

TradeX Dev Team

[TradeX-TPS 交易插件 用户指南]

[标准版 1.0]

[2017-04-25]

目 录

| | |
|--|----|
| 1. 什么是 TradeX-TPS..... | 3 |
| 1.1. 产品功能..... | 4 |
| 1.2. 目录说明..... | 4 |
| 1.3. 配置文件..... | 4 |
| 2. TPS - 大智慧..... | 5 |
| 2.1. 安装调试..... | 5 |
| 2.2. 大智慧公式函数..... | 5 |
| 2.2.1. AccountInfo(Category)..... | 5 |
| 2.2.2. PosInfo(Category)..... | 6 |
| 2.2.3. OrderInfo(OperationType, DataType)..... | 6 |
| 2.2.4. BUY(Price, Condition * Volume, FormulaID)..... | 6 |
| 2.2.5. SELL(Price, Condition * Volume, FormulaID)..... | 7 |
| 2.2.6. CancelOrder(Condition * OperationType)..... | 7 |
| 2.2.7. ClearPosition(Condition * Percentage)..... | 7 |
| 2.2.8. Log(Data)..... | 8 |
| 2.2.9. MsgBox(Data)..... | 8 |
| 2.2.10. SendMsg(AppType, Condition * MsgID)..... | 8 |
| 3. TPS - 通达信..... | 9 |
| 3.1. 安装调试..... | 9 |
| 3.2. 通达信公式函数..... | 9 |
| 3.2.1. TDXDLL1(1, STR2CON(CODE), Category, 0) - 查询账户..... | 9 |
| 3.2.2. TDXDLL1(2, STR2CON(CODE), Category, 0) - 查询持仓..... | 10 |
| 3.2.3. TDXDLL1(3, STR2CON(CODE), OperType, DataType) - 查询订单..... | 10 |
| 3.2.4. TDXDLL1(4, STR2CON(CODE), OrderType, Volume) - 买入..... | 10 |
| 3.2.5. TDXDLL1(5, STR2CON(CODE), OrderType, Volume) - 卖出..... | 11 |
| 3.2.6. TDXDLL1(6, STR2CON(CODE), OperType, 0) - 撤单..... | 11 |
| 3.2.7. TDXDLL1(7, STR2CON(CODE), Percentage, 0) - 按比例清仓..... | 11 |
| 3.2.8. TDXDLL1(8, STR2CON(CODE), Data, 0) - 输出日志..... | 12 |
| 3.2.9. TDXDLL1(9, STR2CON(CODE), Data, 0) - 消息框..... | 12 |
| 3.2.10. TDXDLL1(10, STR2CON(CODE), AppType, MsgID) - 发信息..... | 12 |
| 3.3. 用户安全..... | 13 |

[illegible]

重要提醒

- TradeX-TPS 交易插件符合大智慧、通达信规范，通过 TradeX.dll 和 Trade.dll 连接券商服务器，实现下单、撤单、预设比例清仓、以及账户、持仓、可撤单的查询；
- 支持 TradeX.dll 和 Trade.dll；
- 标准版为免费版，欢迎使用！请联系相关技术支持人员获取最新版文档以及其他技术支持！

[illegible]

技术 QQ 群: 551330649 (TradeX-TPS 插件) 318139137 (TradeX 接口) QQ: 3048747297

技术首页: <https://tradexdl1.com/>

TradeX-TPS 插件 <https://pan.baidu.com/s/1dEOAvpZ>

TradeX 交易接口 <http://pan.baidu.com/s/1jIjYq1K>

→ 版本概要 (2017/04/25) <

新发布的 TPS v1.0 特征简述:

- 支持股票买入、卖出、撤单、预设比例平仓，查询账户信息、持仓、可撤单信息；
- 支持通达信、大智慧的股票池和预警公式系统，完美兼容 TradeX.dll、Trade.dll 等交易接口；

下一版将增加：

- 日志输出、消息对话框功能；
- QQ 信息、邮件、微信信息发送；

1. 什么是 TradeX-TPS

TPS(Trade Plugin Solution)交易插件系统 v1.0 针对通达信、大智慧的股票池和预警公式系统进行了扩展，支持 A 股股票的买入、卖出、撤单、预设比例清仓，并允许用户查询账户、持仓、可撤单的数据信息，完美兼容 TradeX.dll、Trade.dll 等交易接口。

