Information in this document is subject to change without notice.

THE TRADING SYSTEMS IN THIS BOOK ARE EXAMPLES ONLY, AND HAVE BEEN INCLUDED SOLELY FOR EDUCATIONAL PURPOSES. OMEGA RESEARCH DOES NOT RECOMMEND THAT YOU USE ANY SUCH TRADING SYSTEM, AS THE USE OF ANY SUCH TRADING SYSTEM DOES NOT GUARANTEE THAT YOU WILL MAKE PROFITS, INCREASE PROFITS, OR MINIMIZE LOSSES. THE SOLE INTENDED USES OF THE TRADING SYSTEMS INCLUDED IN THIS BOOK ARE TO DEMONSTRATE THE WAYS IN WHICH EASYLANGUAGE CAN BE USED TO DESIGN PERSONAL TRADING SYSTEMS AND TO SHOW SOME EXAMPLES OF HOW CERTAIN POPULAR, WELL-KNOWN TRADING STRATEGIES MAY BE INCORPORATED INTO PERSONAL TRADING SYSTEMS. OMEGA RESEARCH, INC. IS NOT ENGAGED IN RENDERING ANY INVESTMENT OR OTHER PROFESSIONAL ADVICE. IF INVESTMENT OR OTHER PROFESSIONAL ADVICE IS REQUIRED, THE SERVICES OF A COMPETENT PROFESSIONAL SHOULD BE SOUGHT.

Contents

INTRODUCTION

Welcome to Volume 5

Welcome to Volume 5 of the *Omega Research System Trading and Development Club*, the revolutionary learning tool offered by Omega Research to help you make the most of your trading potential.

This fifth volume of the *Omega Research System Trading and Development Club* contains 10 new systems we've created to help you get started developing your own systems.

The systems in Volume 5 incorporate popular concepts such as exponential moving averages, intermarket analysis, displaced indicators, and Fibonacci ratios. We identify the most common problems with these types of systems and provide techniques to overcome them as well as present different ways of using them. By reviewing and testing these 10 systems, you'll be able to understand these techniques and use them, or a variation of them, in your own systems.

This book walks you step-by-step through the process we went through to develop these systems, from coming up with a feasible trading idea, to writing down your trading rules, to writing them in EasyLanguage™, to taking into consideration money management and risk control factors. Our goal is for you to understand why we wrote these systems the way we did.

We recommend you study the EasyLanguage techniques we used, look at the System Report, read about the factors we took into account to deem that a system may have merit, and think about how you can combine some of these popular ideas with your own to develop a system *you* want to trade.

***IMPORTANT NOTICE*: The trading systems in this book are examples only, and have been included solely for educational purposes. Omega Research does not recommend that you use any such trading system, as the use of any such trading system does not guarantee that you will make profits, increase profits, or minimize losses. The sole intended uses of the trading systems included in this book are to demonstrate the ways in which EasyLanguage can be used to design personal trading systems and to show some examples of how certain popular, well-known trading strategies may be incorporated into personal trading systems.**

Contents at a Glance

This book begins by discussing some of the broader concepts of system development before moving on to the description of each system. We grouped the different types of systems together. The contents are:

***IMPORTANT NOTE:** We suggest you read the book from front to back because there is instructional material in each section and it is not repeated throughout.*

Additional Educational Services

Omega Research is committed to enhancing individual trading potential through quality education. To learn more about system trading, an Omega Research product, or EasyLanguage, visit our web site at **www.omegaresearch.com** or call **(800) 439-7995** (outside US 305-551-9991) and ask about the following educational services:

Workshops

Omega Research offers a variety of workshops on the products and technical analysis. Workshops are an excellent way to learn how to use the products, learn about technical analysis and system trading and/or EasyLanguage. Spend a day with a Product Training Specialist and exchange ideas with other users like yourself. All workshops provide a 100% satisfaction guarantee. Call now for more information or to register — space is limited!

EasyLanguage Resource Center

One of the best ways to learn is by example, and the EasyLanguage Resource Center on our web site is an excellent source of examples. In this Resource Center, we list all the analysis techniques — indicators and trading systems—published in the *Technical Analysis of Stocks and Commodities* magazine, as well as popular analysis techniques worth taking a look at. Access to this Resource Center is free of charge. Feel free to download and review any of the analysis techniques and their descriptions. Our web site address is **www.omegaresearch.com**.

Getting Started

To begin reviewing your systems, transfer the analysis techniques into your TradeStation® library and then apply the system you want to review to a chart. Use the System Report to view the system results and take a look at the EasyLanguage instructions by opening the system in the PowerEditor™.

To transfer the analysis techniques into TradeStation:

1. Place the System Trading and Development Club CD in the CD-ROM drive.
2. Start the PowerEditor. In Windows 95, click **Start**, choose **Programs**, choose **Omega Research** and choose **TradeStation PowerEditor**. In Windows 3.x, choose **TradeStation PowerEditor** from the Omega Research program group.
3. In the PowerEditor, use the **File - Open** menu sequence.
4. Click **Transfer**.
5. Select the **Transfer analysis techniques FROM EasyLanguage Archive File** option and click **OK**.
6. Click **Scan**.
7. In the **Enter drive letter to scan** edit box, enter the drive letter for your CD-ROM drive (normally D), and click **OK**. The ELA file on the CD is placed in the list.
8. Choose **STAD5.ELA** from the list and click **OK**.
9. In the **Transfer** dialog box, select **Transfer All** and click **OK**.
10. Once the files are transferred and verified, a dialog box appears informing you that the transfer was performed successfully. Click **OK**.

For your convenience, the names of the systems in this volume all begin with STAD5. You can now open the systems in the PowerEditor and view the EasyLanguage instructions and/or apply them to a chart in TradeStation. You can remove your CD from the CD-ROM drive and store it in a safe place. As you apply the systems and work with them, refer to this book for detailed explanations of the systems and the EasyLanguage used to create them. For instructions on applying systems and viewing the System Report, please refer to your *TradeStation User's Manual*.

Note to SuperCharts® 4 Users: *To transfer the systems into SuperCharts, use the* ***Tools - QuickEditor*** *menu sequence and select* ***Transfer****. Keep in mind, however, that although you can apply the systems in SuperCharts, you will not be able to view the EasyLanguage instructions in the QuickEditor. This is because the systems were designed in the PowerEditor. Also, if you are using SuperCharts End of Day, some of the systems will not apply as they are designed for intraday trading. Since the purpose of the Club is to provide you with a learning tool, and viewing the EasyLanguage instructions is an essential part of this learning process, the use of this club for SuperCharts users is limited.*

Note to TradeStation or SuperCharts 3.x Users: *The systems for the Club were designed using TradeStation 4. As such, some of the features used, such as automatic drawing of trendlines and/or text, are not available in previous versions of TradeStation (or SuperCharts). An effort is made to provide a variety of systems that incorporate both long standing and new features; however, keep in mind that as new features are developed, we will naturally want to showcase and educate users on these features; therefore, users of the most recent version of our software will be able to make the most use of the Club.*

Obtaining Technical Support

Depending on your question, there are two resources at your disposal: the EasyLanguage Support Department and the STAD Club E-Mail Address.

EasyLanguage Support Department

The EasyLanguage Support Department provides EasyLanguage support via fax and is designed to help you troubleshoot an analysis technique or trading system you are currently working on. For example, if you are incorporating a trading system from the Club into your own and have a question about the implementation, the EasyLanguage Support Department can answer it.

Please keep in mind that while this department can answer any EasyLanguage question, it cannot answer questions about the STAD Club specifically, such as the theory behind a system in the Club, why a system was developed a certain way, or why the system is not performing as you expect it to, etc.

Fax Number:
(305) 221-6831

E-Mail Address:
easylang@omegaresearch.com

Be sure to include the following information in your fax:

- Name
- Security Block or Customer ID Number
- Telephone Number
- Fax Number
- Product you own
- EasyLanguage instructions you are working on
- Detailed description of your problem

Please allow 48 hours for a response.

STAD Club E-Mail Address

Another resource at your disposal is the STAD Club e-mail address.

Please realize that when you send a message to this e-mail address, you will not receive a response directly; your message will be reviewed and the answer incorporated into the next volume of the STAD Club, when applicable. Therefore, if you need technical support on EasyLanguage, please use the above fax number or e-mail address.

stadclub@omegaresearch.com

Please send any comment, suggestion, or question regarding the systems in the Club to the STAD Club e-mail address, and each subsequent volume we will publish the most common suggestions and questions.

CHAPTER 1

General System Development Concepts

Since the introduction of computers for the systematic analysis of markets, selecting the data compression to use for trading has been the subject of many discussions. Computers provide the ability not only to display updated information (trades, prices, etc.) in real time or at acceptable delay, but also the ability to perform mathematical and statistical analysis on this data as soon as the data becomes available.

There are many factors that you need to consider when selecting the time compression(s) to trade. The selection of the time compression is as delicate and important a matter as selecting the market and system to trade—as much attention and thought should be given to it.

This chapter covers the main points of consideration when selecting the data compression used to perform your analysis.

In This Chapter

Your Trading Objectives

Just like when you are writing or selecting a trading system to trade, the most important factor when selecting the bar compression to use for trading is your trading objectives.

The bar compression used for your analysis should be in direct proportion to your (point-wise) trade by trade objective. If your intention is to capture an 1/8th or a 1/4 of a point per trade when trading stocks (basically scalping the instrument you are trading), you will want to use intra-day bars. The smaller the trade objective, the smaller the trading bars you should use. Likewise, if your objective is to capture a 10-point trend in a stock that usually has a 1-point range, you will want to work with a larger data compression such as daily bars.

Noise is generally defined as small and/or insignificant movements that distort or interfere with the interpretation of greater more significant moves.

Just as it seems obvious that it would be virtually impossible to identify a monthly trend from 1-minute charts because of the amount noise present in the 1-minute bars, you will have an extremely difficult time attempting to capture a 1/2 point move when analyzing bars that average 10 points in range.

Barring protective stops, the general rule of thumb is to have one trade every 5 - 10 bars. However, when your system is designed to capture trends, this number may be significantly higher since trending systems should let profits run.

Character of the Market

Some markets are not as active as others, and these markets will have either a greater amount of noise or extensive periods where they are just moving sideways in a repetitive “W” pattern (up tick, down tick, up tick, down tick, etc). These markets generally need to be analyzed with larger data compressions because the moves are so gradual or sporadic and the greater time compressions will ‘hide’ these smaller moves, filtering out most of the noise.

On the other hand, fast markets like the S&P 500 futures contracts will move extremely fast and if studied through larger data compresssions, will leave you with very little reaction time for analysis. With this type of market, it is better to use smaller bars—bigger bars will mean less data at your disposal to analyze. You will be much better off and have a much better feel of where the market is moving if you performing your analysis on bars that break the price movements into smaller segments.

In addition to your trading objectives and the character of the market, there is one final item you should consider when choosing a data compression to use, and that is the particular analysis technique you plan to use.

The Analysis You Have in Mind

If you have a particular technique that you want to use for a specific market, you will need to judge the character of the market in relation to the type of the analysis you are planning to use. Regardless of how you want to trade, remember that different markets can look *very* different when you change the data compression.

Symbols that are extremely volatile like the S&P 500 futures can have three or more distinct “looks.” Take a look at Figures 1 through 3.

On short time frames volatile symbols might seem to form trends, with very well defined tops and bottoms (Figure 1), while in intermediate time frames they might look like a sideways market (Figure 2).

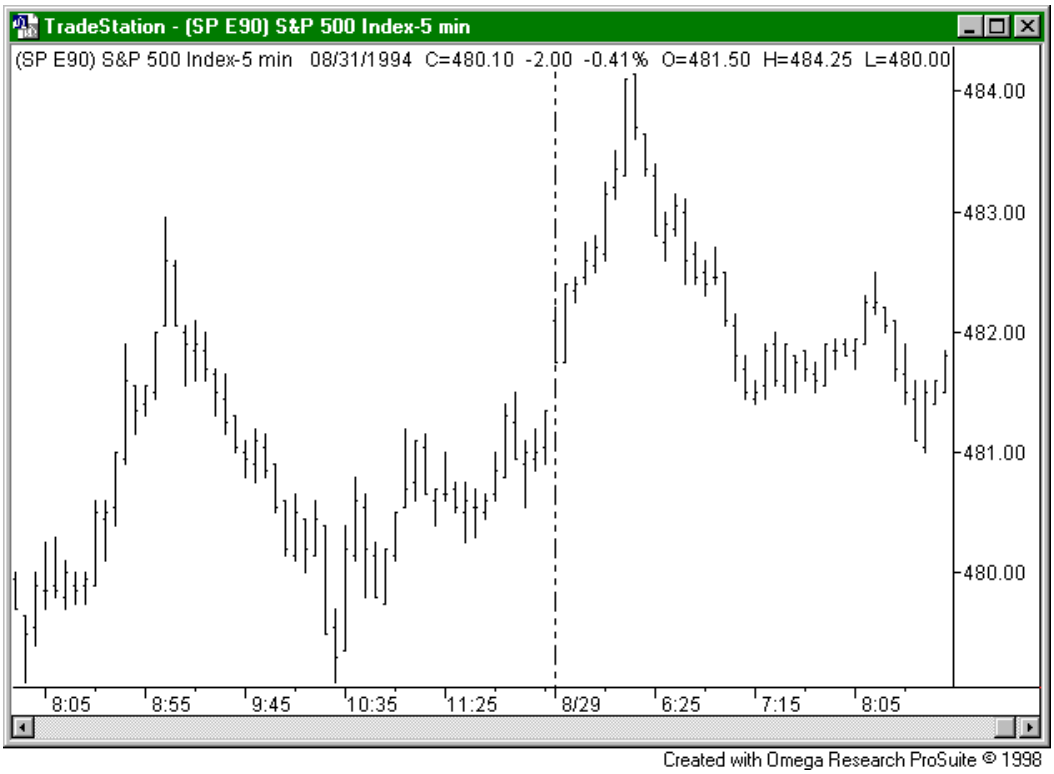


Figure 1. A 5-minute chart of the S&P 500, showing strong top & bottom formations

Figure 2. A 30-minute chart of the S&P, showing the characteristics of a sideways market

Now look at Figure 3. The same market in the long term changes to a strongly trending market.

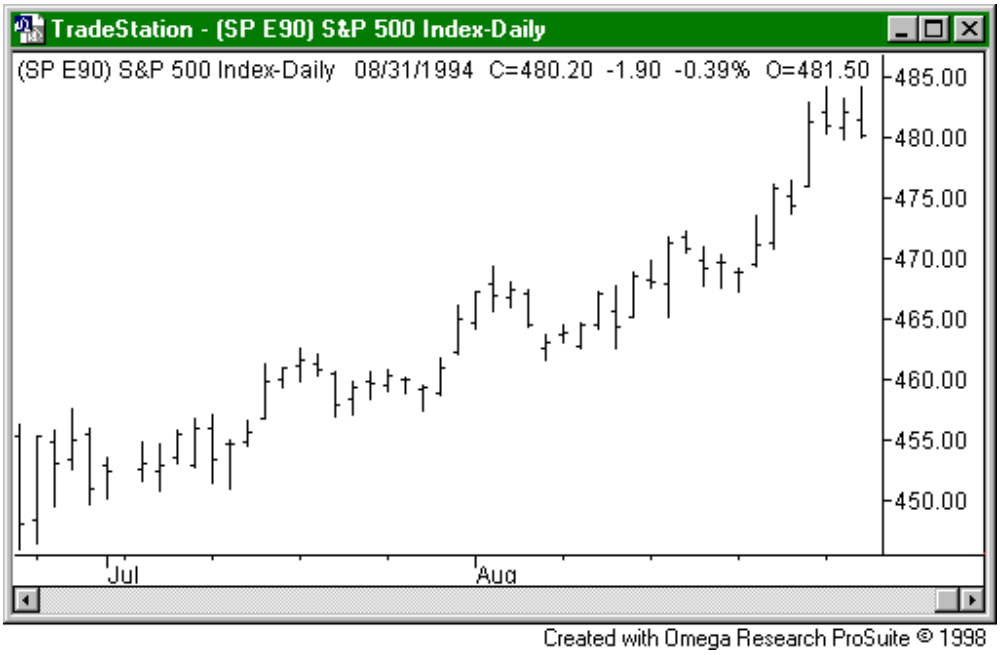


Figure 3. A daily chart of the S&P, demonstrating strong trending characteristics

Most stocks will have sideways market characteristics in the short time compression (intra-day bars), support and resistance characteristics in the intermediate (daily) data compressions, and then revert to trending characteristics in the longer time frames.

If you are trying to capture small price moves on a bar with a large data compression, you can end up with several orders on a bar. When this occurs, the bar assumptions and bouncing tick feature necessarily takes effect and will affect the results of your system tests. Another important issue is that of bouncing ticks—keep in mind that the bouncing ticks feature will be less important and have less relevance on bars with a smaller data compression.

Bar Assumptions & Bouncing Ticks

When you apply a system to a chart and it is updating on a real-time basis, TradeStation can evaluate the system as each tick is collected, so your system executes the trades with exact precision. However, when you create a chart using data that has already been collected, TradeStation must make certain assumptions about price movement. Since the assumed movement may not be how the prices moved in reality, historical system results may vary from results obtained when the same data is gathered on a real-time basis.

For example, say you've been tracking Intel on a 5-minute chart. For any historical bar on the chart, the only pieces of information that TradeStation has available to it for each bar are the date, time, open, high, low, close, and volume (plus open interest, when applicable). This means TradeStation knows the price that opened that 5-minute bar, the high & low prices that were reached during the 5 minutes, and the price that closed the 5-minute bar. However, TradeStation doesn't know the order in which the prices were collected, for example, was the high reached before the low? Did price movement reverse at any point during the 5 minutes?

TradeStation simulates market activity by assuming a certain price order. When the open is closer to the low than to the high, TradeStation assumes that the low was reached first. Likewise, if the open is closer to the high, then TradeStation assumes that the high was reached first. Figure 4 illustrates this assumption:

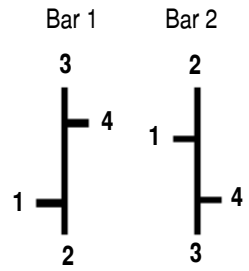


Figure 4. Assumed price movement

In the case of Bar 2, TradeStation looks at the open of the bar, considers every price as it moves up to the high, considers every price as it moves down to the low, and finally, considers every price as it moves back to the close.

