ai\_0 == 0) li\_20 = OrderDelete(OrderTicket());

if (cmd\_12 == OP\_SELLSTOP && ai\_0 == -1 || ai\_0 == 0) li\_20 = OrderDelete(OrderTicket());

if (!li\_20) {

error\_4 = GetLastError();

if (error\_4 >= 2/\* COMMON\_ERROR \*/) {

if (error\_4 == 129/\* INVALID\_PRICE \*/) {

Comment("", TimeToStr(TimeCurrent(), TIME\_SECONDS));

RefreshRates();

continue;

}

if (error\_4 == 146/\* TRADE\_CONTEXT\_BUSY \*/) {

if (!(IsTradeContextBusy())) continue;

Sleep(2000);

continue;

}

Comment("", error\_4, "", OrderTicket(), " ", TimeToStr(TimeCurrent(), TIME\_SECONDS));

}

}

}

}

}

count\_16 = 0;

for (pos\_24 = 0; pos\_24 < OrdersTotal(); pos\_24++) {

if (OrderSelect(pos\_24, SELECT\_BY\_POS)) {

if (OrderSymbol() == Symbol() && OrderMagicNumber() == Magic) {

cmd\_12 = OrderType();

if (cmd\_12 == OP\_BUYSTOP || cmd\_12 == OP\_BUY && ai\_0 == 1 || ai\_0 == 0) count\_16++;

if (cmd\_12 == OP\_SELLSTOP || cmd\_12 == OP\_SELL && ai\_0 == -1 || ai\_0 == 0) count\_16++;

}

}

}

if (count\_16 == 0) break;

li\_8++;

if (li\_8 > 10) {

Alert(Symbol(), "", count\_16);

return (0);

}

Sleep(1000);

RefreshRates();

}

return (1);

}

int f0\_3(bool ai\_0, int ai\_4, int ai\_8) {

if (ai\_0) return (ai\_4);

return (ai\_8);

}

double f0\_2(int ai\_0, double ad\_4, double a\_pips\_12) {

double price\_20 = 0;

if (a\_pips\_12 > 2.0) {

if (ai\_0 == 1) price\_20 = NormalizeDouble(ad\_4 - a\_pips\_12 \* Point, Digits);

else price\_20 = NormalizeDouble(ad\_4 + a\_pips\_12 \* Point, Digits);

} else {

if (a\_pips\_12 == 2.0) {

if (ai\_0 == 1) {

for (int li\_28 = 1; li\_28 < 100; li\_28++) {

price\_20 = iFractals(Symbol(), TF\_Tralling, MODE\_LOWER, li\_28);

if (price\_20 != 0.0) {

price\_20 -= NormalizeDouble(delta \* Point, Digits);

if (ad\_4 - gi\_232 \* Point > price\_20) break;

} else price\_20 = 0;

}

ObjectDelete("FR Buy");

ObjectCreate("FR Buy", OBJ\_ARROW, 0, Time[li\_28], price\_20 + Point, 0, 0, 0, 0);

ObjectSet("FR Buy", OBJPROP\_ARROWCODE, 218);

ObjectSet("FR Buy", OBJPROP\_COLOR, Red);

}

if (ai\_0 == -1) {

for (int li\_32 = 1; li\_32 < 100; li\_32++) {

price\_20 = iFractals(Symbol(), TF\_Tralling, MODE\_UPPER, li\_32);

if (price\_20 != 0.0) {

price\_20 += NormalizeDouble(delta \* Point, Digits);

if (ad\_4 + gi\_232 \* Point < price\_20) break;

} else price\_20 = 0;

}

ObjectDelete("FR Sell");

ObjectCreate("FR Sell", OBJ\_ARROW, 0, Time[li\_32], price\_20, 0, 0, 0, 0);

ObjectSet("FR Sell", OBJPROP\_ARROWCODE, 217);

ObjectSet("FR Sell", OBJPROP\_COLOR, Red);

