**ซอสโค้ด การดำเนินการตามขอบเขตที่กำหนดผ่านตัวชี้วัด CCI บนรูปแบบการบริหารจัดการเงิน**

ในส่วนนี้จะเป็นส่วนของซอสโค้ดของโปรแกรมที่ใช้ในโปรแกรม Metatrader 4 ของกลุ่ม การดำเนินการตามขอบเขตที่กำหนดผ่านตัวชี้วัด CCI บนรูปแบบการบริหารจัดการเงิน

#property copyright "KZM Type C Version.011 , The project start at 8/02/2017 Software Corp."

#property link "https://www.facebook.com/groups/239641229810857/?fref=ts"

enum MinZoneOpen{

minopen0 = 25, //25 %

minopen1 = 20, //20 %

minopen2 = 15, //15 %

minopen3 = 10, //10 %

minopen4 = 5, //5 %

minopen5 = 0, //0 %

};

enum MaxZoneOpen{

maxopen0 = 50, //50 %

maxopen1 = 55, //55 %

maxopen2 = 60, //65 %

maxopen3 = 65, //65 %

maxopen4 = 70, //70 %

maxopen5 = 75, //75 %

maxopen6 = 80, //80 %

maxopen7 = 85, //95 %

maxopen8 = 90, //90 %

maxopen9 = 95, //95 %

maxopen10 = 100, //100 %

};

enum Zone\_Time{

Zone\_Time0 = 1, //1 Minute

Zone\_Time1 = 5, //5 Minute

Zone\_Time2 = 15, //15 Minute

Zone\_Time3 = 30, //30 Minute

Zone\_Time4 = 60, //1 Hour

Zone\_Time5 = 240, //4 Hour

Zone\_Time6 = 1440, //1 Day

Zone\_Time7 = 10080, //1 Weekly

Zone\_Time8 = 43200, //1 Monthly

};

input string \_\_\_\_\_\_\_\_KZM\_Order\_Set\_1 = "Order Setting.";

input int MagicExpert = 8025;

input bool UseZoneClose = TRUE;

input string CommentOrder = "FOREX KZM";

input double Lot = 0.01;

input bool AutoMaticLot = FALSE;

input bool CommentEA = TRUE;

input int MaxOrders = 20;

input Zone\_Time ZoneTime = Zone\_Time4;

input Zone\_Time IndicatorTime = Zone\_Time2;

input string \_\_\_\_\_\_\_\_Range\_Percent\_MaxMin\_Set\_2 = "Percent Of Zone Order.";

input bool UseCandHighPercent = TRUE;

input MaxZoneOpen MaximumZone = maxopen0;//UseMaximumZone;

input bool UseCandLowPercent = TRUE;

input MinZoneOpen MinimumZone = minopen0;//UseMinimumZone;

input int RangeCandlestick = 0;

int RangeCandlesticks = RangeCandlestick;

input bool UseLowZone = TRUE;

input string \_\_\_\_\_\_\_\_Manager\_Profile\_Set\_3 = "Management Accountant.";

input int Slippage = 1;

input string \_\_\_\_\_\_\_\_Min\_Deposit\_4 = "Good capital for deposit";

double GridPerPipW;

double GridPerPip;

input double PipDepositTest = 0.0;

double PipDeposit = PipDepositTest;

double Ori;

double cm01,cmi02;

int maxsend;

double useGrid;

double o1;

double o2;

double llo2;

double save;

double maxh, minl;

void OnInit() {

if (RangeCandlesticks == 0) {

RangeCandlesticks = iBars(NULL,ZoneTime);

}

}

void OnDeinit(const int reason)

{

ObjectDelete("nameofea");

ObjectDelete("zonemax");

ObjectDelete("nameofaccount");

ObjectDelete("accmoney");

ObjectDelete("accshowtime");

ObjectDelete("TradingMaxZone");

ObjectDelete("TradingLowZone");

ObjectDelete("zonemin");

ObjectDelete("fordeposit");

ObjectDelete("forpip");

ObjectDelete("forshowgrid");

ObjectDelete("forshoworder");

ObjectDelete("forlot");

ObjectDelete("forneeddeposit");