Say we are currently in a long position and have placed both a profit target and a money management stop, as shown in Figure 5. Based on TradeStation's price movement assumption, the profit target would be filled and the money management stop would be cancelled.

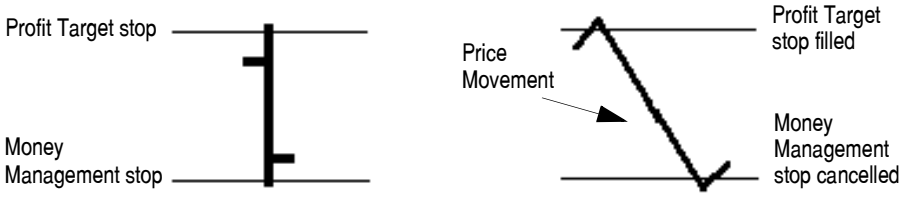


Figure 5. Profit target hit and filled, money management stop cancelled

TradeStation always assumes this price movement, but in a further attempt to simulate actual market movement, TradeStation makes us of additional logic when evaluating stop and limit orders. This logic is referred to as "Bouncing Ticks."

With the Bouncing Ticks logic, TradeStation goes through a bar, and when it reaches the target price of any active order, it fills the order and then "bounces back" a certain percentage of the bar's range to check for additional orders before it proceeds looking at the price movement in the assumed direction. By default, Bouncing Ticks are set to 10% of the bar's range, and this setting is user-definable. If TradeStation finds any other order during the bounce, the order is filled before TradeStation proceeds looking at price movement in the assumed direction.

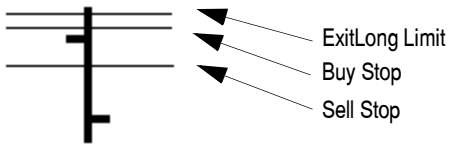


Figure 6. Three orders placed

In Figure 6, the first order filled is the Buy Stop. Without Bouncing Ticks, the second order that would have been filled is the ExitLong Limit order; however, because of Bouncing Ticks, TradeStation fills the Buy Stop and then looks back 10% for any other order. In this example, it finds and fills the Sell Stop order, and you are exited from the long position and entered into a short position. The ExitLong Limit order is canceled because you are now in a short position. Figure 7 shows the price movement along the bar and the orders as they are filled.

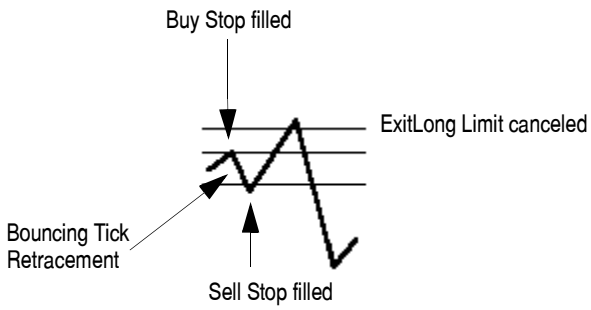


Figure 7. TradeStation's assumed price movement

Summary

Time compression is a fundamental aspect of the systematic analysis of data, and by keeping these points in mind, you can make the time compression work for you—changing it from another issue that needs to be dealt with to a source of new trading opportunities!

CHAPTER 2

Trending Systems

Trending systems are systems designed for trending markets — they have the following characteristics:

- The systems are designed never to miss the big move; they will either always be in the market or contain stop orders that will stop you into the market.
- They attempt to limit losses during the market's sideways mode; no system will make money in every market condition, but a good system will limit losses in market conditions for which it was not designed.
- Profits are concentrated in a few big trades; they have a low percentage of profitable trades. This makes them psychologically difficult to trade and underscores the importance of never missing a big move.

Even though trending systems are difficult to trade, they are popular — it's human nature to want to cash in on the big moves. In this chapter, we present three trending systems that differ in their approach but that are all designed to capture big moves and limit losses during directionless and/or volatile phases.

In This Chapter

Currency/Bond System

The Currency/Bond System is based on the positive correlation between foreign currencies and U.S. treasury bonds. When the bond market is bullish, foreign currencies tend to do well against the U.S. dollar; when bonds are bearish, foreign currencies usually are weak in comparison to the dollar.

We use three exponential moving averages to determine the trend of the currency. Then, we use an exponential moving average of bonds to confirm (or not to confirm) the trend of the currency.

If the moving averages indicate an uptrend, we have a setup to buy. Our buy entry will be at the high of the setup bar plus 50% of the average true range. When the moving averages indicate a downtrend, we have a setup to sell short. Our sell entry will be at the low of the setup bar minus 50% of the average true range. The setup will remain in effect for five bars.

For both long and short positions, we'll set a money management stop and use a trailing stop. Our signal to exit the position will be a crossover of two moving averages. Figure 1 shows the system applied to a daily Swiss Franc and U.S. Treasury Bond chart.

Figure1.The Currency/Bond system applied to a daily chart of the Swiss Franc and US Treasury Bonds

Defining Your Trading Rules

In this system, we defined long and short entries and exits. We also did some setup work to calculate the moving averages and the entry points. The setups, entries, and exits are defined next.

Setup

a) Calculate the 4-, 12-, and 23-bar exponential moving averages (EMAs) of the currency's closing prices.

b) Calculate a 50-bar EMA of the bond closes.

c) Calculate the 10-bar average true range (ATR) of the currency and 50% of the ATR.

Long Entries

a) Check for the 4-bar EMA to be greater than it was one bar ago, for the 12-bar EMA to be above 23-bar EMA, and for the 50-bar EMA (of bonds) to be greater than it was one bar ago.

b) Once the trends of the currency and bonds are up, add 50% of the 10-bar ATR to the high of the setup bar. That is our buy point; we'll go long when the market reaches that price.

c) The buy point will remain active for five bars.

Short Entries

a) Check for the 4-bar EMA to be less than it was one bar ago, for the 12-bar EMA to be below the 23-bar EMA, and for the 50-bar EMA (of bonds) to be less than it was one bar ago.

b) Once the trends of the currency and bonds are down, subtract 50% of the 10-bar ATR from the low of the setup bar. That is our sell point; we'll sell short when the market reaches that price.

c) The sell point will remain active for five bars.

Exit Orders

a) After we enter a position, we'll set a money management stop and a trailing stop

b) We'll also exit our long position when the 12-bar EMA crosses below the 23-bar EMA; we'll exit our short position when the 12-bar EMA crosses above the 23-bar EMA.

Designing & Formatting

This section presents the EasyLanguage instructions and formatting for the system, with the EasyLanguage instructions broken down and explained line by line.

EasyLanguage Instructions: Currency/Bond (STAD5: Currency/Bond)

```
Input: Fast(4), Med(12), Slow(23), BondMA(50), ActvBar(5), StopAmt(1000);
Vars: FastXAvg(0), MedXAvg(0), SlowXAvg(0), LongXAvg(0), LSetupbar(0), SSetupbar(0), BuyLevel(0),
SellLevel(0);

{Assignment of Exponential Average calculations}
FastXAvg = XAverage(Close, Fast);
MedXAvg = XAverage(Close, Med);
SlowXAvg = XAverage(Close, Slow);
LongXAvg = XAverage(Close, BondMA) of Data2;

{Buy Setup}
If FastXAvg > FastXAvg[1] AND MedXAvg > SlowXAvg AND
LongXAvg > LongXAvg[1] then Begin
        LSetupBar = BarNumber;
        BuyLevel = High + .50*Average(TrueRange, 10);
End;

{Sell Setup}
If FastXAvg < FastXAvg[1] AND MedXAvg < SlowXAvg AND
LongXAvg < LongXAvg[1] then Begin
        SSetupBar = BarNumber;
```

```
        SellLevel = Low - .50*Average(TrueRange, 10);
End;

{Long Entry}
If BarNumber - LSetupbar < Actvbar thenBuy at BuyLevel Stop;

{Short Entry}
If BarNumber - SSetupbar < Actvbar thenSell at SellLevel Stop;

{Exits}
If MedXAvg < SlowXAvg then ExitLong next bar at Market;
If MedXAvg > SlowXAvg thenExitShort next bar at Market;
```

We wrote the $ Risk Trailing stop using EasyLanguage rather than enabling it in the Format System dialog. You can do it either way.

```
{Money Management Stops}
ExitLong next bar at Close - StopAmt/BigPointValue Stop;
ExitShort next bar at Close + StopAmt/BigPointValue Stop;
```

Inputs

Following is the list of all the inputs we used in this system:

Input	Default	Description
Fast	4	A length value, expressed in bars, for the short-term exponential moving average of the close.
Med	12	A length value, expressed in bars, for the medium-term exponential moving average of the close.
Slow	23	A length value, expressed in bars, for the long-term exponential moving average of the Close.
BondMA	50	A length value, expressed in bars, for the exponential moving average of the bonds in Data2.
ActvBar	5	A length, expressed in bars, in which the entry signals are valid.
StopAmt	1000	The dollar amount to be used for the money management stop.

In addition to these inputs, we define the following variables:

```
Vars: FastXAvg(0), MedXAvg(0), SlowXAvg(0), LongXAvg(0), LSetupbar(0), SSetupbar(0),
BuyLevel(0), SellLevel(0);
```

Setup

The values of the exponential moving averages are assigned to variables for easy reference throughout the system. FastXAvg is the short-term exponential moving average of the close. MedXAvg is the medium-term exponential moving average of the close. SlowXAvg is the long-term exponential moving average of the close. LongXAvg is the long-term exponential moving average of the bonds, which are plotted in Data2.

```
FastXAvg = XAverage(Close, Fast);
MedXAvg = XAverage(Close, Med);
SlowXAvg = XAverage(Close, Slow);
LongXAvg = XAverage(Close, BondMA) of Data2;
```

When the FastXAvg is greater than the FastXAvg of the previous bar, the MedXAvg is greater than the SlowXAvg, and the LongXAvg is greater than the LongXAvg of the previous bar, then the LSetupBar variable is assigned the current bar number to mark the beginning of the buy setup.

The BuyLevel variable, which will be used as the long entry stop price, is calculated as the setup bar high plus half of the 10-bar average true range.

```
If FastXAvg > FastXAvg[1] AND MedXAvg > SlowXAvg AND
LongXAvg > LongXAvg[1] then Begin
                                LSetupBar = BarNumber;
                                EntryLevel = High + .50*Average(TrueRange, 10);
End;
```

When the FastXAvg is less than the FastXAvg of the previous bar, the MedXAvg is less than the SlowXAvg, and the LongXAvg is less than the LongXAvg of the previous bar, the SSetupBar variable is assigned the current bar number to mark the beginning of the sell setup. The SellLevel variable, which will be used as the short entry stop price, is calculated as the setup bar low minus half of the 10-bar average true range.

```
If FastXAvg < FastXAvg[1] AND MedXAvg < SlowXAvg AND
LongXAvg < LongXAvg[1] then Begin
                                SSetupBar = BarNumber;
                                EntryLevel = Low - .50*Average(TrueRange, 10);
End;
```

Long Entry

While within the buy setup period (as determined by the ActvBar input), a buy stop order will be placed at BuyLevel. The buy stop order will continue to be generated until the difference between the current bar number and the bar number of the setup bar (LSetupBar) meets or exceeds the value of the ActvBar input.

```
If BarNumber - LSetupbar < Actvbar then Buy at BuyLevel Stop;
```

Short Entry

While within the sell setup period (as determined by the ActvBar Input), a sell stop order will be placed at SellLevel. The sell stop order will continue to be generated until the difference between the current bar number and the bar number of the setup bar (SSetupBar) meets or exceeds the value of the ActvBar input.

```
If BarNumber - SSetupbar < ActvBar then Sell at SellLevel Stop;
```

Long & Short Exits

The exits for the system are very straightforward. If MedXAvg is less than SlowXAvg, a long exit is ordered at the open of the next bar. If MedXAvg is greater than SlowXAvg, a short exit is ordered at the open of the next bar.

```
If MedXAvg < SlowXAvg then ExitLong next bar at Market;
If MedXAvg > SlowXAvg then ExitShort next bar at Market;
```

Money Management Stops

The money management stops for the system are based on the StopAmt input. For the long exit, StopAmt is divided by the value returned by the BigPointValue function, which gives a number of points that are then subtracted from the current close. The short exit is very similar—StopAmt is divided by BigPointValue and added to the current close.

```
ExitLong next bar at Close - StopAmt/BigPointValue Stop;
ExitShort next bar at Close + StopAmt/BigPointValue Stop;
```

General System Format

When we apply a system to a chart, we can use the options in the **Format** dialog box to format costs, stops, and properties. We did not enter an amount for slippage and commission, although those costs must be taken into account before a system is traded.

Note: *Remember that commissions are calculated on a per contract/share basis. When you are trading stocks, you would enter the average commission you are charged divided by the number of shares the system is buying and selling.*

Under the **Stops** tab, we enabled a money management stop of $1,200. The money management stop option holds the dollar amount per position or per contract/share we are willing to risk before exiting from the position. You should change this number according to the market you are trading and your own preference.

Note: *When you are trading stocks and you choose the stop option* ***Apply on a per share (contract) basis****, you will type in the number of points you are willing to lose before you exit. When you are trading futures or any other instrument that has a different dollar-point value, you would type the maximum number of dollars you are willing to risk per contract traded.*

In the **Properties** tab, we selected the option **Do not allow multiple entries in the same direction**. If the system is in a long position, and market conditions generate another long entry order, the order is ignored; if the system is in a short position, and market conditions generate another short entry order, the order is ignored.

Testing & Improving

We applied the Currency/Bond System to the Swiss Franc and Bonds for approximately five years of daily data, from June, 1993, to May, 1998. Using Portfolio Maximizer, we created an Individual Report and five graphs that provided insight into the system's performance.

Figure 2 is the Portfolio Maximizer Individual Report. For those of you who don't have Portfolio Maximizer, although this report contains more fields, it does include the same fields as the System Report.

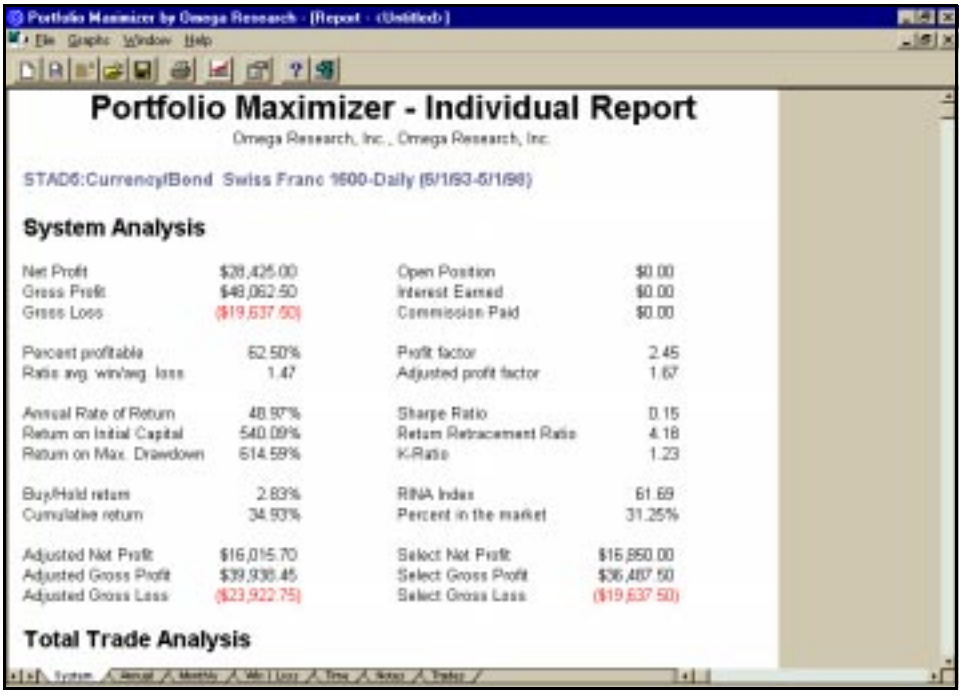


Figure 2. Portfolio Maximizer Individual Report for the Currency/Bond system

The performance results were encouraging. The system earned $28,425 per contract on 56 trades, with 62% winners and a ratio of average win to average loss of 1.47. The system made $2.45 (profit factor) for each dollar that it lost.

Most of the reward-to-risk ratios exceeded their reference points. The Return Retracement Ratio was 4.18 (compared to a guideline of 3.0 for a “good” system). Net Profit/Largest Loss was 19.44 (compared to 7.0), and Net Profit/Maximum Drawdown was 6.15 (compared to 5.0). The RINA Index was 61.69, far above its guideline of 30. Only the Sharpe Ratio (at .15) and the closely related K-Ratio (at 1.23) failed to meet their reference points of .25 and 2.5, respectively.

The Currency/Bond System Equity Curve is displayed in Figure 3.

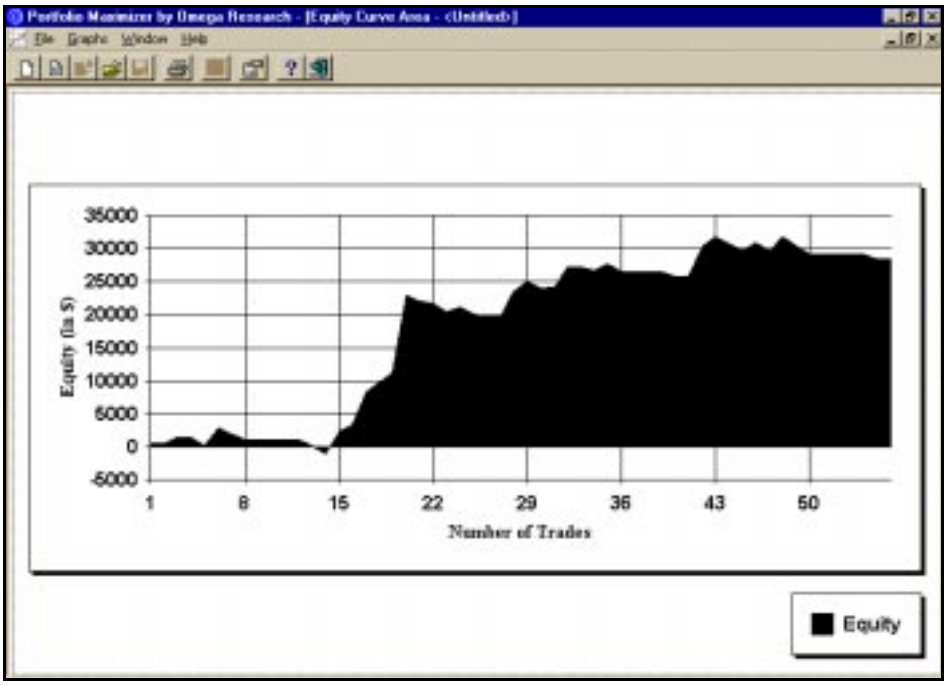


Figure 3. The System Equity curve for the Currency/Bond system

The system struggled for its first 14 trades but produced a fairly strong equity curve from trade 15 on. Figure 4 shows the Underwater Equity Curve.