目前日志导出、信息框、QQ/邮件/微信暂未实现，后续会予以添加。本标准版完全免费，欢迎大家使用，感谢大家对 TradeX 团队的支持！

1.1. 产品功能

TPS 系统的插件 XP 启动后，同 TPS 服务器建立一个 TCP 长连接，所有数据交互都是通过该 TCP 连接实现。流程如下：



1.2. 目录说明

. \TPSvr\

| | |
|----------------|----------------------------|
| TPSvr.exe | --> TPS 交易插件服务器 |
| TPSvr.ini | --> TPS 配置文件 |
| TradeX.dll | --> TradeX.dll 交易/行情接口 DLL |
| TradeX.lic | --> TradeX 授权许可文件 |
| Trade-mock.dll | --> Trade.dll 交易接口 DLL |
| TradeX.xml | --> 券商数据格式配置文件（一键打新等） |
| TradeX-L2.ini | --> Level 2 行情配置文件 |
| *.dll | --> 其他所需的 32 位 MFC DLL 文件 |

. \插件\

| | |
|--------|---------------|
| XP.dll | --> 插件 DLL 文件 |
| XP.ini | --> 插件配置文件 |

1.3. 配置文件

1. 检查 TPSvr.ini，填写正确的券商账户信息；包括券商代码、券商交易服务器、服务器端口、版本信息、营业部 ID、登录账户号、交易账户、交易密码、通信密码、沪市股东代码、深市股东代码；

【TPSvr.ini】

```
[TRADE]
Qsid=117
Host=mock.tdx.com.cn
Port=7708
Version=6.40
YybId=9000
AccountNo=net828@163.com
TradeAccount=001001001005792
JyPassword=123123
TxPassword=
```

ShareHolderSH=p001001001005793

ShareHolderSZ=p001001002005793

[COMMON]

DebugMode=0

ApiDll=tradeX.dll

BindAddr=127.0.0.1

ListenPort=9099

2. TPS - 大智慧

2.1. 安装调试

1. 从网盘连接/Q 群下载 TPS 软件包. TPS-svr.rar，解压后目录下包括 2 个子目录：TPSvr 和插件；
2. TradeX.dll 用户，需要将购买的 TradeX.lic 复制到.\TPSvr 目录下，修改 TPsvr.ini 中的券商账户信息，并修改 “ApiDll=TradeX.dll”；
Trade.dll 用户，可以直接修改 TPsvr.ini 中的券商账户信息，确保 “ApiDll=TradeX.dll”。
3. 检查 TPsvr.ini，修改正确的券商信息，包括券商代码、券商交易服务器、服务器端口、版本信息、营业部 ID、登录账户号、交易账户、交易密码、通信密码、沪市股东代码、深市股东代码；
4. 复制 XP 插件 (XP.dll, XP.ini) 到大智慧工作根目录；
5. 启动 TPSvr，完成服务器窗口初始化，等待数据的输出显示；
6. 在大智慧的公式系统下调用 “XP@” 函数，加载到预警系统、股票池即可工作。
7. 大智慧当前目录下会生成 XP.log 日志文件。

2.2. 大智慧公式函数

2.2.1. AccountInfo(Category)

说明：

查询账户信息

参数：

| | |
|----------|------------|
| Category | - 查询信息的种类， |
| 0 | 当前账号的总资产 |
| 1 | 当前账号的可用资金 |
| 2 | 当前账号的总持仓市值 |
| 3 | 当前账号的总盈利金额 |
| 4 | 当前账号的持仓数量 |

示例：

“XP@AccountInfo”(2); {查询当前账号的总持仓市值}

2.2.2. PosInfo(Category)

说明:

查询各种交易数据

参数:

- | | |
|----------|------------|
| Category | - 查询信息的种类, |
| 0 | 当前持仓股票数量 |
| 1 | 当前可卖股票数量 |
| 2 | 当前股票成本价 |
| 3 | 当前浮动盈亏金额 |
| 4 | 当前浮动盈亏百分比 |
| 5 | 最新持仓市值 |

示例:

```
"XP@PosInfo"(3);      {查询浮动盈亏金额}
```

2.2.3. OrderInfo(OperationType, DataType)

说明:

查询各种委托数据

参数:

- | | |
|---------------|----------------------------------|
| OperationType | - 查询买卖类型 |
| 0 | 委托买入 |
| 1 | 委托卖出 |
| DataType | - 查询信息类型 |
| 0 | 委托数量 |
| 1 | 委托时间距离当前的时间(秒) -- (本版尚未实现, 以后添加) |

示例:

```
"XP@OrderInfo"(0, 0);      {可撤买单的数量}
```

2.2.4. BUY(Price, Condition * Volume, FormulaID)

说明:

条件成立时(即 条件:=1), 买入指定数量的股票

参数

- | | |
|-----------|--------------------------------------|
| Price | - 买入价; 价格为 0 则市价单 (类型 4) , 价格>0 则限价单 |
| Condition | - 条件返回值 |
| | 0 - 条件不符 |
| | 1 - 条件符合 |
| Volume | - 委托数量 |
| FormulaID | - 公式编号, 区分是哪个公式/条件买入, 避免相同买入和卖出的编号。 |

示例:

```
{上穿 5 日均线限价买入 1000 股}
```

```
条件:Cross(C, MA(C, 5));
```

```
if 条件 > 0 then
```

```
    执行买入:"XP@BUY"(c, 条件*1000, 1);
```

2.2.5. SELL(Price, Condition * Volume, FormulaID)

说明:

条件成立时(即 条件:=1), 卖出指定数量的股票

参数

| | |
|-----------|--------------------------------------|
| Price | - 买入价; 价格为 0 则市价单 (类型 4) , 价格>0 则限价单 |
| Condition | - 条件返回值 0 - 条件不符 1 - 条件符合 |
| Volume | - 委托数量 |
| FormulaID | - 公式编号, 区分是哪个公式/条件买入, 避免相同买入和卖出的编号。 |

示例:

```
{下穿 5 日均线卖出}
条件:Cross( MA(C,5), C);
if 条件 > 0 then
    执行卖出:"XP@SELL"(c, 条件*1000, 1);
```

2.2.6. CancelOrder(Condition * OperationType)

说明:

条件成立时撤销当前股票的买单

参数

| | |
|---------------|---------------------------------|
| Condition | - 条件返回值 0 - 条件不符 1 - 条件符合 |
| OperationType | - 可撤单买卖类型 1 - 买入 2 - 卖出 |

示例:

```
买单已过秒数:= "XP@OrderInfo"(0,1);
条件:买单已过秒数 > 60;
if 条件 then
    撤买单 : "XP@CancelOrder"(条件*1);
```

2.2.7. ClearPosition(Condition * Percentage)

说明:

条件成立时按照预设的百分比清仓。

当预设条件成立, 则读取当前持仓数量, 根据预设的清仓百分比计算出允许的委托数量 (持仓数量*n%), 以

每手数量（100）为最小单位进行取整。

参数

| | |
|------------|-----------------------------------|
| Condition | - 条件返回值 |
| | 0 - 条件不符 |
| | 1 - 条件符合 |
| Percentage | - 清仓的百分比（1 --> 100），例如 70，则清仓 70% |

示例：

{此例子假定当前股票池只有上证指数一个股票, 当上证指数跌破 5 日均线清仓 50%}

条件:Cross(MA(C,5), C);

if 条件 then

清仓: "XP@ClearPosition"(条件*50);

2.2.8. Log(Data)

说明：

把数据写入当前日志文件，日志文件默认存于大智慧的根目录

参数

| | |
|------|------|
| Data | - 数值 |
|------|------|

示例：

{把当前最新周期的收盘价写到日志文件}

写日志:="XP@Log"(close);

2.2.9. MsgBox(Data)

说明：

以弹出消息框的形式显示数据

参数

| | |
|------|------|
| Data | - 数值 |
|------|------|

示例：

{把当前最新周期的收盘价弹出消息框}

消息框:="XP@MsgBox"(c);

2.2.10. SendMsg(AppType, Condition * MsgID)

说明：

条件成立时(即 条件:=1)则发送邮件, QQ, 微信信息

参数

| | |
|-----------|----------|
| AppType | - 发送渠道类型 |
| | - 0 邮件 |
| | - 1 QQ |
| | - 2 微信 |
| Condition | - 条件返回值 |

0 - 条件不符

1 - 条件符合

MsgID

- 消息编号

示例:

{价格大于 10 元发送编号为 5 的邮件}

条件: $C > 10$;

if 条件 then

发邮件: "XP@SendMsg"(0, 条件*5);

3. TPS - 通达信

3.1. 安装调试

1. 从网盘连接/Q 群下载 TPS 软件包. TPS-svr.rar, 解压后目录下包括 2 个子目录: TPSvr 和插件;
2. 检查 TPSvr.ini, 修改正确的券商信息, 包括券商代码、券商交易服务器、服务器端口、版本信息、营业部 ID、登录账户号、交易账户、交易密码、通信密码、沪市股东代码、深市股东代码;
3. TradeX.dll 用户, 需要将购买的 TradeX.lic 复制到.\TPSvr 目录下, 修改 TPsvr.ini 中的券商账户信息, 并修改 "ApiDll=TradeX.dll";
Trade.dll 用户, 可以直接修改 TPsvr.ini 中的券商账户信息, 确保 "ApiDll=Trade.dll".
4. 将 XP 目录下的所有文件 (XP.DLL、XP.ini) 复制到通达信\T0002\dlls 目录下;
5. 登录通达信客户端, 绑定 DLL; 绑定到第几号 DLL, 函数调用的函数名就是 TDXDLL 数字, 通常绑定为第一个 DLL, 因此调用的函数名是 TDXDLL1。详情参见函数描述。
6. 启动 TPSvr 服务器, 完成服务器窗口初始化, 等待数据的输出显示;
7. 在通达信的公式系统下调用 "TDXDLL1@" 函数, 加载到预警系统、股票池即可工作。
8. 通达信当前目录下会生成 XP.log 日志文件。

3.2. 通达信公式函数

通达信的函数最多只能有 4 个数值型参数, 且参数不能省略; 所以部分函数的参数不足 4 个则以 0 填充, 无实际意义。股票代码需要调用 "STR2CON(CODE)" 转为数值后才能作为参数传递。

3.2.1. TDXDLL1(1, STR2CON(CODE), Category, 0) - 查询账户

说明:

查询账户信息

参数:

CODE

- 股票代码

Category

- 查询信息的种类,

0 当前账号的总资产

1 当前账号的可用资金

- 2 当前账号的总持仓市值
- 3 当前账号的总盈利金额
- 4 当前账号的持仓数量

示例：

账户持仓市值：TDXDLL1(1, STR2CON(CODE), 2, 0);

3.2.2. TDXDLL1(2, STR2CON(CODE), Category, 0) - 查询持仓

说明：

查询持仓信息

参数：

- | | |
|----------|-------------|
| CODE | - 股票代码 |
| Category | - 查询信息的种类, |
| | 0 当前持仓股票数量 |
| | 1 当前可卖股票数量 |
| | 2 当前持仓股票成本价 |
| | 3 当前浮动盈亏金额 |
| | 4 当前浮动盈亏百分比 |
| | 5 最新持仓市值 |

示例：

持仓股票数量：TDXDLL1(2, STR2CON(CODE), 0, 0);

3.2.3. TDXDLL1(3, STR2CON(CODE), OperType, DataType) - 查询订单

说明：

查询订单信息

参数：

- | | |
|----------|------------------|
| CODE | - 股票代码 |
| OperType | - 查询买卖类型 |
| | 0 委托买入 |
| | 1 委托卖出 |
| DataType | - 查询信息类型 |
| | 0 委托数量 |
| | 1 委托时间距离当前的时间(秒) |

示例：

可撤买单数量：TDXDLL1(3, STR2CON(CODE), 0, 0);

3.2.4. TDXDLL1(4, STR2CON(CODE), OrderType, Volume) - 买入

说明：

条件成立时买入指定数量的股票

参数

| | |
|--------|----------------------------------|
| CODE | - 股票代码 |
| Price | - 买入价；价格等于 0 为市价单，价格不等于 0 则为限价单。 |
| Volume | - 委托数量，如果数量为 0 则直接返回 |

示例：

```
{上穿 5 日均线买入 1000 股}
条件:Cross(C, MA(C, 5));
买入股票:=TDXDLL1(4, STR2CON(CODE), 9.80, 条件*1000);
```