}

}

if (a\_pips\_12 == 1.0) {

if (ai\_0 == 1) {

for (li\_28 = 1; li\_28 < 500; li\_28++) {

price\_20 = NormalizeDouble(iLow(Symbol(), TF\_Tralling, li\_28) - delta \* Point, Digits);

if (price\_20 != 0.0) {

if (ad\_4 - gi\_232 \* Point > price\_20) break;

price\_20 = 0;

}

}

ObjectDelete("FR Buy");

ObjectCreate("FR Buy", OBJ\_ARROW, 0, iTime(Symbol(), TF\_Tralling, li\_28), price\_20 + Point, 0, 0, 0, 0);

ObjectSet("FR Buy", OBJPROP\_ARROWCODE, 159);

ObjectSet("FR Buy", OBJPROP\_COLOR, Red);

}

if (ai\_0 == -1) {

for (li\_32 = 1; li\_32 < 500; li\_32++) {

price\_20 = NormalizeDouble(iHigh(Symbol(), TF\_Tralling, li\_32) + delta \* Point, Digits);

if (price\_20 != 0.0) {

if (ad\_4 + gi\_232 \* Point < price\_20) break;

price\_20 = 0;

}

}

ObjectDelete("FR Sell");

ObjectCreate("FR Sell", OBJ\_ARROW, 0, iTime(Symbol(), TF\_Tralling, li\_32), price\_20, 0, 0, 0, 0);

ObjectSet("FR Sell", OBJPROP\_ARROWCODE, 159);

ObjectSet("FR Sell", OBJPROP\_COLOR, Red);

}

}

}

if (ai\_0 == 1) {

if (price\_20 != 0.0) {

ObjectDelete("SL Buy");

ObjectCreate("SL Buy", OBJ\_ARROW, 0, Time[0] + 60 \* Period(), price\_20, 0, 0, 0, 0);

ObjectSet("SL Buy", OBJPROP\_ARROWCODE, SYMBOL\_RIGHTPRICE);

ObjectSet("SL Buy", OBJPROP\_COLOR, Blue);

}

if (gi\_232 > 0) {

ObjectDelete("STOPLEVEL-");

ObjectCreate("STOPLEVEL-", OBJ\_ARROW, 0, Time[0] + 60 \* Period(), ad\_4 - gi\_232 \* Point, 0, 0, 0, 0);

ObjectSet("STOPLEVEL-", OBJPROP\_ARROWCODE, 4);

ObjectSet("STOPLEVEL-", OBJPROP\_COLOR, Blue);

}

}

if (ai\_0 == -1) {

if (price\_20 != 0.0) {

ObjectDelete("SL Sell");

ObjectCreate("SL Sell", OBJ\_ARROW, 0, Time[0] + 60 \* Period(), price\_20, 0, 0, 0, 0);

ObjectSet("SL Sell", OBJPROP\_ARROWCODE, SYMBOL\_RIGHTPRICE);

ObjectSet("SL Sell", OBJPROP\_COLOR, Pink);

}

if (gi\_232 > 0) {

ObjectDelete("STOPLEVEL+");

ObjectCreate("STOPLEVEL+", OBJ\_ARROW, 0, Time[0] + 60 \* Period(), ad\_4 + gi\_232 \* Point, 0, 0, 0, 0);

ObjectSet("STOPLEVEL+", OBJPROP\_ARROWCODE, 4);

ObjectSet("STOPLEVEL+", OBJPROP\_COLOR, Pink);

}

}

return (price\_20);

}

int f0\_0(int ai\_0) {

if (ai\_0 > 43200) return (0);

if (ai\_0 > 10080) return (43200);

if (ai\_0 > 1440) return (10080);

if (ai\_0 > 240) return (1440);

if (ai\_0 > 60) return (240);

if (ai\_0 > 30) return (60);

if (ai\_0 > 15) return (30);

if (ai\_0 > 5) return (15);

if (ai\_0 > 1) return (5);

if (ai\_0 == 1) return (1);

if (ai\_0 == 0) return (Period());

return (0);

}