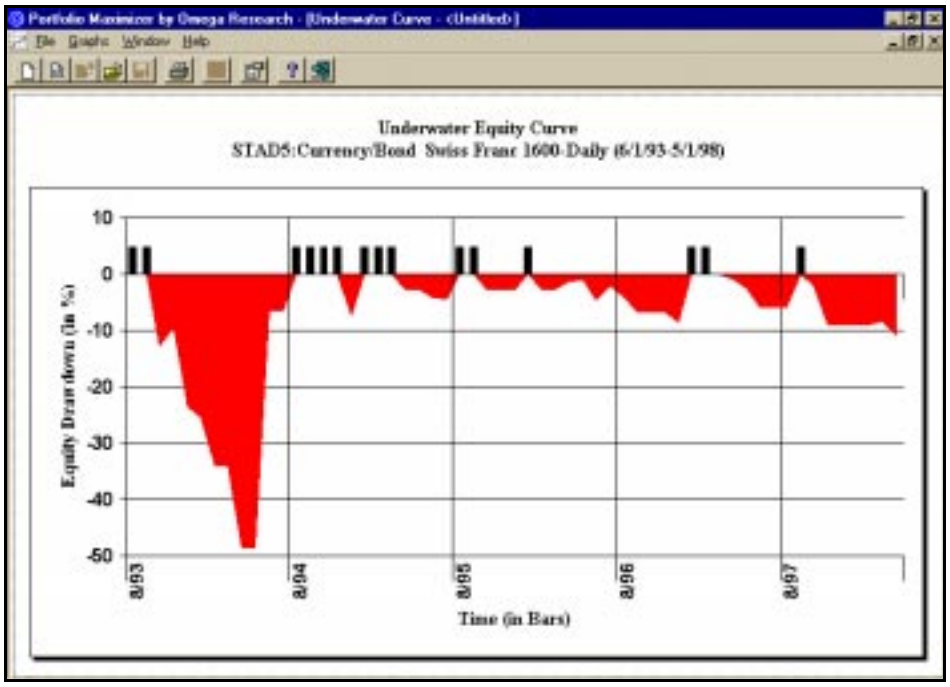


Figure 4. The Underwater Equity Curve for the Currency/Bond system

Each vertical bar rising from the zero line represents a new equity high based on monthly performance data. The negative "underwater" curves between new equity highs show the percent retracement from the previous equity high. We wouldn't want to trade this system without finding and fixing the problem that produced a decline of almost 50% between equity peaks 2 and 3. With that exception, the system performed well, holding subsequent drawdowns to only about 10%. The Average Profit By Month is graphed in Figure 5.

Figure 5. The Average Profit By Month graph for the Currency/Bond system

During our five-year test, only the months of April, May, and December failed to return a profit. December was the biggest losing month, giving back almost $5,000 of profit per contract. January and September were the best-performing months with gains of about $2,100 and $1,600 per contract, respectively.

Figure 6, Monthly Rolling Net Profit, is a different type of monthly equity graph. It shows the growth of equity month-by-month on a mark-to-market basis.

Figure 6. The Monthly Rolling Net Profit graph for the Currency/Bond System

Mark-to-market in this case means that equity is calculated at the end of each month for both open (unrealized) and closed (realized) profits. According to this graph, our system did not perform well for its first two years (ending the two-year period just a little above breakeven), but it made significant profits for the subsequent two years.

Maximum Adverse Excursion is the focus of Figure 7.

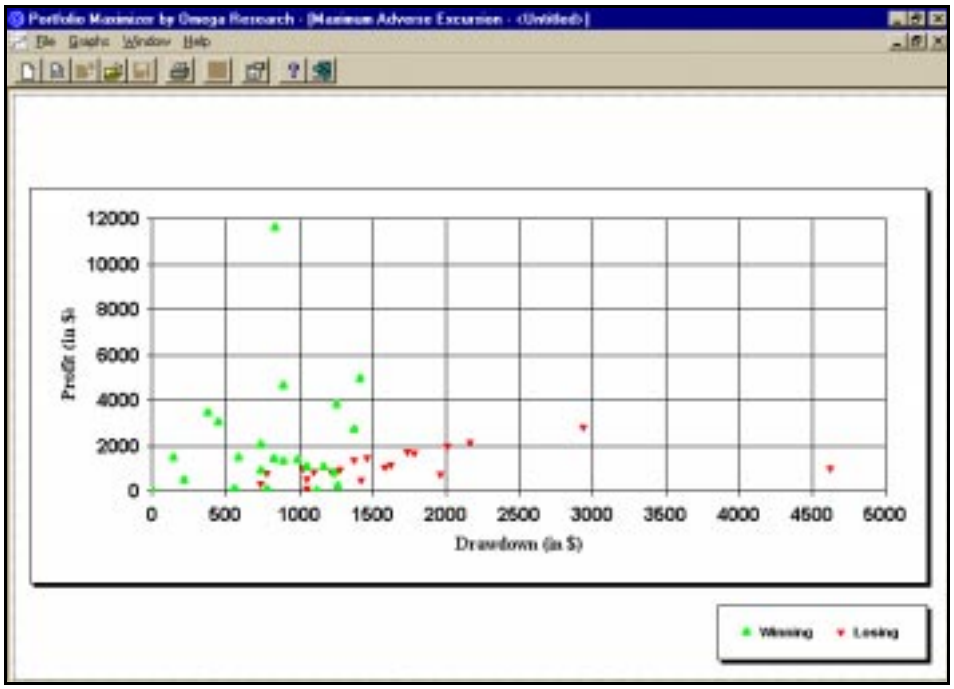


Figure 7. Maximum Adverse Excursion graph for the Currency/Bond system

Maximum Adverse Excursion represents each trade's realized profit or loss on the vertical axis and each trade's worst drawdown on the horizontal axis. The upward-pointing triangles represent winning trades; the downward-pointing arrows are losing trades. Note that there are four losing trades but no winning trades that experienced a drawdown of more than $1,500, suggesting that $1,500 might be an effective money management stop. The trades that incurred $1,500 in drawdown did not reverse and become winners, so we wouldn't risk more than $1,500 per contract for this system.

Suggestions for Improvement

Since this system performed somewhat inconsistently — struggling on its first 14 trades and its last 7 trades — we'd like to find a way to make the system more robust. A possible next step would be to add another market to the mix. The market we're thinking of is the U.S. Dollar Index. If bonds and the currency we want to buy are bullish, perhaps we should also require a bearish Dollar Index, which would suggest a broadly based weakness in the dollar. Conversely, if bonds and the currency we want to sell short are bearish, it might be beneficial to also require a bullish Dollar Index, indicating a broadly based dollar strength.

Intermarket I System

Our Intermarket I System looks for a decline in the U.S. Treasury Rate to signal a buying opportunity in the S&P. When the Treasury Rate closes below its previous close plus a long bond yield trigger of 25%, we'll buy the S&P on the next open. Once a trade is initiated, we'll remain in the position for 50 days and then exit on the close.

Defining Your Trading Rules

In Intermarket I, we defined long entries only; this is in keeping with our goal of using the Treasury Rate to signal buying opportunities in stocks. We also defined exits to exit from the long position.

Long Entries

a) We buy the S&P on the next open after the U.S. Treasury Rate closes below its previous close plus 25% of the long bond yield.

Exit Orders

a) We exit our S&P position on the close after 50 days in the trade.

Designing & Formatting

This section presents the EasyLanguage instructions and formatting for the system, with the EasyLanguage instructions broken down and explained line by line.

EasyLanguage Instructions: Intermarket One (STAD5: Intermarket 1)

```
Inputs: TNBars(50), LYldTrg(.25);

{Long Entry}
IF Close of Data2 < (Close[1] of Data2 + LYldTrg) ThenBuy ("TBond") This Bar on Close;

{Long Exit}
IF BarsSinceEntry > TNBars ThenExitLong This Bar on Close;
```

Inputs

Following is the list of all the inputs we used in this system:

Input	Default	Description
TNBars	50	The number of bars during which we will remain in the trade.
LYldTrg	.25	The percentage of the yield of the U.S. Treasury Bond to be used to trigger the buy order (.25 = 25%).

There are no variables defined for this system.

Long Entry

The system will place a buy order on the close if the yield of the U.S. Treasury Bond for the current bar is less than the yield for the previous bar plus 25% (LYldTrg). When the criteria are met, the buy order “TBond” is placed for the close of the current bar.

```
IF Close of Data2 < (Close[1] of Data2 + LYldTrg) ThenBuy ("TBond") This Bar on Close;
```

Long Exit

The exit will be placed after we have been in a long position for 50 bars (TNBars). At 50 bars, we place a long exit order on the close of the current bar.

```
IF BarsSinceEntry > TNBars Then ExitLong This Bar on Close;
```

General System Format

When we apply a system to a chart, we can use the options in the **Format** dialog box to format costs, stops, and properties. We did not enter an amount for slippage and commission, although those costs must be taken into account before a system is traded.

Note: *Remember that commissions are calculated on a per contract/share basis. When you are trading stocks, you would enter the average commission you are charged divided by the number of shares the system is buying and selling.*

We did not enable any stops for this system.

In the **Properties** tab, we selected the option **Do not allow multiple entries in the same direction**. If the system is in a long position, and market conditions generate another long entry order, the order is ignored.

Testing & Improving

We applied Intermarket I to a daily chart of the S&P Index from January, 1988, to September, 1998. Using Portfolio Maximizer, we created an Individual Report and six graphs that provided insight into the system's performance.

Figure 1 is the Portfolio Maximizer Individual Report.

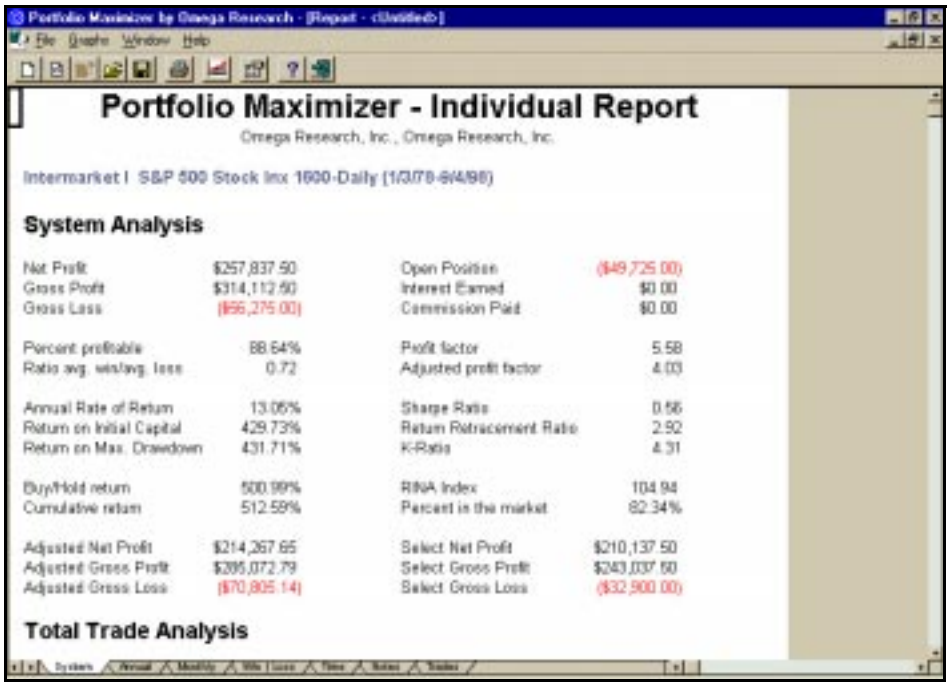


Figure 1. Portfolio Maximizer Individual Report the Intermarket I system

It shows that our system generated 132 trades, of which 88% were profitable. The profit factor (the number of dollars gained for each dollar lost) was excellent at $5.58.

The Sharpe Ratio, K-Ratio, and RINA Index all surpassed the guidelines for a good system. The Sharpe Ratio was .56 compared to a guideline of .25, the K-Ratio was 4.31 compared to a guideline of 2.5, and the RINA Index was 104.94 compared to a guideline of 30.0. Only the Return Retracement failed to meet or exceed our reference point (2.92 compared to a guideline of 3.0).

Figure 2 is the Equity Curve. It depicts the system's performance on a trade-by-trade basis. The Equity Curve rises slowly but surely through trade number 99 and then skyrockets through trade 132.

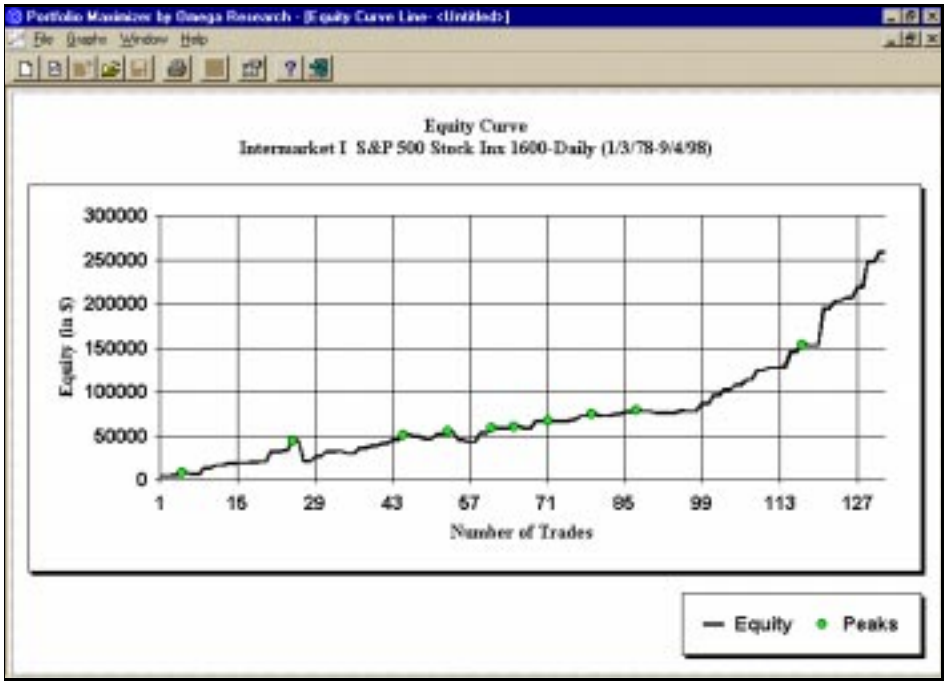


Figure 2. Equity Curve for the Intermarket 1 system

Figure 3, Monthly Net Profit, shows the system's profit or loss for each month of the test period.

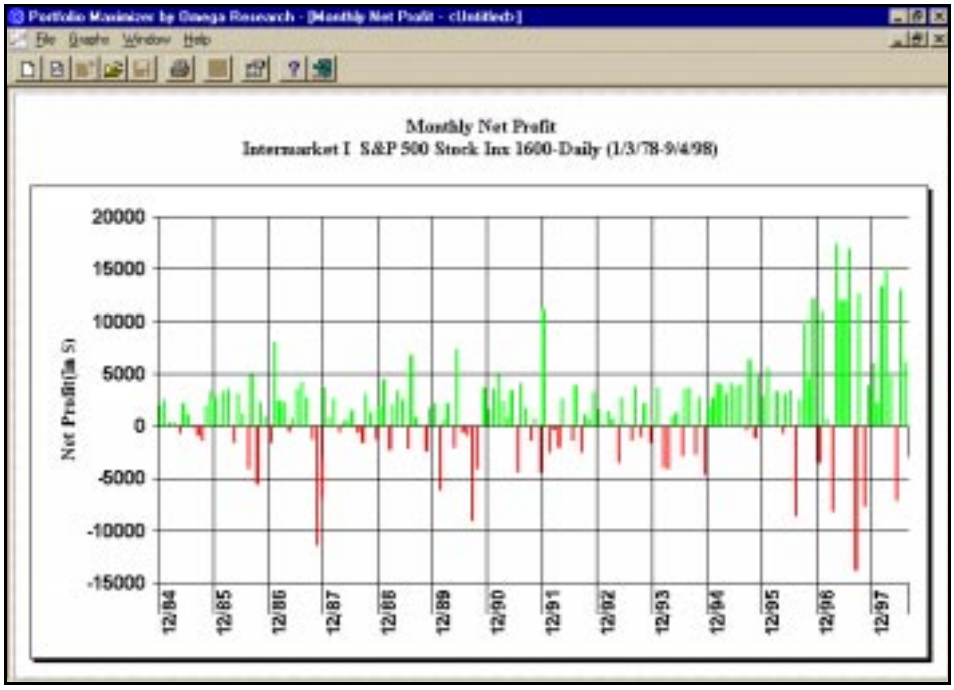


Figure 3. Monthly Net Profit graph for the Intermarket 1 system

Figure 4 displays the Average Profit By Month. Only the months of August and October did not yield a profit over the approximately 10-year period.