3.2.5. TDXDLL1(5, STR2CON(CODE), OrderType, Volume) - 卖出

说明：

条件成立时卖出指定数量的股票

参数

| | |
|--------|----------------------------------|
| CODE | - 股票代码 |
| Price | - 买入价；价格等于 0 为市价单，价格不等于 0 则为限价单。 |
| Volume | - 委托数量，如果数量为 0 则直接返回 |

示例：

```
{下穿 5 日均线卖出 1000 股}
条件:Cross(MA(C, 5), C);
买入股票:=TDXDLL1(5, STR2CON(CODE), 9.80, 条件*1000);
```

3.2.6. TDXDLL1(6, STR2CON(CODE), OperType, 0) - 撤单

说明：

条件成立时撤销当前股票的委托单

参数

| | |
|----------|-----------|
| CODE | - 股票代码 |
| OperType | - 可撤单买卖类型 |
| | 1 - 买入 |
| | 2 - 卖出 |

示例：

```
买单已过秒数:= TDXDLL1(4, STR2CON(CODE), 0, 1);
条件:买单已过秒数 > 60;
撤买单:=TDXDLL1(6, STR2CON(CODE), 条件*1, 0);
```

3.2.7. TDXDLL1(7, STR2CON(CODE), Percentage, 0) - 按比例清仓

说明：

条件成立时按照预设的百分比清仓。

当预设条件成立，则读取当前持仓数量，根据预设的清仓百分比计算出允许的委托数量（持仓数量*n%），以每手数量（100）为最小单位进行取整。

参数：

| | |
|------|--------|
| CODE | - 股票代码 |
|------|--------|

Percentage - 清仓的百分比 (1 --> 100) , 例如*70, 则清仓 70%

示例:

{此例子假定当前股票池只有上证指数一个股票, 当上证指数跌破 5 日均线清仓 70%}

条件:Cross(MA(C, 5), C);

清仓:= TDxDLL1(9, STR2CON(CODE), 条件*70, 0);

3.2.8. TDxDLL1(8, STR2CON(CODE), Data, 0) - 输出日志

说明:

把数据写入当前日志文件, 日志文件默认存于 XP 的目录

参数

CODE - 股票代码

Data - 数值

示例:

{把当前最新周期的收盘价写到日志文件}

写日志:=TDxDLL1(8, STR2CON(CODE), Close, 0);

3.2.9. TDxDLL1(9, STR2CON(CODE), Data, 0) - 消息框

说明:

以弹出消息框的形式显示数据

参数

CODE - 股票代码

Data - 数值

示例:

{把当前最新周期的收盘价弹出消息框}

消息框:=TDxDLL1(9, STR2CON(CODE), Close, 0);

3.2.10. TDxDLL1(10, STR2CON(CODE), AppType, MsgID) - 发信息

说明:

条件成立时则发送邮件, QQ, 微信信息

参数

CODE - 股票代码

AppType - 发送渠道类型

- 0 邮件

- 1 QQ

- 2 微信

MsgID - 消息编号 (如果为 0, 则函数不执行并直接返回)

示例:

{价格大于 10 元发送编号为 1 的微信信息}

条件:C > 10;

发邮件:=TDxDLL1(10, STR2CON(CODE), 2, 条件*1);

3.3. 用户安全

TradeX.dll/Trade.dll 只是一个 DLL 分享库，而非软件平台，因而无法知道上层应用程序的策略代码；TPSvr 只通过 TradeX.dll/Trade.dll 同券商的交易服务器有数据往来，不连接其它任何不明 IP 地址的服务器；

TradeX 的 License 文件是一个经过加密的本地 License 文件，dll 文件调用时将 Lic 文件内容授权匹配，没有任何对外的 IP 连接。用户可以通过打开防火墙进行监测保护自己的数据安全！

您可以三种方式检查安全：

- >> 通过网络监控软件监控网络连接；
- >> 在任务管理器中实时查看程序当前所连接的服务器 IP 地址；
- >> 您还可以设置 windows 自带的防火墙，限制本接口可以访问哪些 IP，消除安全顾虑。

众所周知 360 杀毒软件的误报率极高，会提示威胁；建议您改用其他杀毒软件，比如腾讯电脑管家，卡巴斯基等。请不要使用破解版，可能会被网上其他黑客植入后门，窃取交易密码，由此造成的损失我们不予负责。