}

int maxopen(int ai\_0 = -1) {

int n = 0;

for (int li = OrdersTotal() - 1; li >= 0 && OrderSelect(li, SELECT\_BY\_POS); li--)

if (OrderSymbol() == Symbol() && OrderMagicNumber() == MagicExpert && ai\_0 == -1) {

n++; } else

if (ai\_0 == OrderType() && OrderCloseTime() == 0) n++;

return (n);

}

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

//| Expert tick function |

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

void OnTick()

{

int Cand = RangeCandlesticks-1;

double pj[500],pm[500],h1[500],minh,maxl,Zq,lgm,mizone;

int x,j;

double z = iHigh(Symbol(),NULL,0);

double y = iLow(Symbol(),NULL,0);

Ori = iClose(Symbol(),NULL,0);

o1 = Ask;

o2 = Bid;

for(x=0;x <= Cand; x++) {

z = iHigh(Symbol(),NULL,x);

y = iLow(Symbol(),NULL,x);

pj[x] = z;

pm[x] = y;

//Print("Bid=",z," of round=",x," in aray=",pj[x]);

//Print("Bid=",y," of round=",x," in aray=",pm[x]);

}

minh = pj[0];

minl = pm[0];

for(j=0 ;j <= Cand; j++) {

if (UseCandHighPercent) {

if(minh > pj[j]) {

minh = pj[j];

} else

if (maxh < pj[j]) {

maxh = pj[j];

}

//Print("Max is:",maxh,"Min is:",minh);

}

if (UseCandLowPercent) {

if(minl > pm[j]) {

minl = pm[j];

} else

if (maxl < pm[j]) {

maxl = pm[j];

}

//Print("Max is:",maxl,"Min is:",minl);

}

}

////////////////////////////////////////////////Zone trading./////////////////////

double cm;

Zq = (maxh - minl);

if (Digits == 3 || Digits == 5) {

Zq = 10000.0 \* Zq;//pip

} else {

Zq = 1000 \* Zq; //pip

}

//Print("Zone all is:",Zq," Pips");

cm = minl + (Zq / 10000); // cm = maxh

//Print("Min + Pips:",cm);

lgm = ((Zq / 100) \* MaximumZone);

//Print("Zone Max is:",lgm," Pips");

if (Digits == 3 || Digits == 5) {

cm01 = minl + (lgm / 10000);///////Zone Maximum.

//Print("Zone Max is:",cm01);

}

mizone = ((Zq / 100) \* MinimumZone);

//Print("Zone Min is:",mizone," Pips");

if (Digits == 3 || Digits == 5) {

cmi02 = minl + (mizone / 10000);//////Zone Minimum.

//Print("Zone Min is:",cmi02);

}

GridPerPip = (cm01 - cmi02) / MaxOrders;

useGrid = GridPerPip;

if (Digits == 3 || Digits == 5) {

GridPerPipW = (GridPerPip / Point) / 10.0;

}

//////////////////////////////////////////////////////////Zone End.

maxsend = maxopen();

//////////////////////////////////////////////////////////pip

double Mlot = Lot\*10;

if (PipDeposit == 0.0) {

PipDeposit = (maxh - minl)/Point/10;

}

/////////////////////////////////////////////////////////start Cal

double llo;

for(int h=0 ; h <= MaxOrders; h++) {

h1[h] = ((PipDeposit - GridPerPipW \* h) \* Mlot) / MarketInfo( Symbol(), MODE\_TICKVALUE);

llo += h1[h];

}

llo2 = MathAbs(llo);

/////////////////////////////////////////////////////////end cal

if (!UseLowZone) {

int pop = 0;

} else

if (UseLowZone)

{

pop = 1;

}

if (AccountBalance() >= llo2 && cmi02 > 0.0 && cm01 > 0.0 && Ori >= cmi02 \* pop && Ori <= cm01 && maxsend < MaxOrders) {

if(iCCI(Symbol(),IndicatorTime,36,PRICE\_CLOSE,0 >= 100) && iCCI(Symbol(),IndicatorTime,36,PRICE\_CLOSE,1 < 100)) {

if (maxsend == 0) {

if(OrderSend( Symbol(), OP\_BUY, callotsize(), Ask, Slippage \* 10, 0, 0, CommentOrder + ":" + OrdersTotal(), MagicExpert, 0, Blue ))

maxsend++;

Print("KZM BUY ORDER!" + OrdersTotal());