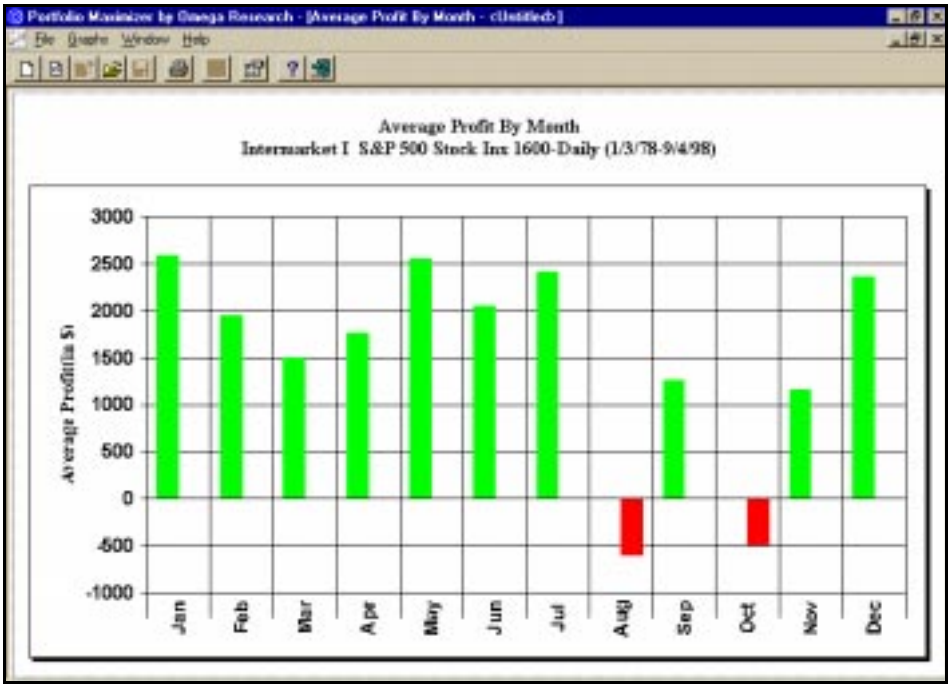


Figure 4. Average Profit By Month graph for the Intermarket 1 system

Figure 5 presents Total Trades. The two winning outlier trades on this graph are its most significant feature. We've defined outliers as trades that are more than three standard deviations above or below the average trade. Because Intermarket I is a trend-following system, we weren't too surprised that a large percentage of the total profit was won on a few outliers.

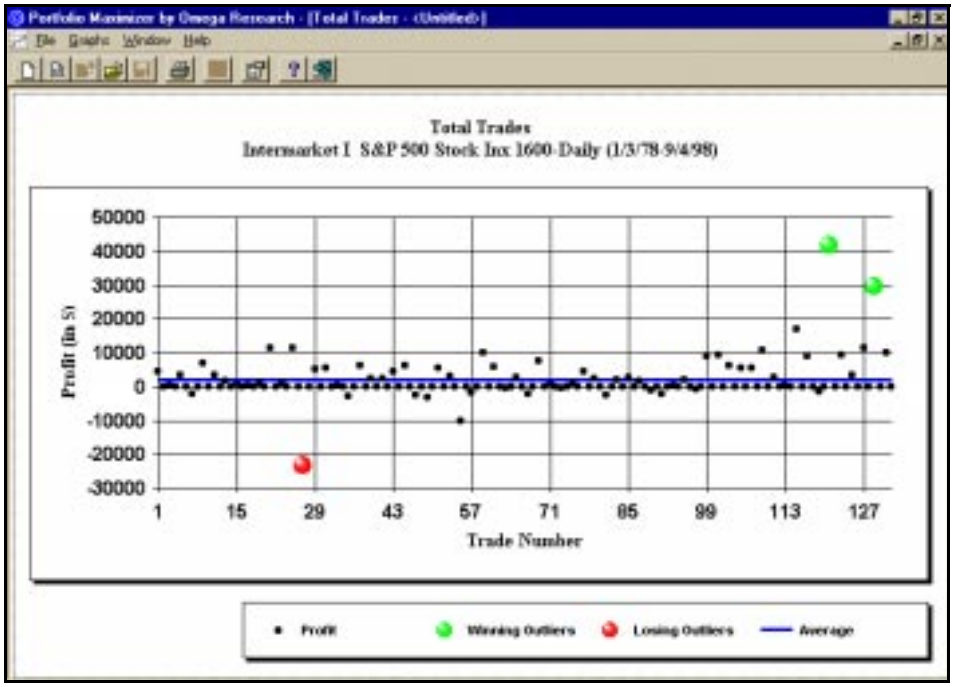


Figure 5. Total Trades graph for the Intermarket 1 system

Figures 6 and 7 show Maximum Adverse Excursion and Maximum Favorable Excursion, respectively.

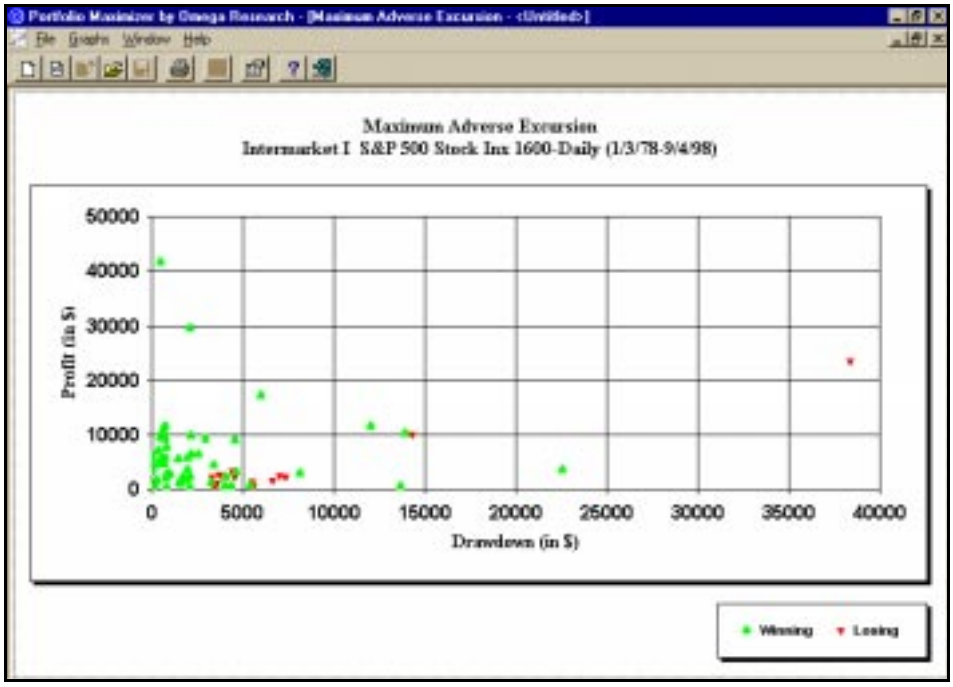


Figure 6. Maximum Adverse Excursion graph for the Intermarket 1 system

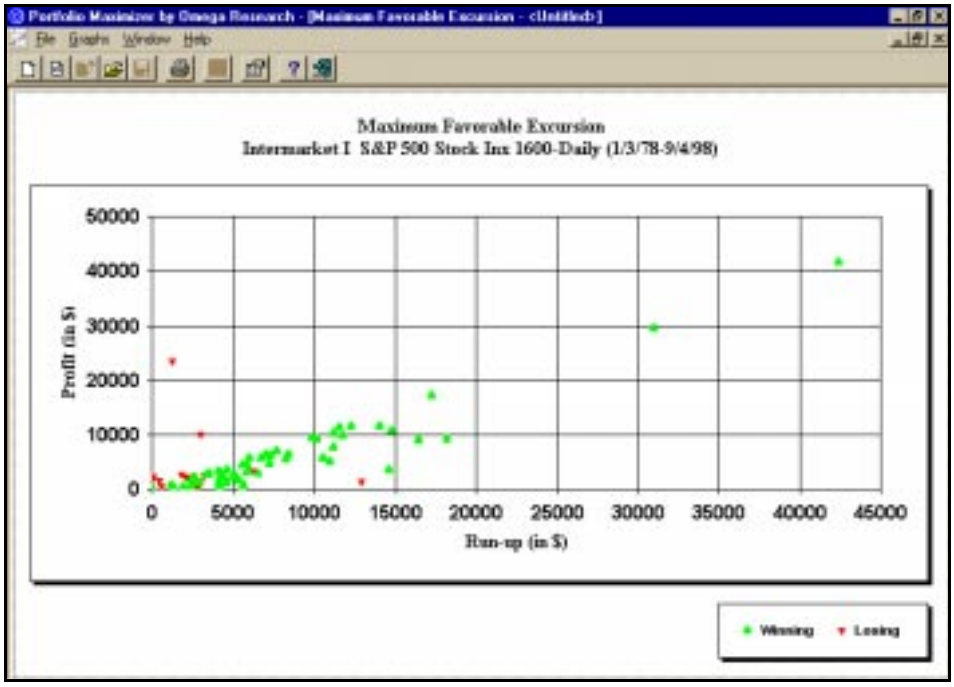


Figure 7. Maximum Favorable Excursion graph for the Intermarket 1 system

One useful bit of information in Figure 6 is that only one winning trade (a small winner at that) suffered a drawdown of more than $20,000 while the largest losing trade had a drawdown of

about $38,000. An initial protective stop would have to be enabled for this system to become tradeable.

Figure 7, Maximum Favorable Excursion, can be used to help us determine appropriate trailing stops. The vertical scale measures each trade's profit or loss in dollars. The upward-pointing triangles (green in Portfolio Maximizer) represent winning trades, and the downward-pointing triangles (red in Portfolio Maximizer) represent losing trades. The horizontal axis measures each trade's run-up. A study of a system's Maximum Favorable Excursion performance will enable us to set trailing stops that capture most of the winning trades while also limiting the system's exposure to profit erosion.

Suggestions for Improvement

Intermarket I was designed with no stops and an automatic exit after 50 days in the trade. The system could almost certainly be improved with strategic placement of initial and trailing stops and an exit condition based on market activity rather than a fixed number of days.

Intermarket II System

Falling interest rates are usually bullish for stocks, and rising interest rates are usually bearish for stocks. The 30-year U.S. Treasury Bond can often serve as a leading indicator for stocks. Of course, no indicator or correlation works perfectly in the financial markets. A bullish bond market (lower interest rates) doesn't guarantee a bullish stock market; however, it would be very unusual to see a bullish stock market without a strong bond market.

Our Intermarket II system seeks to profit from the positive correlation between stocks and bonds. When the U.S. Federal Reserve lowers interest rates, we'll go long the S&P futures. Like the previous system, we will remain in the trade for a certain number of days, in this case 100, before we exit at the close.

Defining Your Trading Rules

In this system, we defined long entries and exits. The entries and exits are defined next.

Long Entries

a) When the close of the prime rate is less than it was one day ago, the prime rate steps variable increases by one. If the close of the prime rate is greater than it was one day ago, the prime rate steps variable equals zero.

b) When the prime rate steps variable is equal to or greater than the prime step trigger, and the prime steps variable is greater than it was one day ago, then buy the S&P futures contract on the next open.

Exit Orders

a) We will exit from the long position at the close once we have held the trade for 100 days.

Designing & Formatting

This section presents the EasyLanguage instructions and formatting for the system, with the EasyLanguage instructions broken down and explained line by line.

EasyLanguage Instructions: Intermarket Two (STAD5: Intermarket 2)

```
Inputs: PNbars(100), PStepTrg(1);
Vars: prSteps(0);
```

{Buy Setup}

```
IF Close of Data3 < Close[1] of Data3 ThenprSteps = prSteps + 1
Else
        IF Close of Data3 > Close[1] of Data3 ThenprSteps = 0;
```

{Long Entry}

```
IF prSteps >= PStepTrg AND prSteps > prSteps[1] ThenBuy ("PrimeCut") This Bar on Close;
```

{Long Exit}

```
IF BarsSinceEntry > PNBars ThenExitLong This Bar on Close;
```

Inputs

Following is the list of all the inputs we used in this system:

Input	Default	Description
PNBars	100	The number of bars during which we will remain in the trade.
PstepTrg	1	Number of Prime Rate Easings (cuts) that must occur.

In addition to these inputs, we define the following variable:

```
Vars:  prSteps(0);
```

Setup

The prSteps variable is incremented by one each time the prime rate is eased or cut (measured by comparing the current close of Data3 to the close of Data3 of the previous bar). If the prime rate is increased, the prSteps variable is reset to zero (0).

```
IF Close of Data3 < Close[1] of Data3 thenprSteps = prSteps + 1
Else
        IF Close of Data3 > Close[1] of Data3 ThenprSteps = 0;
```

Long Entry

We want to generate a long entry when the prime rate is lower than its previous value. The setup instructions increment the variable prSteps each time an easing of the prime rate occurs. Then, we measure the value of prSteps against the input PstepTrg. This input is the number of easings we require before placing a long order. We also requiring that prSteps be greater than the same value on the previous bar, this ensures that the long entry is not repeated for the same prime rate reduction. When both criteria are met, we place the buy order “PrimeCut” on the close of the current bar.

```
IF prSteps >= PStepTrg AND prSteps > prSteps[1] ThenBuy ("PrimeCut") This Bar on Close;
```

Long Exit

We will exit from our long position after we have held it for more than 100 bars (PNBars). When this occurs, we will place a long exit order on the close of the current bar.

```
IF BarsSinceEntry > PNBars ThenExitLong This Bar on Close;
```

General System Format

When we apply a system to a chart, we can use the options in the **Format** dialog box to format costs, stops, and properties. We did not enter an amount for slippage and commission, although those costs must be taken into account before a system is traded.

Note: *Remember that commissions are calculated on a per contract/share basis. When you are trading stocks, you would enter the average commission you are charged divided by the number of shares the system is buying and selling.*

Under the **Stops** tab, we enabled a $4,500 money management stop. The money management stop option holds the dollar amount per position or per contract/share we are willing to risk before exiting from the position. You should change this number according to the market you are trading and your own preference.

Note: *When you are trading stocks and you choose the stop option* ***Apply on a per share (contract) basis****, you will type in the number of points you are willing to lose before you exit. When you are trading futures or any other instrument that has a different dollar-point value, you would type the maximum number of dollars you are willing to risk per contract traded.*

In the Properties tab, we selected the option **Do not allow multiple entries in the same direction**. If the system is in a long position, and market conditions generate another long entry order, the order is ignored.

Testing & Improving

We applied Intermarket II to a daily chart of the S&P futures contract from January, 1978, to September, 1998. Using Portfolio Maximizer, we created an Individual Report and five graphs that provided insight into the system's performance.

Figure 1 is the Portfolio Maximizer Individual Report.

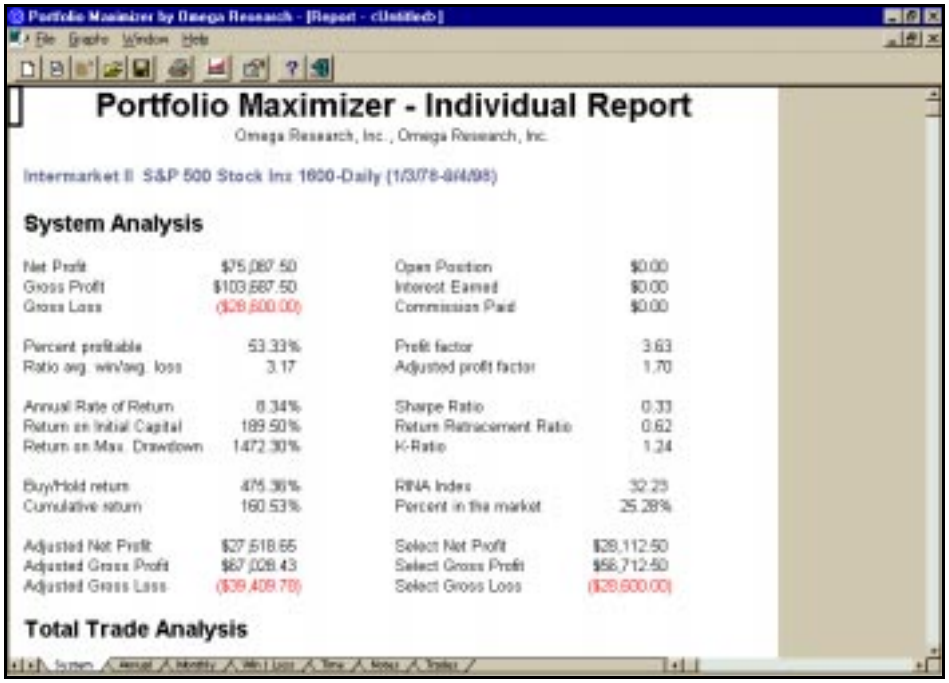


Figure 1. Portfolio Maximizer Individual Report for the Intermarket II system

It shows that our system generated 15 trades in our test period of about 20.5 years. The winning percentage was 53, the ratio of average win to average loss was 3.17, and the profit factor (the number of dollars won per dollar lost) was 3.63. The system earned a net profit of more than $75,000 on an average trade of $5,005.

Most of the performance rates we're featuring in this volume were very good: the Sharpe Ratio was .33 (compared to a guideline of .25), and the RINA Index was 32 (compared to a guideline of 30). Net Profit/Largest Loss was 14.72 (guideline is 7.0), and Net Profit/Maximum

Drawdown was 14.72 (guideline is 5.0). One ratio that did not meet our reference point was the Return Retracement Ratio, which was .62 compared to a reference point of 3.0.

Figure 2 graphs Total Trades.

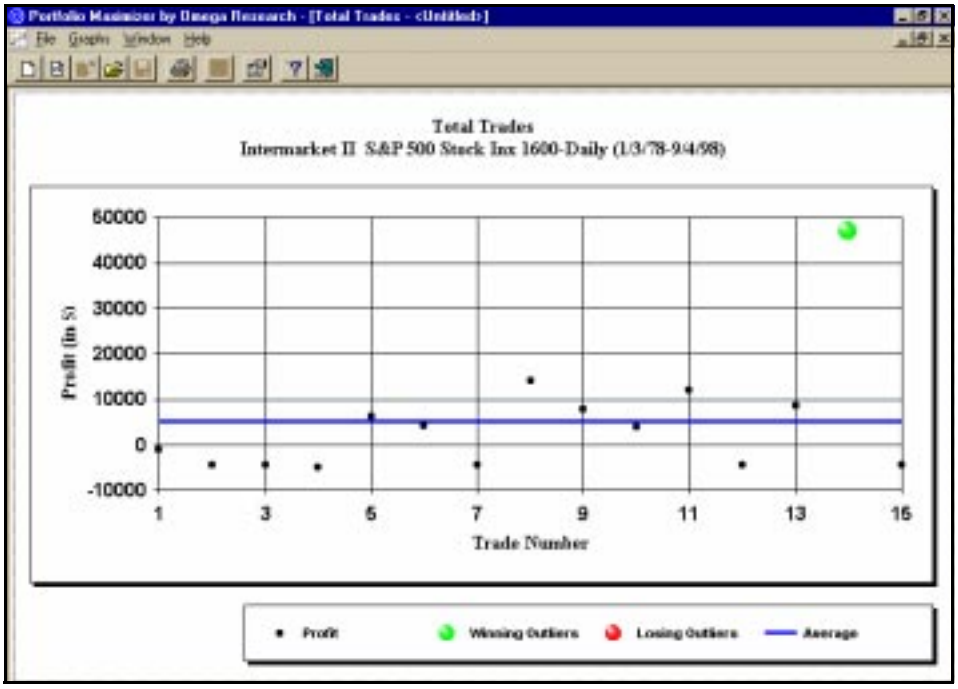


Figure 2. Total Trades graph for the Intermarket II system

Note the Winning Outlier (a trade more than three standard deviations from average). It's the one trade that really made the system work well.

On one hand, we don't like our systems to be too dependent on one huge winner, but, on the other hand, that's how most good trend-following systems perform. We win a little and lose a little, staying in the game, waiting to hit the homerun. In fact, the key to making money as a trend follower is risk management: the small wins and small losses approximately cancel each other out, we suffer no or very few large losses, and our large winners constitute our net profit.

To eliminate large losses, we suggest the following: (1) trade in liquid markets only, not thinly traded markets, (2) avoid holding position through major reports, and (3) place logical protective stops. Although these suggestions may not completely eliminate large losses, they will at least minimize the number of them.

Intermarket II's trade-by-trade Equity Curve is displayed in Figure 3.

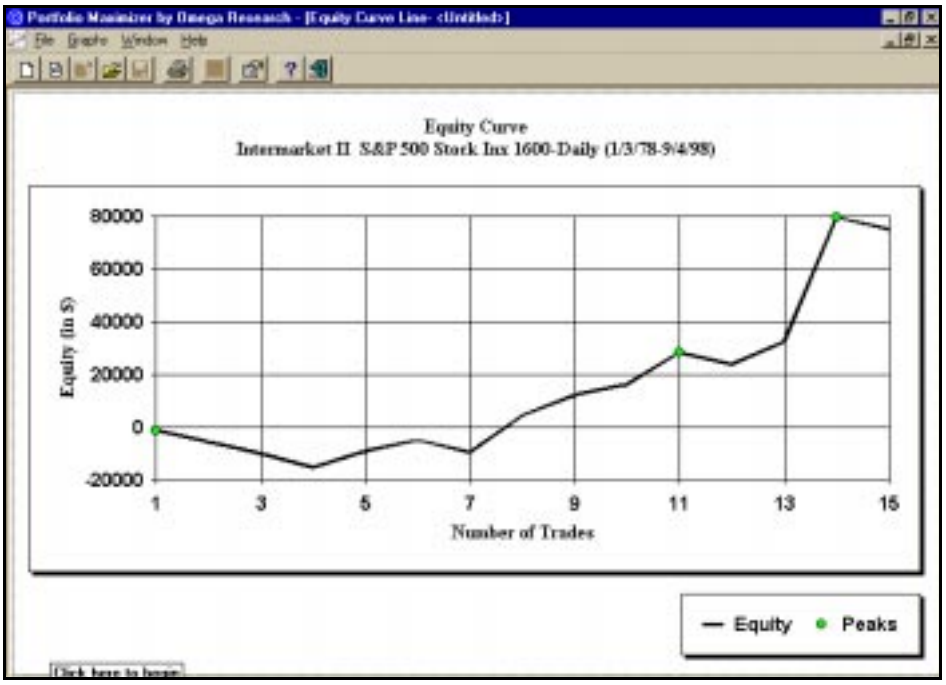


Figure 3. Equity Curve for the Intermarket II system

The system performed poorly during its first seven trades but did very well from trade 8 to trade 14.

Figure 4 shows the Monthly Rolling Net Profit.

Figure 4. Monthly Rolling Net Profit graph for the Intermarket II system

The Monthly Rolling Net Profit performs a mark-to-market calculation at the end of each month, including both realized (closed) and unrealized (open) profits. The graph depicts the progression of equity in dollars by month. We're not pleased with the system's three-year flat period from May, 1988, to May, 1991, and we'd like to see stronger performance from May, 1993, to May, 1998, during a major bull market in stocks.

Average Profit By Month is graphed in Figure 5.

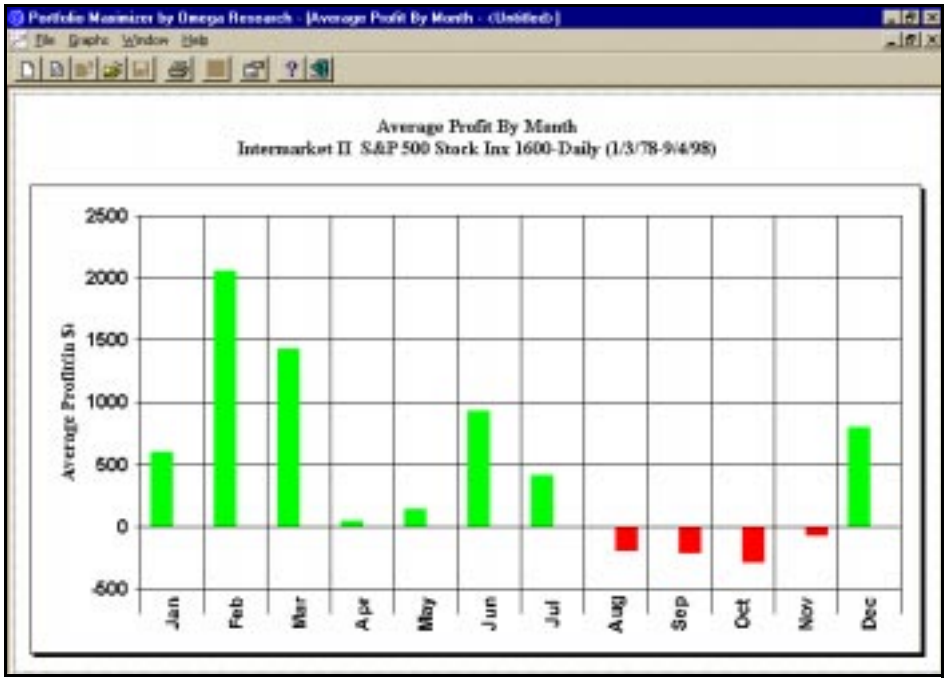


Figure 5. Average Profit by Month graph for the Intermarket II system

It shows the system's individual monthly return in dollars, thus serving as a general seasonal overview of trading performance. In this case, Average Profit By Month indicates that the system lost money on average in August, September, October, and November from 1978 through August of 1998. An idea at least worthy of consideration and more research is to avoid trading the system during the four months that have lost money historically.

Intermarket II's drawdown is shown trade by trade in Figure 6.

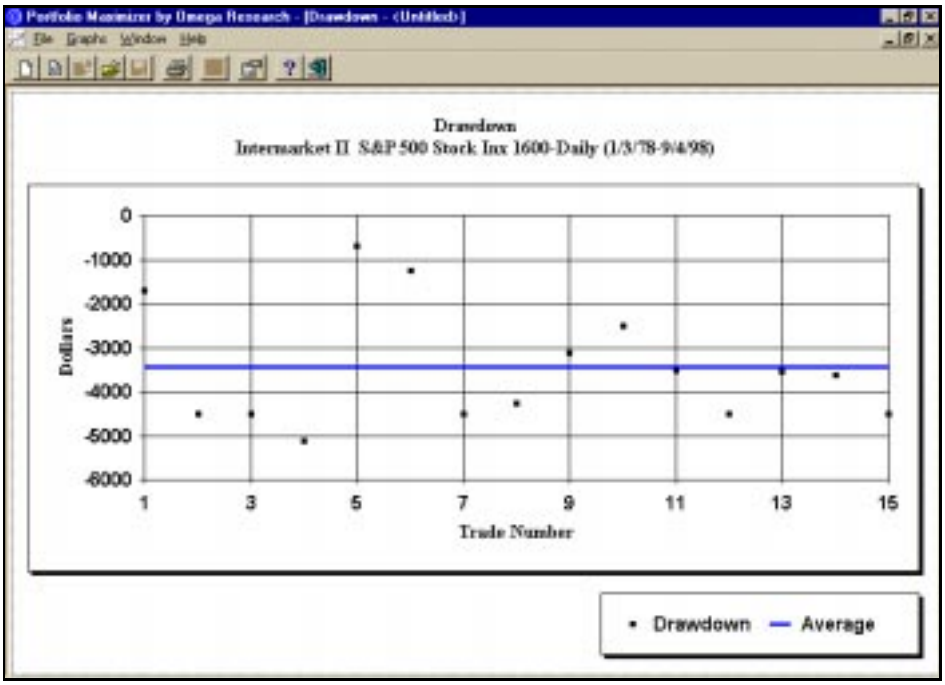


Figure 6. Drawdown graph for the Intermarket II system

Note that all 15 trades experienced at least some drawdown and that this average drawdown was about $3,500. That information should be useful in specifying initial protective stops and trailing stops.

Suggestions for Improvement

We can make two suggestions that might improve the performance of Intermarket II. First, we would consider raising the number of consecutive easings of the Prime Rate from one to two or three. Our test results were based on one easing, but requiring two or three (while reducing the number of trades) might yield higher profits. Second, we would consider not initiating new positions during the months of August September, October, and November — months in which the system has lost money historically. An alternative to not trading the system during those four months would be to increase the required number of easings from one to two or from two to three in that time period.

CHAPTER 3

Support & Resistance Systems

Support & Resistance systems are designed for sideways or directionless markets, and they typically have the following attributes:

- They buy low and sell high in an attempt to take advantage of the sideways price movement characterizing directionless markets.
- They have a high number of winning trades, with small profits on each trade. They sell as the market goes higher and take small losses until the market finally turns down and results in a profitable trade.
- They are easier to trade emotionally.

By design, these systems miss the big move—they usually have small profits and larger losses as markets trend. The system keeps shorting a market that is in an uptrend or buying a market that is in a downtrend. Therefore, when traders use Support &Resistance systems, they use them within a group of systems that also includes trending systems and perhaps one or more volatility systems.

In this chapter, we present four Support & Resistance systems, designed to make the most of the sideways movement of a market.

In This Chapter

Fibonacci .382 Retracement System

In the 13th century, Leonardo Fibonacci, an Italian mathematician, wrote a very influential book called *Liber Abaci* (Book of Calculations). One small section of the book described an intriguing number sequence. The sequence was know to the ancient Greeks and Egyptians, but because Fibonacci popularized the number sequence in his book, we refer to it today as the Fibonacci numbers.

The sequence begins 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144 and extends to infinity. Each number after 1 is the sum of the two preceding numbers. Fibonacci numbers yield important ratios that are prevalent in art, architecture, nature, and many other areas.

The ratios of alternate numbers fluctuate narrowly around 2.618 or .382, which is its inverse. Many traders believe Fibonacci numbers can be used in technical analysis of the financial markets. In our Fibonacci system, we'll use .382 (38.2%) as a measure of price retracement, buying after a .382 downward correction in an uptrend and selling short after a .382 upward correction in a downtrend. Figure 1 shows this system applied to a daily chart. Also applied is the Fibonacci Retracement indicator (STAD 5: Fibonacci Ret) we wrote to go along with the system.

Figure 1. Fibonacci .382 Retracement system applied to a daily Gold chart

Defining Your Trading Rules

In this system, we defined both long and short entries as well as exit orders. We also did some setup work for an exponential moving average, the 50-bar high-low range, and the .382 retracements. The setups, entries, and exits are defined next.

Setup

a) Calculate a 15-bar exponential moving average (EMA) of closes.

b) Calculate the high-low range of the last 50 bars and .382 retracements from the high and low.

Long Entries

a) Check for a close at or below the .382 retracement level.

b) Check for the 15-bar EMA to be greater than it was one bar ago or for a close above the EMA.

c) Buy on the next open.

Short Entries

a) Check for a close at or above the .382 retracement level.

b) Check for 15-bar EMA to be less than it was one bar ago or for a close below the EMA.

c) Sell short on the next open.

Exit Orders

a) Once we enter a position, we'll set a money management stop.

b) We'll also set a $ Risk trailing stop to lock in profits while also attempting to let profits run.

Designing & Formatting

This section presents the EasyLanguage instructions and formatting for the system, with the EasyLanguage instructions broken down and explained line by line.

EasyLanguage Instructions: Fibonacci .382 Retracement (STAD5: Fibonacci)

```
Inputs: Price(Close), XAvgLen(15), HiLoLen(50), Retrace(.382), SetUpLen(24);
Vars: XAvg(0), HiHi(0), HiHiBar(0), LoLo(0), LoLoBar(0), Retracement(0), BuySetup(0), SellSetup(0);

{Calculation of the necessary values, and assignment to variables}
XAvg = XAverage(Price, XAvgLen);
HiHi = Highest(High, HiLoLen);
LoLo = Lowest(Low, HiLoLen);
HiHiBar = HighestBar(High, HiLoLen);
LoLoBar = LowestBar(Low, HiLoLen);
Retracement = (HiHi - LoLo) * Retrace;

{Conditions for a Buy Setup}
IF Close <= HiHi - Retracement AND HiHiBar < LoLoBar Then BuySetup = 0;
{Conditions for a Sell Setup}
IF Close >= LoLo + Retracement AND HiHiBar > LoLoBar ThenSellSetup = 0;

{Accumulates to count the number of bars in the setups}
BuySetup = BuySetup + 1;
SellSetup = SellSetup + 1;

{Buy Criteria Evaluation}
IF BuySetup <= SetUpLen Then Begin
        IF SwingLow(1, XAvg, 1, 2) <> -1 OR Close > XAvg Then Begin
                Buy Next Bar at Market;
                BuySetup = SetUpLen;
        End;
End;
```

```
{Sell Criteria Evaluation}
IF SellSetup <= SetUpLen Then Begin
        IF SwingHigh(1, XAvg, 1, 2) <> -1 OR Close < XAvg Then Begin
                Sell Next Bar at Market;
                SellSetup = SetUpLen;
        End;
End;
```

Inputs

Following is the list of all the inputs we used in this system:

Input	Default	Description
Price	Close	Price used to calculate the exponential moving average.
XAvgLen	15	Length, expressed in bars, used to calculate the exponential moving average.
HiLoLen	50	Length, expressed in bars, used in the calculation of the Highest High and the Lowest Low.
Retrace	.382	Percent Retracement used by the highest high and lowest low respectively (the .382 value represents 38.2%).
SetupLen	24	The Setup Length. A length, expressed in bars, of the Buy and Sell setups, within which an entry may occur.

In addition to these inputs, we define the following variables:

```
Vars: XAvg(0), HiHi(0), HiHiBar(0), LoLo(0), LoLoBar(0), Retracement(0), BuySetup(0),
SellSetup(0);
```

Setup

We calculate and assign to variables key values that will be referenced throughout the system. These values include: XAvg, the exponential moving average; HiHi, the highest high; LoLo, the lowest low; HiHiBar, the highest bar high (returns the number of bars since the most recent highest high); and LoLoBar, the lowest bar low (returns the number of bars since the most recent lowest low).

```
XAvg = XAverage(Price, XAvgLen);
HiHi = Highest(High, HiLoLen);
LoLo = Lowest(Low, HiLoLen);
HiHiBar = HighestBar(High, HiLoLen);
LoLoBar = LowestBar(Low, HiLoLen);
```

Once the basic variables have been assigned, we subtract the LoLo from HiHi (total range) and multiply it by the Retrace value, as specified in the inputs. This value is then assigned to the variable Retracement, which determines the retracement value that will be used on each bar (this value is recalculated on each bar).

```
Retracement = (HiHi - LoLo) * Retrace;
```

For a buy setup, we are looking for a 38.2% pullback from an upward trend. This means two conditions must be met for the initiation of the buy setup. First, the close must be less than or equal to the HiHi minus the retracement value. Second, HiHiBar must be less than LoLoBar. This means that the highest high must have occurred after the lowest low, thus indicating an up-trend prior to the pullback. If both conditions are met, the BuySetup variable is assigned a value of zero, effectively resetting the setup.

```
IF Close <= HiHi - Retracement AND HiHiBar < LoLoBar Then BySetup = 0;
```

The sell setup is just the opposite of the buy setup—we are looking for a 38.2% retracement from a downward trend. Once again, two conditions must be met for the initiation of the sell setup. First, the close must be greater than or equal to the LoLo plus the Retracement value. Second, HiHiBar must be greater than LoLoBar. This means that the lowest low must have occurred after the highest high, thus indicating a downward trend prior to the retracement. If both conditions are met, the SellSetup variable is assigned a value of zero, effectively resetting the setup.

```
IF Close >= LoLo + Retracement AND HiHiBar > LoLoBar Then SellSetup = 0;
```

The two counters, BuySetup and SellSetup, are incremented on each bar in order to keep track of the number of bars in the buy/sell setups.

```
BuySetup = BuySetup + 1;
SellSetup = SellSetup + 1;
```

Long Entry

If the current bar is within the allowable BuySetup period (as defined by the SetUpLen input), we will look for a resumption of the prior upward trend. The system requires that one of two criteria be true in order for a buy signal to be triggered.

The first is the occurrence of a swing low (low pivot) of the exponential moving average of the close (represented by the Price input). If a swing low does not exist, the SwingLow function will return a "-1", otherwise, it will return the value of the pivot (a value not equal to '<>' –1). Our only concern is that the swing low value not equal "-1". The second criteria is a close that is greater than the exponential moving average. If either of these two criteria are met, a buy signal will be generated on the open of the next bar. In addition, the BuySetup variable will be set equal to SetUpLen, thus nullifying the remainder of the original setup period.

```
IF BuySetup <= SetUpLen Then Begin
        IF SwingLow(1, XAvg, 1, 2) <> -1 OR Close > XAvg Then Begin
                Buy Next Bar at Market;
                BuySetup = SetUpLen;
        End;
End;
```

Short Entry

If the current bar is within the allowable SellSetup period (as defined by the SetUpLen input), we will look for a resumption of the prior downward trend. The system requires that one of two criteria be true in order for a sell signal to be triggered. The first is the occurrence of a swing high (high pivot) of the exponential moving average. If a swing high does not exist, the SwingHigh function will return a "-1," otherwise, it will return the value of the pivot (a value not equal to '<>' –1). Again, our only concern is that the swing high value not equal –1. The second criteria is a close that is less than the exponential moving average of the close (represented by the Price input). If either of these two criteria are met, a sell signal will be generated on the open of the next bar. In addition, the SellSetup variable will be set equal to SetUpLen, thus nullifying the remainder of the original setup period.

```
IF SellSetup <= SetUpLen Then Begin
        IF SwingHigh(1, XAvg, 1, 2) <> -1 OR Close < XAvg Then Begin
                Sell Next Bar at Market;
                SellSetup = SetUpLen;
        End;
End;
```

General System Format

When we apply a system to a chart, we can use the options in the **Format** dialog box to format costs, stops, and properties. We did not enter an amount for slippage and commission, although those costs must be taken into account before a system is traded.