} else if (OrderMagicNumber() == MagicExpert && maxsend > 0) {

Grid();

}

}

}

if (maxsend > 0) {

if (UseZoneClose && Ori > cm01) {

ExitClose();

} else if (!UseZoneClose) {

ExitClose();

}

}

if (CommentEA) {

showmanycomment();

}

}

void ExitClose() {

int res = 0;

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

if(OrderType()==OP\_BUY) {

if (iCCI(Symbol(),IndicatorTime,36,PRICE\_CLOSE,0 <= -100) && iCCI(Symbol(),IndicatorTime,36,PRICE\_CLOSE,1 > -100) && OrderProfit() + OrderCommission() + OrderSwap() > 0.0) {

res = OrderClose(OrderTicket(), OrderLots(), NormalizeDouble(Bid,Digits), Slippage \* 10, Blue);

Print("Order Closed buy!"); } }

}

if (save == 0) {

save = AccountBalance() / 2;

} else

if (save > 0 && AccountBalance() < save) {

res = OrderClose(OrderTicket(), OrderLots(), NormalizeDouble(Bid,Digits), Slippage \* 10, Blue);

Print("Order Closed buy!");

}

}

void buy() {

int ticket2;

ticket2 = OrderSend( Symbol(), OP\_BUY, callotsize(), Ask, Slippage \* 10, 0, 0, CommentOrder + ":" + OrdersTotal(), MagicExpert, 0, Blue );

if (ticket2 > 0){

maxsend++;

Print("KZM BUY ORDER!" + OrdersTotal());

} else {

Print("ERROR SEND ORDER!!");

}

}

void Grid() {

if (OrderSymbol() == Symbol() && OrderComment() == CommentOrder + ":0") {

if (o1 > OrderOpenPrice()) {

for(int g = 1 ; g <= MaxOrders-1; g++) {

if (maxsend == g && o1 > OrderOpenPrice() + useGrid \* g) {

buy();

}

}

} else

if (o2 < OrderOpenPrice()) {

for(int a = 1 ; a <= MaxOrders-1; a++) {

if (maxsend == a && o2 < OrderOpenPrice() - useGrid \* a) {

buy();

}

}

}

}

}

double callotsize() {

double callot;

if (AutoMaticLot) {

callot = (AccountBalance() / llo2) \* Lot;

} else {

callot = Lot;

}

return (callot);

}

void showmanycomment() {

string nameea = "nameofea", showhzone = "zonemax", accountname = "nameofaccount", accbalace = "accmoney", acctime = "accshowtime", MaxzoneForTrade = "TradingMaxZone";

string LowzoneForTrade = "TradingLowZone", showlzone = "zonemin", deposit = "fordeposit", pip = "forpip", showgrid = "forshowgrid", showorders = "forshoworder", showlot = "forlot", NeedDeposit = "forneeddeposit";

ObjectCreate(nameea, OBJ\_LABEL, 0, 0, 0, 0);

ObjectSet(nameea, OBJPROP\_XDISTANCE, 10);

ObjectSet(nameea, OBJPROP\_YDISTANCE, 20);

ObjectSetText(nameea, "KZM v0.11 Forex ea", 13, "Broadway", Gold);

ObjectCreate(showhzone, OBJ\_LABEL, 0, 0, 0, 0);

ObjectSet(showhzone, OBJPROP\_XDISTANCE, 10);

ObjectSet(showhzone, OBJPROP\_YDISTANCE, 39);

ObjectSetText(showhzone, " HZone : " + DoubleToStr(maxh, 5), 10, "@Arial Unicode MS", clrAquamarine);

ObjectCreate(MaxzoneForTrade, OBJ\_LABEL, 0, 0, 0, 0);

ObjectSet(MaxzoneForTrade, OBJPROP\_XDISTANCE, 10);

ObjectSet(MaxzoneForTrade, OBJPROP\_YDISTANCE, 52);

ObjectSetText(MaxzoneForTrade, " Max zone trading : " + DoubleToStr(cm01, 5), 10, "@Arial Unicode MS", clrAquamarine);

ObjectCreate(LowzoneForTrade, OBJ\_LABEL, 0, 0, 0, 0);

ObjectSet(LowzoneForTrade, OBJPROP\_XDISTANCE, 10);

ObjectSet(LowzoneForTrade, OBJPROP\_YDISTANCE, 68);