Note: *Remember that commissions are calculated on a per contract/share basis. When you are trading stocks, you would enter the average commission you are charged divided by the number of shares the system is buying and selling.*

Under the **Stops** tab, we enabled a $200 money management stop and a $ Risk Trailing stop at $400. The money management stop option holds the dollar amount per position or per contract/ share we are willing to risk before exiting from the position. The $ Risk trailing stop option allows us to indicate how much of the maximum open profit we are willing to give back before the position is automatically closed out.

Note: *When you are trading stocks and you choose the stop option* ***Apply on a per share (contract) basis****, you will type in the number of points you are willing to lose before you exit. When you are trading futures or any other instrument that has a different dollar-point value, you would type the maximum number of dollars you are willing to risk per contract traded.*

In the **Properties** tab, we selected the option **Do not allow multiple entries in the same direction**. If the system is in a long position, and market conditions generate another long entry order, the order is ignored; if the system is in a short position, and market conditions generate another short entry order, the order is ignored.

Testing & Improving

We tested the Fibonacci .382 Retracement system on Gold futures from June, 1993 to July, 1998. The system generated 85 trades, of which 35% were profitable, as shown in Figure 2.

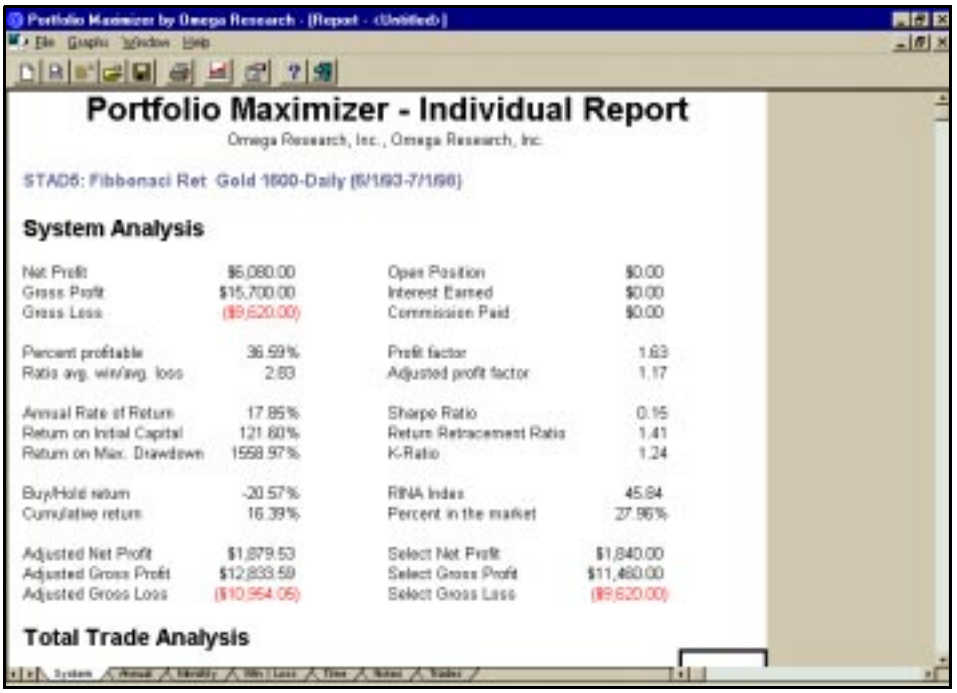


Figure 2. Portfolio Maximizer Individual Report for the Fibonacci .382 Retracement system applied to the daily Gold chart

The ratio of average win to average loss was 2.76, and the profit factor (the number of dollars won per dollar lost) was 1.51.

Four of the ratios we're emphasizing in this volume (Sharpe Ratio, Return Retracement, K-Ratio, and RINA Index) performed below our reference points (our guidelines for a good system).

Two other ratios we're emphasizing surpassed our reference points, however. Net Profit/Largest Loss was 12.85 (compared to a reference point of 7.0), and Net Profit/Maximum Drawdown was also 12.85 (compared to a guideline of 5.0).

The Equity Curve for our Fibonacci .382 Retracement System is displayed in Figure 3.

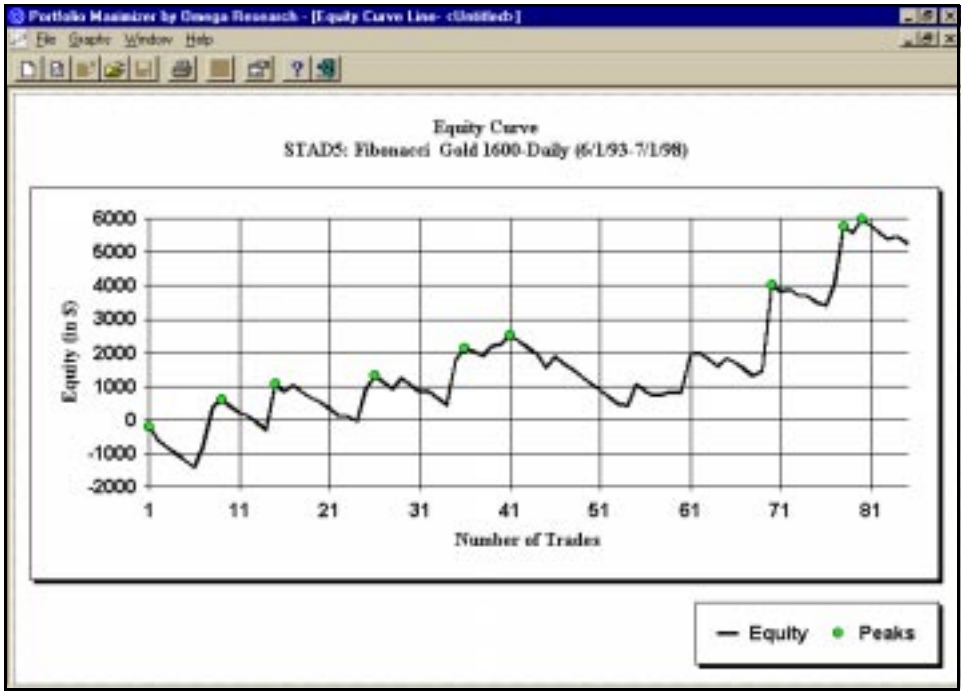


Figure 3. Equity curve for the Fibonacci .382 Retracement system

After a disappointing beginning, performance improved, and the system earned about $6,000 in profit by trade number 81.

Figure 4 depicts the Monthly Rolling Net Profit.

Figure 4. Monthly Rolling Net Profit graph for the Fibonacci .382 Retracement system

The Monthly Rolling Net Profit is the growth of equity on a mark-to-market basis from month end to month end. With the exception of the drawdown between September, 1995, and July, 1996, the graph looks good.

Monthly Net Profit is the focus of Figure 5.

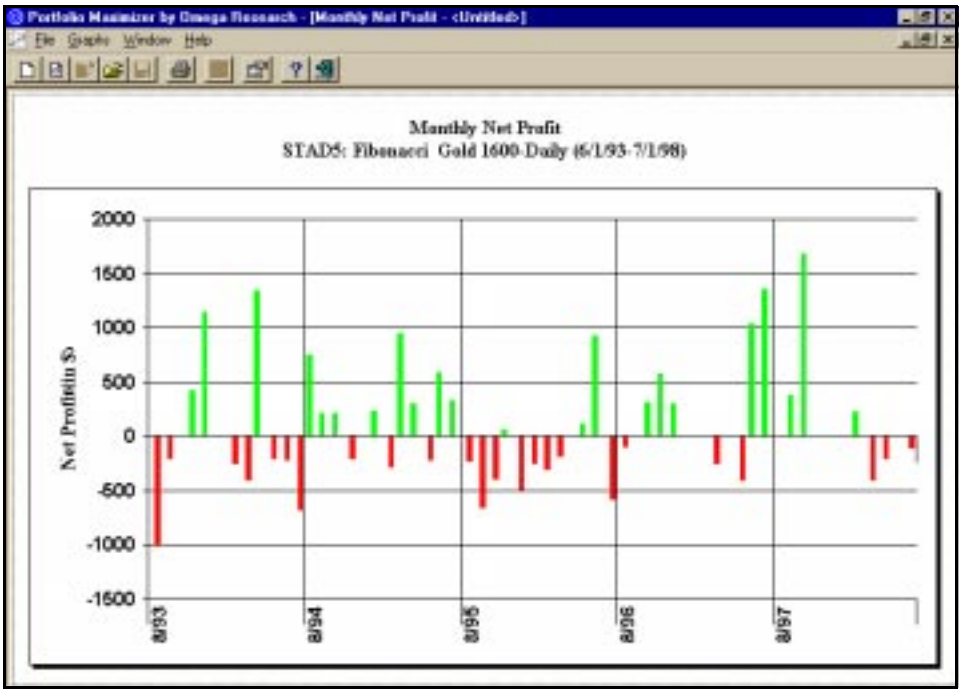


Figure 5. Monthly Net Profit graph for the Fibonacci .382 Retracement system

It shows that the system produced 22 winning months and 24 losing months. The period from August, 1995, to August, 1996, resulted in only three winning months and eight losing months, accounting for our equity drawdown.

Figure 6 graphs Maximum Adverse Excursion.

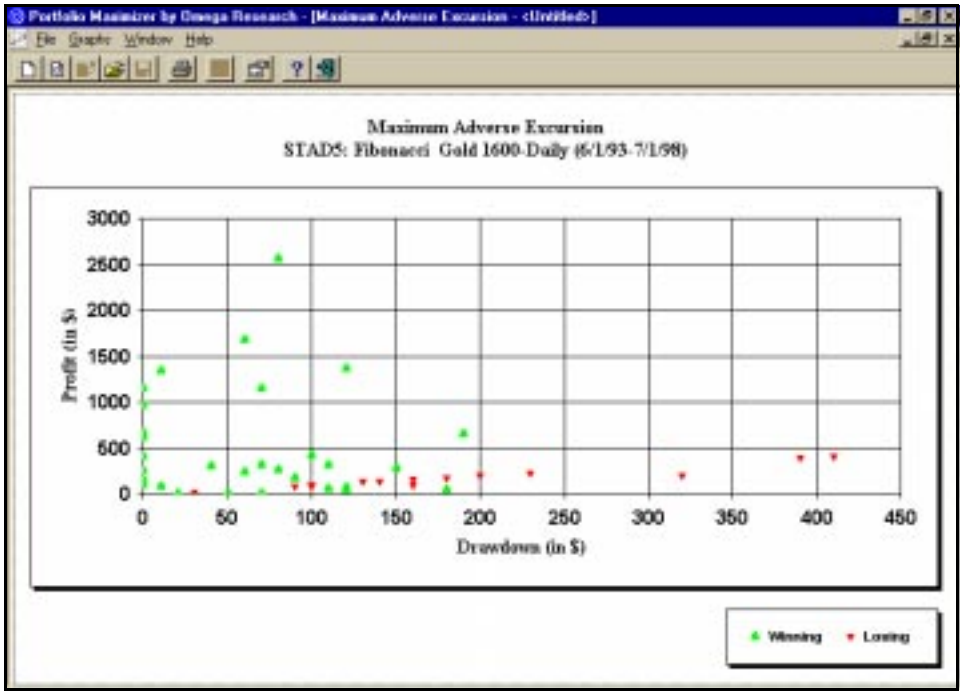


Figure 6. Maximum Adverse Excursion graph for the Fibonacci .382 Retracement system

Note that five losing trades but no winning trades experienced a drawdown of $200 per contract or more. This suggests that a protective stop just under $200 from a trade's entry price could cut losses short while have a minimal impact (if any) on winning trades.

Suggestions for Improvement

We can identify two possible ways to make this system better. First, we may want to add a few other Fibonacci retracement levels (such as the well-known 50% and 68.2% levels) to the system. Although adding two more levels would make the system much more complex, it might be worthwhile in this case to sacrifice simplicity for stronger performance.

Second, we did not define an entry technique that would require the market to confirm the validity of the setup. Rather than simply entering a new position on the next open after a setup, we could write an entry technique that might lead to a higher winning percentage. For example, we could specify that the market has to penetrate the extreme of the setup bar on volume that is greater than the 10-bar average of volume.

As another example, we could require the trend of the market one timeframe higher than the timeframe we're trading to be in the same direction as the entry signal, or we would pass on the trade. If we get a buy signal on the daily bars, we could also check the weekly trend. If it's up, we'd take the signal; if it's down, we'd ignore the signal because it's counter to the market's direction in the next higher timeframe. In the case of a sell signal on the hourly bars, we could check the daily chart (one timeframe higher) to see if the daily trend is down. If so, we'll sell short on the hourly-bar signal; if not, we'll pass on the trade.

High/Low Oscillator System

One of the most popular momentum oscillators is constructed by subtracting a moving average from a closing price. When the close is above the moving average and accelerating upward from it, momentum is bullish and increasing; when the close is below the moving average and accelerating downward from it, momentum is bearish and getting even more bearish.

We wanted to design a momentum-based system but with a twist: we'll use highs, lows, and a moving average rather than closes and a moving average.

We'll construct two momentum oscillators. One will represent the high minus the moving average; the other will represent the low minus the moving average. Our system will also employ a longer-term moving average of closes to identify the trend. When the trend is up, we'll use the oscillator of lows to set up an opportunity to buy. When the trend is down we'll use the oscillator of highs to signal a selling opportunity.

For both long and short positions, we'll set a money management stop and enable a trailing stop. Our signal to exit a long position will be when the oscillator of highs crosses below zero; we'll exit our short position when the oscillator of lows crosses above zero. Figure 7 shows the High Low Oscillator system applied to a daily chart of WalMart. Also applied are two indicators, High Oscillator and Low Oscillator. We created these two simple indicators in the QuickEditor.

Figure 7. The High Low Oscillator system applied to a daily chart of WalMart, along with two indicators, High Oscillator and Low Oscillator

Creating the Oscillator Indicators in the QuickEditor

To create the High Oscillator, use the Quick Editor to create an indicator, and call it High Oscillator. The first plot is called High Osc and the formula is High - XAverage (High, Length). The second plot is called Zero Line, and the formula is the numeral 0. We also used one input, named Length; its default value is 20.

To create the Low Oscillator, use the Quick Editor to create an indicator, and call it Low Oscillator. The first plot is called Low Osc and the formula is Low - XAverage (Low, Length). The second plot is called Zero Line, and the formula is the numeral 0. We also used one input, named Length; its default value is 20.

We drew the High Oscillator as a histogram in subgraph 2, and the Low Oscillator as a histogram in subgraph 3.

Defining Your Trading Rules

In this system, we defined both long and short entries as well as exit orders. We also calculated a moving average of closes. The setups, entries, and exits are defined next.

Setup

c) Calculate a 60-bar EMA (exponential moving average) of closes.

d) Calculate the high oscillator and the low oscillator.

Long Entries

a) Check for the 60-bar EMA to be greater than it was one bar ago.

b) Check for the low oscillator to be below zero but rising. In other words, the low sscillator should be a negative value but less negative than one bar ago.

c) Buy on the next open.

Short Entries

a) Check for the 60-bar EMA to be less than it was one bar ago.

b) Check for the high oscillator to be above the zero line but declining. In other words, the high oscillator should be a positive value, but less than it was one bar ago.

c) Sell short on the next open.

Exit Orders

a) Once we enter a position, we'll set a money management stop, a breakeven stop, and a trailing stop.

b) Also, we'll exit our long position on the next open when the high sscillator crosses below zero; we'll exit our short position when the low oscillator crosses above zero.

Designing & Formatting

This section presents the EasyLanguage instructions and formatting for the system, with the EasyLanguage instructions broken down and explained line by line.

EasyLanguage Instructions: High/Low Oscillator (STAD5: High/Low Osc)

```
Input: Len1(10), Len2(60), TrailStp(700);
Vars: HighOsc(0), LowOsc(0), CloseAvg(0), BuySetup(False), SellSetup(False);

{Variable Assignments}
HighOsc = High - XAverage(High, Len1);
LowOsc = Low - XAverage(Low, Len1);
CloseAvg = XAverage(Close, Len2);

{Entry Setups}
BuySetup = CloseAvg > CloseAvg[1] AND Close > CloseAvg AND LowOsc < 0;
SellSetup = CloseAvg < CloseAvg[1] AND Close < CloseAvg AND HighOsc > 0;
```

```
{Long Entry}
If BuySetup AND LowOsc > LowOsc[1] then
        Buy next bar at Market;

{Long Exit}
If MarketPosition = 1 then begin
        ExitLong ("LTrailStop") next bar at Close-(TrailStp/BigPointValue)
        Stop;
        If HighOsc Crosses Under 0 then
                ExitLong ("LTiming") next bar at Market;
End;

{Short Entry}
If SellSetup AND HighOsc < HighOsc[1] then
        Sell next bar at Market;

{Short Exit}
If MarketPosition = -1 then Begin
        ExitShort ("STrailStop")next bar at Close+(TrailStp/BigPointValue)
        Stop;
        If LowOsc crosses Over 0 then
                ExitShort("STiming") next bar at Market;
End;
```

Inputs

Following is the list of all the inputs we used in this system:

Input	Default	Description
Len1	10	The oscillator length value, expressed in bars, for the calculation of the exponential moving average of the High and Low
Len2	60	A length value, expressed in bars, for the calculation of the exponential moving average of the Close.
TrailStp	700	A monetary amount used as the basis for the system Trailing Stop.

In addition to these inputs, we define the following variables:

```
Vars: HighOsc(0), LowOsc(0), CloseAvg(0), BuySetup(false), SellSetup(false);
```

Setup

For the sake of efficiency and ease of use, the important system calculations are assigned to variables. The HighOSc variable is the difference between the high and the exponential moving average of the high. The LowOsc variable is the difference between the low and the exponential moving average of the low. The CloseAvg variable is the exponential moving average of the close.

```
HighOsc = High - XAverage(High, Len1);
LowOsc = Low - XAverage(Low, Len1);
CloseAvg = XAverage(Close, Len2);
```

The BuySetup variable is a true/false variable that represents half of the required criteria in order to place a long entry. For the BuySetup to return a value of true, the CloseAvg must be greater than the CloseAvg of the previous bar, the close must be greater than the CloseAvg, and the LowOsc must be less than 0. If these three criteria are met, BuySetup will return true and half of the required criteria for a long entry will have been met on that bar.

```
BuySetup = CloseAvg > CloseAvg[1] AND Close > CloseAvg AND LowOsc < 0;
```

Likewise, the SellSetup variable is a true/false variable that represents half of the required criteria in order to place a short entry. For the SellSetup to return a value of true, the CloseAvg must be less than the CloseAvg of the previous bar, the close must be less than the CloseAvg, and the HighOsc must be greater than 0. If these three criteria are met, BuySetup will return true and half of the required criteria for a short entry will have been met on that bar.

```
SellSetup = CloseAvg < CloseAvg[1] AND Close < CloseAvg AND HighOsc >0;
```

Long Entry

If the BuySetup is true and the LowOsc is greater than the LowOsc value for the previous bar a buy order is placed on the open of the next bar.

```
If BuySetup AND LowOsc > LowOsc[1] then Buy next bar at Market;
```

Long Exit

If the current market position is Long, at least one of the two system exits will be placed. The trailing stop, which is based on the TrailStp input, will always be enabled when the system is long. The trailing stop is essentially an exit at the current close, less the TrailStp input divided by BigPointValue. The division converts the monetary TrailStp value to a point value that can be used as part of the stop price. If the HighOsc variable crosses below 0, a timing stop will be ordered at the open of the next bar.

```
If MarketPosition = 1 then begin
        ExitLong ("LTrailStop") next bar at Close -(TrailStp/BigPointValue) Stop;
        If HighOsc Crosses Under 0 then ExitLong ("LTiming") next bar at Market;
End;
```

Short Entry

If the SellSetup variable is true and the HighOsc is less than the HighOsc value for the previous bar a sell order is placed on the open of the next bar.

```
If SellSetup AND HighOsc < HighOsc[1] then Sell next bar at Market;
```

Short Exit

If the current market position is short, at least one of the two system exits will be placed. The trailing stop, which is based on the TrailStp input, will always be enabled when the system is short. The trailing stop is essentially an exit at the current close, plus the TrailStp input divided by BigPointValue. The division converts the monetary TrailStp value to a point value that can be used as part of the stop price. If the LowOsc variable crosses above 0, a timing stop will be ordered at the open of the next bar.

```
If MarketPosition = -1 then Begin
        ExitShort ("STrailStop")next bar at Close+(TrailStp/BigPointValue) Stop;
        If LowOsc crosses Over 0 then ExitShort("STiming") next bar at Market;
End;
```

General System Format

When we apply a system to a chart, we can use the options in the **Format** dialog box to format costs, stops, and properties. We did not enter an amount for slippage and commission, although those costs must be taken into account before a system is traded.

Note: *Remember that commissions are calculated on a per contract/share basis. When you are trading stocks, you would enter the average commission you are charged divided by the number of shares the system is buying and selling.*

Under the **Stops** tab, we enabled a money management stop at $300, and a breakeven stop with a floor of $200. The money management stop option holds the dollar amount per position or per contract/share we are willing to risk before exiting from the position. The Breakeven stop allows us to specify the dollar amount of open profit that our position must achieve before we move our stop to our entry price (breakeven).

Note: *When you are trading stocks and you choose the stop option* ***Apply on a per share (contract) basis****, you will type in the number of points you are willing to lose before you exit. When you are trading futures or any other instrument that has a different dollar-point value, you would type the maximum number of dollars you are willing to risk per contract traded.*

In the **Properties** tab, we selected the option **Do not allow multiple entries in the same direction**. If the system is in a long position, and market conditions generate another long entry order, the order is ignored; if the system is in a short position, and market conditions generate another short entry order, the order is ignored.

Testing & Improving

We tested the High/Low Oscillator system on WalMart daily data from 1988 to 1998. The system generated 129 trades, of which 60% were profitable. Figure 8 shows Portfolio Maximizer's Individual Report for the High/Low Oscillator system.

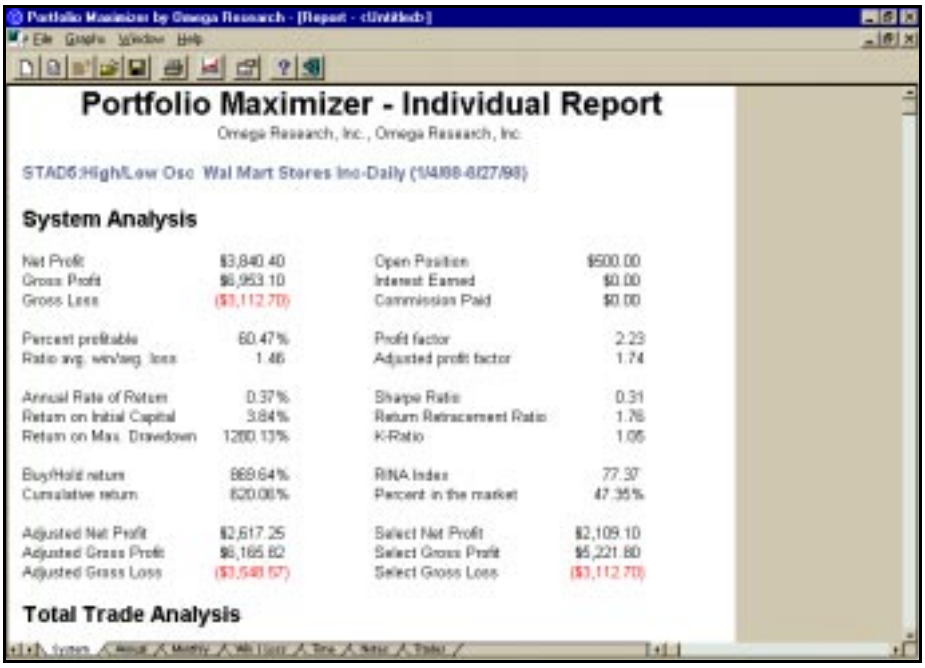


Figure 8. Portfolio Maximizer Individual Report for the High/Low Oscillator system

The ratio of average win to average loss was 1.46, and the profit factor (the number of dollars won per dollar lost) was 2.23. The Sharpe Ratio was .31 (compared to a reference point of .25), but the Return Retracement Ratio was only 1.76 (compared to a guideline of 3.0), and the K-Ratio was also low at 1.05 (compared to a standard of 2.5).

The RINA Index, however, was excellent at 77.37 (against a reference point of 30). Also, Net Profit/Largest Loss was superior at 12.8 (reference point of 7.0), and Net Profit/Maximum

Drawdown was excellent at 12.8 (against a guideline of 5.0). The High/Low Oscillator System Equity Curve is shown in Figure 9.

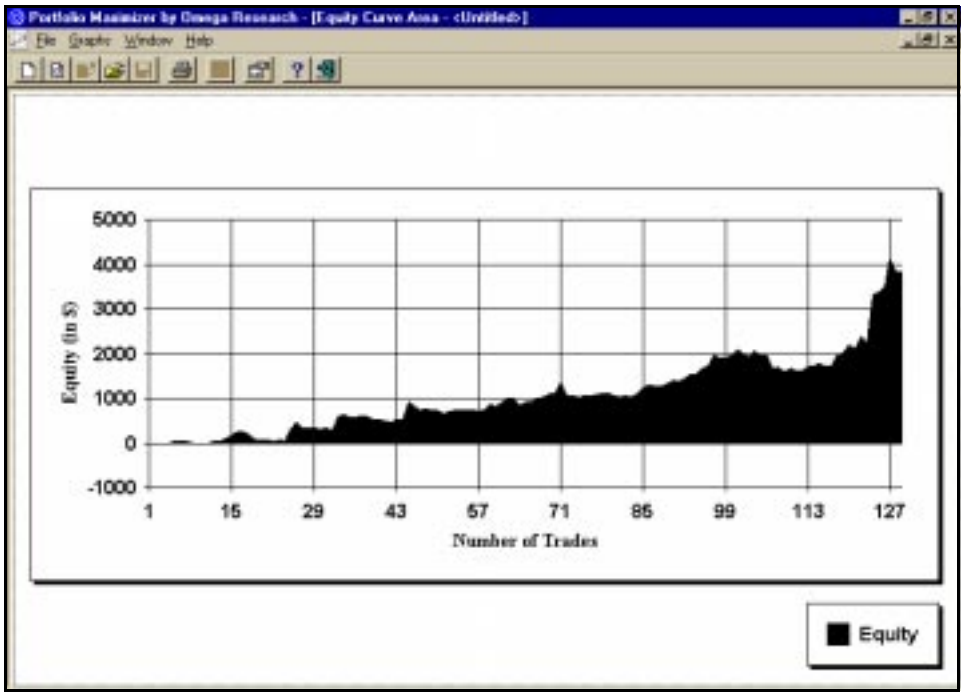


Figure 9. Equity curve for the High/Low Oscillator system

Although we would like to see it rising more sharply, at least it is rising throughout most of the test period and finishing strongly with a new equity high on trade 127.

Figure 10 displays the Monthly Rolling Net Profit.

Figure 10. Monthly Rolling Net Profit graph for the High/Low Oscillator system

The Monthly Rolling Net Profit posts the growth of equity month-to-month. We would describe this progression of equity as outstanding.

Maximum Adverse Excursion is the subject of Figure 11.

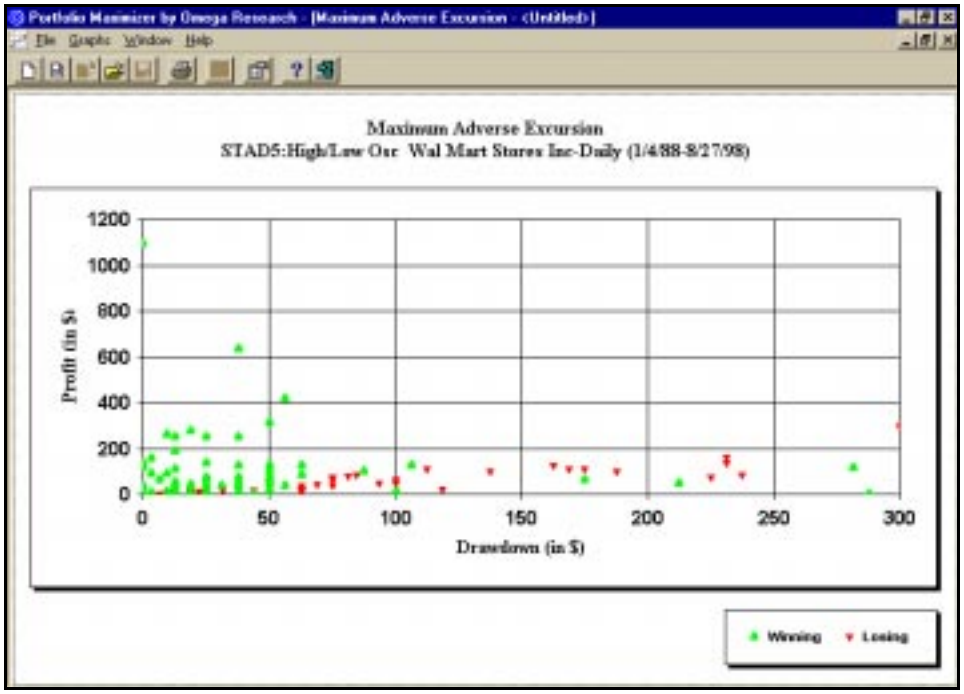


Figure 11. Maximum Adverse Excursion graph for the High/Low Oscillator system

Note that setting a stop at $150 (per 100 shares) would have cut short nine losses while sacrificing only four small winners.

Suggestions for Improvement

The High/Low Oscillator system identifies very good setups. For this system, however, we didn't define an entry requirement beyond the setup; instead, we simply entered a new position on the next open after a setup. We believe that this system could be improved by adding specific entry requirements. For example, to buy we could require that the market makes a new three-bar high or that to sell short the market declines 30% of an average true range from the open. Most systems perform much better with setups and entry conditions than they do with setups alone.

Stochastic Overbought/Oversold System

In Volume 3 of STAD Club, we included a system called *Stochastics*. It was fairly complex and produced good results. For STAD Volume 5, we're going to see if we can design a simpler Stochastic system that still makes a good profit.

Following is a brief explanation of Stochastic for our STAD Club members who may not be familiar with this indicator.

The Stochastic oscillator was developed by George Lane at Investment Educators in the late 1950s. Stochastic evaluates a market's momentum by determining the relative position of closing prices within the high-low range of a specified number of bars. A 14-bar Stochastic, for example, measures the location of closing prices within the total high-low range of the previous 14 bars. Stochastic expresses the relationship between the close and the high-low range as a percentage between zero and 100. A Stochastic value of 70 or higher indicates that

the close is near the top of the range; a Stochastic value of 30 or lower means that the close is near the bottom of the range.

In a robust uptrend, prices generally close near the top of the recent range; in a strong downtrend, prices usually close near the bottom of the range. When an uptrend is approaching a turning point, prices begin to close farther away from the high of the range, and when a downtrend is weakening, prices tend to close farther away from the low of the range. The purpose of the Stochastic oscillator is to alert technicians to the failure of bulls to close prices near the highs of an uptrend or the inability of bears to close prices near the lows of a downtrend.

The Stochastic is plotted as two lines: %K and %D. The formula for %K is %K = 100 [(C - L*n*)/(H*n* - L*n*)], where C is the current close, L*n* is the low of the *n*-bar period, and H*n* is the high of the *n*-bar period. The %D formula is %D = 100 (H_3/L_3), in which H_3 is the three-bar sum of (C - L*n*) and L_3 is the three-bar sum of (H*n* - L*n*).

The %K and %D formulas produce the *fast Stochastic oscillator*, which is generally considered too sensitive and erratic. Fast Stochastic can be subjected to a further three-bar smoothing, however, which results in the *slow Stochastic* that most analysts prefer. In the smoothed version of Stochastic, the fast %D becomes the slow %K, and a three-bar moving average of the fast %D becomes the slow %D. Slow %K is usually drawn as a solid line and slow %D as a dotted or dashed line.

In the Stochastic Overbought/Oversold System, our setup to buy will be %K crossing above %D in oversold territory. Figure 12 shows the Stochastic Overbought/Oversold system applied to a weekly Caterpillar chart. We also applied the built in Stochastic - Slow Indicator to the chart.

Figure 12. The Stochastic Overbough Oversold system applied ot a weekly chart of Caterpillar

Our setup to sell short will be %K crossing below %D in overbought territory.

Our buy entry will be at the high of the setup bar plus one point; our sell entry will be one point below the low of the setup bar. When we enter a long position, we'll place a protective stop at the low of the setup bar, and when we're short, our protective stop will be set at the high of the setup bar.

The signal to exit a long position will be %K crossing below %D above the oversold level. We'll exit a short position when %K crosses above %D below the overbought level.

Defining Your Trading Rules

In this system, we defined both long and short entries as well as exit orders. We also did some setup work to calculate a 12-bar slow Stochastic with an overbought level at 70 and an oversold level at 30. The setups, entries, and exits are defined next.

Setup

a) Calculate a 12-bar slow Stochastic.

b) Set an overbought level at 70.

c) Set an oversold level at 30.

Long Entries

a) Check for %K to cross above %D in oversold territory (below 30).

b) Place a buy stop one point above the high of the setup bar. This buy stop will remain in effect for six bars.

Short Entries

a) Check for %K to cross below %D in overbought territory (above 70).

b) Place a sell stop one point below the low of the setup bar. This sell stop will remain active for six bars.

Exit Orders

a) Once we enter a long position, we'll place a protective stop at the low of the setup bar; when we enter a short position, we'll place a protective stop at the high of the setup bar.

Designing & Formatting

This section presents the EasyLanguage instructions and formatting for the system, with the EasyLanguage instructions broken down and explained line by line.

EasyLanguage Instructions: Stochastic Overbought/Oversold (STAD5: Stochastic)

```
Inputs: Length(12), OBought(70), OSold(30), SetUpLen(6);
Vars: KVal(0), DVal(0), BuySetup(0), SellSetup(0), BuyPeriod(99), SellPeriod(99), BuyStop(0),
SellStop(0);

{Stochastic Line Calculations}
KVal = SlowKCustom(High,Low,Close,Length);
DVal = SlowDCustom(High, Low, Close, Length);

{Entry Setups}
IF KVal < OSold AND KVal Crosses Above DVal Then Begin
        BuySetup = Highest(High, 2);
        BuyStop = Low;
        BuyPeriod = 0;
End;
```

```
IF KVal > OBought AND KVal Crosses Below DVal Then Begin
        SellSetup = Lowest(Low, 2);
        SellStop = High;
        SellPeriod = 0;
End;

{Counters}
IF MarketPosition = 1 ThenBuyPeriod = SetUpLen + 1
Else
        BuyPeriod = BuyPeriod + 1;
IF MarketPosition = -1 ThenSellPeriod = SetUpLen + 1
Else
        SellPeriod = SellPeriod + 1;

{Entry Signals}
IF BuyPeriod <= SetUpLen ThenBuy Next Bar at BuySetup + 1 Point Stop;
IF SellPeriod <= SetUpLen ThenSell Next Bar at SellSetup - 1 Point Stop;

{System Exits}
IF KVal Crosses Below DVal AND KVal > OSold Then Begin
        BuyPeriod = SetUpLen;
        ExitLong Next Bar at Market;
End;

IF Kval Crosses Above DVal AND KVal < OBought Then Begin
        SellPeriod = SetUpLen;
        ExitShort Next Bar at Market;
End;

{Stops}
ExitLong Next Bar at BuyStop Stop;
ExitShort Next Bar at SellStop Stop;
```

Inputs

Following is the list of all the inputs we used in this system:

Input	Default	Description
Length	12	Length, expressed in bars, used in the calculation of the %K and %D.
Obought	70	The value used to identify an Overbought market.
Osold	30	The value used to identify an Oversold market.
SetUpLen	6	The Setup Length. A length, expressed in bars, of the Buy and Sell setups, within which an entry may occur.