ObjectSetText(LowzoneForTrade, " Min zone trading : " + DoubleToStr(cmi02, 5), 10, "@Arial Unicode MS", clrAquamarine);

ObjectCreate(showlzone, OBJ\_LABEL, 0, 0, 0, 0);

ObjectSet(showlzone, OBJPROP\_XDISTANCE, 10);

ObjectSet(showlzone, OBJPROP\_YDISTANCE, 84);

ObjectSetText(showlzone, " Lzone : " + DoubleToStr(minl, 5), 10, "@Arial Unicode MS", clrAquamarine);

ObjectCreate(deposit, OBJ\_LABEL, 0, 0, 0, 0);

ObjectSet(deposit, OBJPROP\_XDISTANCE, 10);

ObjectSet(deposit, OBJPROP\_YDISTANCE, 100);

ObjectSetText(deposit, " Recommended Minimum Capital : " + DoubleToStr(llo2, 1) + "USD.", 10, "@Arial Unicode MS", clrAquamarine);

ObjectCreate(deposit, OBJ\_LABEL, 0, 0, 0, 0);

ObjectSet(deposit, OBJPROP\_XDISTANCE, 10);

ObjectSet(deposit, OBJPROP\_YDISTANCE, 100);

ObjectSetText(deposit, " From : " + DoubleToStr(PipDeposit,1) + "Pips.", 10, "@Arial Unicode MS", clrAquamarine);

ObjectCreate(showgrid, OBJ\_LABEL, 0, 0, 0, 0);

ObjectSet(showgrid, OBJPROP\_XDISTANCE, 10);

ObjectSet(showgrid, OBJPROP\_YDISTANCE, 116);

ObjectSetText(showgrid, " GridPer : " + DoubleToStr(GridPerPipW,5) + "Pips.", 10, "@Arial Unicode MS", clrAquamarine);

ObjectCreate(accountname, OBJ\_LABEL, 0, 0, 0, 0);

ObjectSet(accountname, OBJPROP\_XDISTANCE, 10);

ObjectSet(accountname, OBJPROP\_YDISTANCE, 132);

ObjectSetText(accountname, " Name : " + AccountName(), 10, "@Arial Unicode MS", clrAquamarine);

ObjectCreate(accbalace, OBJ\_LABEL, 0, 0, 0, 0);

ObjectSet(accbalace, OBJPROP\_XDISTANCE, 10);

ObjectSet(accbalace, OBJPROP\_YDISTANCE, 148);

ObjectSetText(accbalace, " Balance : " + AccountBalance() +" "+ AccountCurrency() , 10, "@Arial Unicode MS", clrAquamarine);

ObjectCreate(acctime, OBJ\_LABEL, 0, 0, 0, 0);

ObjectSet(acctime, OBJPROP\_XDISTANCE, 10);

ObjectSet(acctime, OBJPROP\_YDISTANCE, 164);

ObjectSetText(acctime, " Date - Time : " + TimeToStr(TimeLocal()) , 10, "@Arial Unicode MS", clrAquamarine);

ObjectCreate(showorders, OBJ\_LABEL, 0, 0, 0, 0);

ObjectSet(showorders, OBJPROP\_XDISTANCE, 10);

ObjectSet(showorders, OBJPROP\_YDISTANCE, 180);

ObjectSetText(showorders, " MaxOrders : " + IntegerToString(maxsend) , 10, "@Arial Unicode MS", clrAquamarine);

ObjectCreate(showlot, OBJ\_LABEL, 0, 0, 0, 0);

ObjectSet(showlot, OBJPROP\_XDISTANCE, 10);

ObjectSet(showlot, OBJPROP\_YDISTANCE, 196);

ObjectSetText(showlot, " Lot is : " + DoubleToStr(Lot, 2), 10, "@Arial Unicode MS", clrAquamarine);

ObjectCreate(NeedDeposit, OBJ\_LABEL, 0, 0, 0, 0);

ObjectSet(NeedDeposit, OBJPROP\_XDISTANCE, 10);

ObjectSet(NeedDeposit, OBJPROP\_YDISTANCE, 212);

ObjectSetText(NeedDeposit, " Minimum Capital : " + DoubleToStr(llo2, 1) +" "+ AccountCurrency(), 10, "@Arial Unicode MS", clrAquamarine);

}