In addition to these inputs, we define the following variables:

```
Vars: KVal(0), DVal(0), BuySetup(0), SellSetup(0), BuyPeriod(99), SellPeriod(99), BuyStop(0),
SellStop(0);
```

Setup

The values of the %K and %D lines are assigned to the variables KVal and DVal respectively. Since the Stochastic calculations are referenced throughout the system, it is much easier and more efficient to assign them to variables.

```
KVal = SlowKCustom(High,Low,Close,Length);
DVal = SlowDCustom(High, Low, Close, Length);
```

When KVal is below the OSold (Oversold) level and KVal crosses above DVal, the buy setup is initiated, and the variable BuySetup is assigned the higher high value between the current bar and the previous bar. The BuyStop variable, which will be used for the protective stop, is set to the low of the setup bar. Lastly, the BuyPeriod counter is reset to 0. A long entry can only be placed within the Setup period (SetUpLen).

```
IF KVal < OSold AND KVal Crosses Above DVal Then Begin
        BuySetup = Highest(High, 2);
        BuyStop = Low;
        BuyPeriod = 0;
End;
```

When KVal is above the OBought (Overbought) level and KVal crosses below DVal, the sell setup is initiated and the variable SellSetup is assigned the lower low value between the current bar and the previous bar. The SellStop variable, which will be used for the protective stop, is set to the high of the setup bar. Lastly, the SellPeriod counter is reset to zero (0). A short entry can only be placed within the setup period (SetUpLen).

```
IF KVal > OBought AND KVal Crosses Below DVal Then Begin
        SellSetup = Lowest(Low, 2);
        SellStop = High
        SellPeriod = 0;
End;
```

The increase of the counter values in this system, BuyPeriod and SellPeriod, is based on the market position of the system. In the case of a long position, even though the BuyPeriod variable was reset as a result of a new buy setup, the BuyPeriod variable is automatically set to value greater than the SetUpLen because the current position is already long, thus preventing any additional long entries. If the current position is not long, the BuyPeriod variable will be incremented by 1. The procedure for the SellPeriod counter is analogous. If the current position is short, even though the SellPeriod variable was reset as a result of a new sell setup, the SellPeriod variable is automatically set to a value greater than the SetUpLen because the current position is already short, thus preventing any additional short entries. If the current position is not short, the SellPeriod variable will be incremented by 1.

```
IF MarketPosition = 1 Then BuyPeriod = SetUpLen + 1
Else
        BuyPeriod = BuyPeriod + 1;

IF MarketPosition = -1 Then SellPeriod = SetUpLen + 1
Else
        SellPeriod = SellPeriod + 1;
```

Long Entry

For the long entry, while within the specified buy setup period (SetUpLen), a buy stop order will be placed at the BuySetup value (which was defined in the buy setup) plus 1 point.

```
IF BuyPeriod <= SetUpLen Then Buy Next Bar at BuySetup + 1 Point Stop;
```

Short Entry

For the short entry, while within the specified sell setup period (SetUpLen), a sell stop order will be placed at the SellSetup value (which was defined in the sell setup) minus 1 point.

```
IF SellPeriod <= SetUpLen Then Sell Next Bar at SellSetup - 1 Point Stop;
```

System Exits

For a long position, a long exit at the open of the next bar will be generated if the %K line (KVal) is greater than the oversold (OSold) level, and the %K line crosses below the %D line (Dval). In addition, if there is an existing buy setup, it will be nullified by the assignment of the SetUpLen value to the BuyPeriod variable. For a short position, a short exit at the open of the next bar will be generated if the %K line is below the overbought (OBought) level, and the %K lines crosses above the %D line (DVal). In addition, if there is an existing sell setup, it will be nullified by the assignment of the SetUpLen value to the SellPeriod variable.

```
IF KVal Crosses Below DVal AND KVal > OSold Then Begin
        BuyPeriod = SetUpLen;
        ExitLong Next Bar at Market;
End;

IF Kval Crosses Above DVal AND KVal < OBought Then Begin
        SellPeriod = SetUpLen;
        ExitShort Next Bar at Market;
End;
```

Protective Stops

Protective stops are placed at BuyStop (low of the buy setup bar) and SellStop (high of the sell setup bar). These stops help to reduce the loss when the market moves against us.

```
ExitLong Next Bar at BuyStop Stop;
ExitShort Next Bar at SellStop Stop;
```

General System Format

When we apply a system to a chart, we can use the options in the **Format** dialog box to format costs, stops, and properties. We did not enter an amount for slippage and commission, although those costs must be taken into account before a system is traded.

Note: *Remember that commissions are calculated on a per contract/share basis. When you are trading stocks, you would enter the average commission you are charged divided by the number of shares the system is buying and selling.*

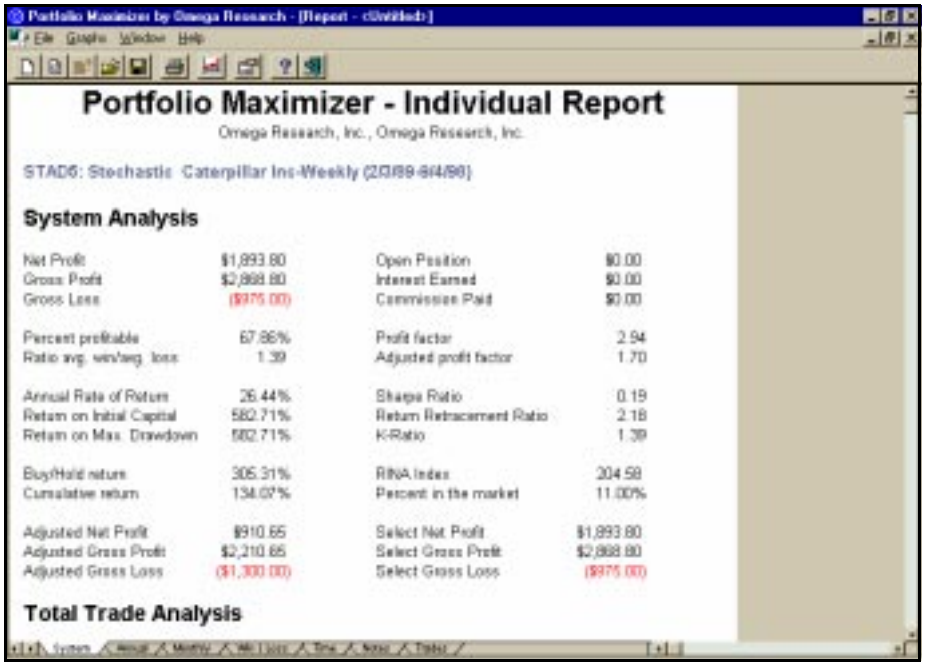
Under the **Stops** tab, we did not enable any stops for this system.

Note: *When you are trading stocks and you choose the stop option* ***Apply on a per share (contract) basis****, you will type in the number of points you are willing to lose before you exit. When you are trading futures or any other instrument that has a different dollar-point value, you would type the maximum number of dollars you are willing to risk per contract traded.*

In the **Properties** tab, we selected the option **Do not allow multiple entries in the same direction**. If the system is in a long position, and market conditions generate another long entry order, the order is ignored; if the system is in a short position, and market conditions generate another short entry order, the order is ignored.

Testing & Improving

We applied the Stochastic Overbought/Oversold System to weekly data of Caterpillar from February, 1989, to September, 1998. The system generated 28 trades, of which 67% were profitable. Figure 13 shows the Portfolio Maximizer Individual Report.



Portfolio Maximizer - Individual Report

Omega Research, Inc., Omega Research, Inc.

STAD6: Stochastic Caterpillar Inc-Weekly (2/3/89-9/4/98)

System Analysis

Net Profit	$1,893.80	Open Position	$0.00
Gross Profit	$2,868.80	Interest Earned	$0.00
Gross Loss	($975.00)	Commission Paid	$0.00
Percent profitable	67.86%	Profit factor	2.94
Ratio avg. win/avg. loss	1.39	Adjusted profit factor	1.70
Annual Rate of Return	26.44%	Sharpe Ratio	0.19
Return on Initial Capital	582.71%	Return Retracement Ratio	2.18
Return on Max. Drawdown	582.71%	K-Ratio	1.39
Buy/Hold return	305.31%	RINA Index	204.58
Cumulative return	134.07%	Percent in the market	11.00%
Adjusted Net Profit	$910.65	Select Net Profit	$1,893.80
Adjusted Gross Profit	$2,210.65	Select Gross Profit	$2,868.80
Adjusted Gross Loss	($1,300.00)	Select Gross Loss	($975.00)

Total Trade Analysis

Figure 13. The Portfolio Maximizer Individual Report for the Stochastic Overbought/Oversold system

The ratio of average win to average loss was 1.39. That's not bad for a support-and-resistance system that is 67% accurate. The Sharpe Ratio, Return Retracement Ratio, and K-Ratio all came in a little below the guidelines we shoot for. Net Profit/Largest Loss (at 5.83) was also below our reference point of 7.0, but Net Profit/Maximum Drawdown was acceptable at 5.83, with 5.0 being the standard.

Figure 14 is a graph of Total Trades.

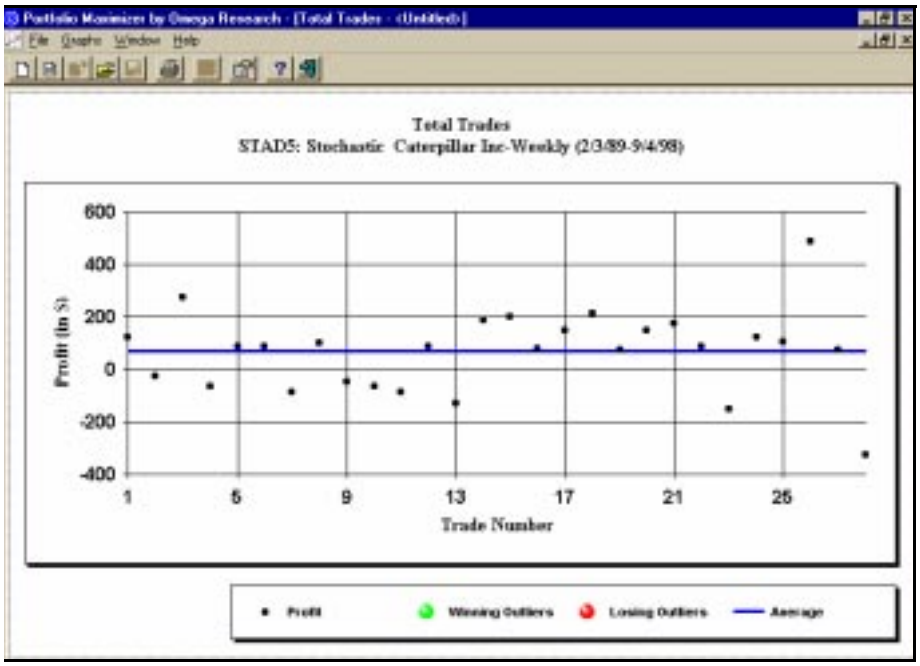
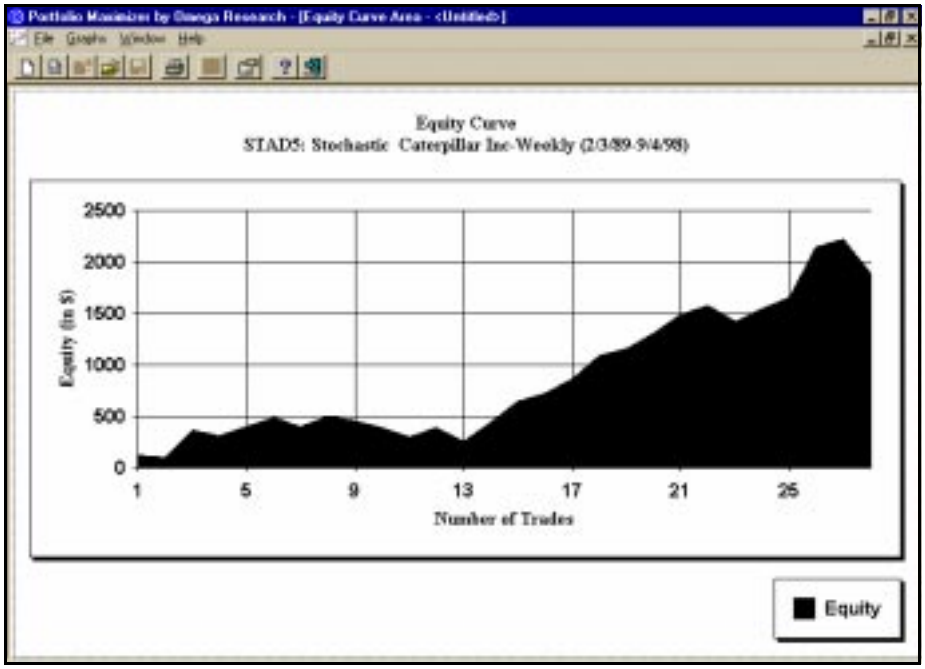


Figure 14. Total Trades graph for the Stochastic Overbought/Oversold system

It shows that there were no winning or losing outliers (trades more than three standard deviations from average). The biggest winner was about $500, the biggest loser about $300. In a support-and-resistance system such as this one, we would not expect to see outliers as we're playing more for consistent profits than a few huge profits. The Equity Curve for our Stochastic Overbought/Oversold System is displayed in Figure 15.



igure 15. Equity Curve for the Stochastic Overbought/Oversold system

It shows only marginally profitable trading through trade number 13 but improves thereafter.

Figure 16 graphs Monthly Rolling Net Profit.

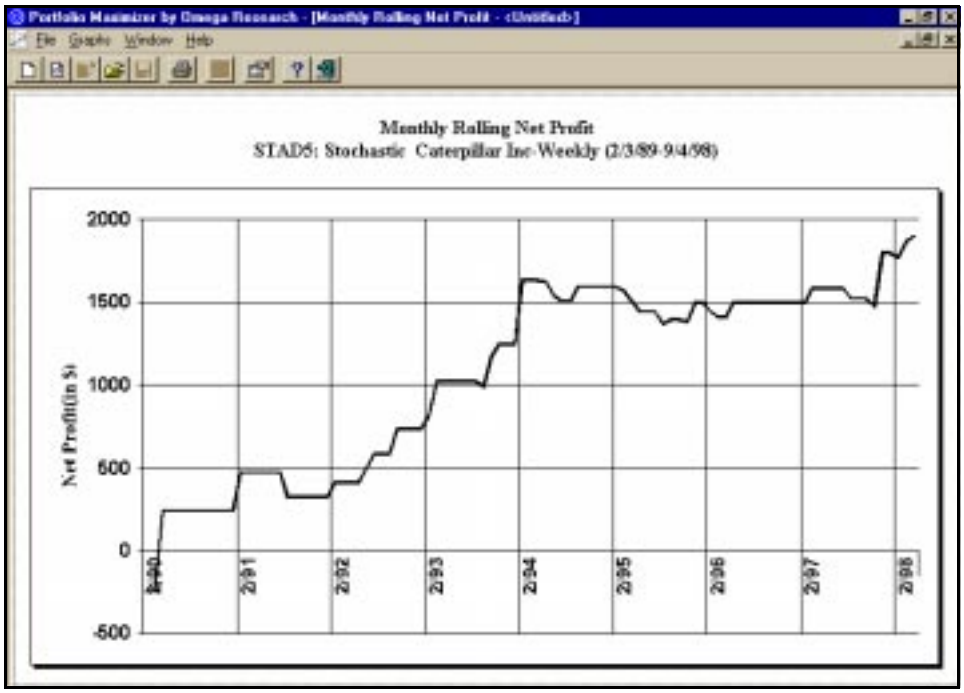


Figure 16. Monthly Rolling Net Profit graph for the Stochastic Overbought/Oversold system

Here the striking feature is the system's flat performance from 1994 to 1998.

The Underwater Equity Curve is the focus of Figure 17.

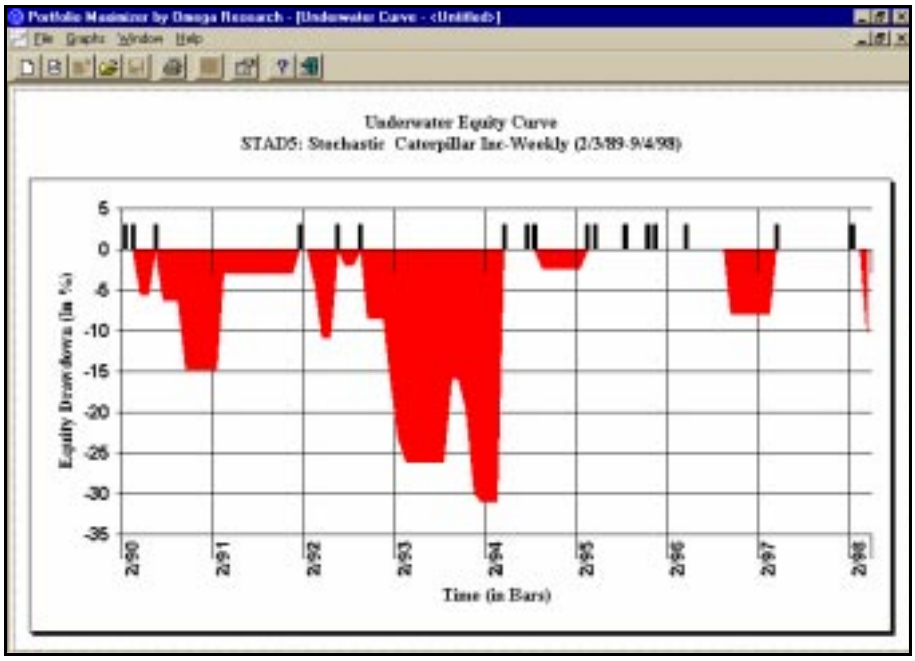


Figure 17. Underwater Equity Curve for the Stochastic Overbought/Oversold system

Note that the biggest drawdown between new equity highs (about 30%) occurred early in 1994. The three drawdowns during the system's relatively flat period from 1994 to early 1998 were

modest at approximately 2%, 8%, and 10%. It's not that our system lost a lot of money during that four-year period; the problem is that it didn't make much money for us either.

Figure 18, a Maximum Adverse Excursion graph, shows that no winning trades suffered a drawdown of more than $275.

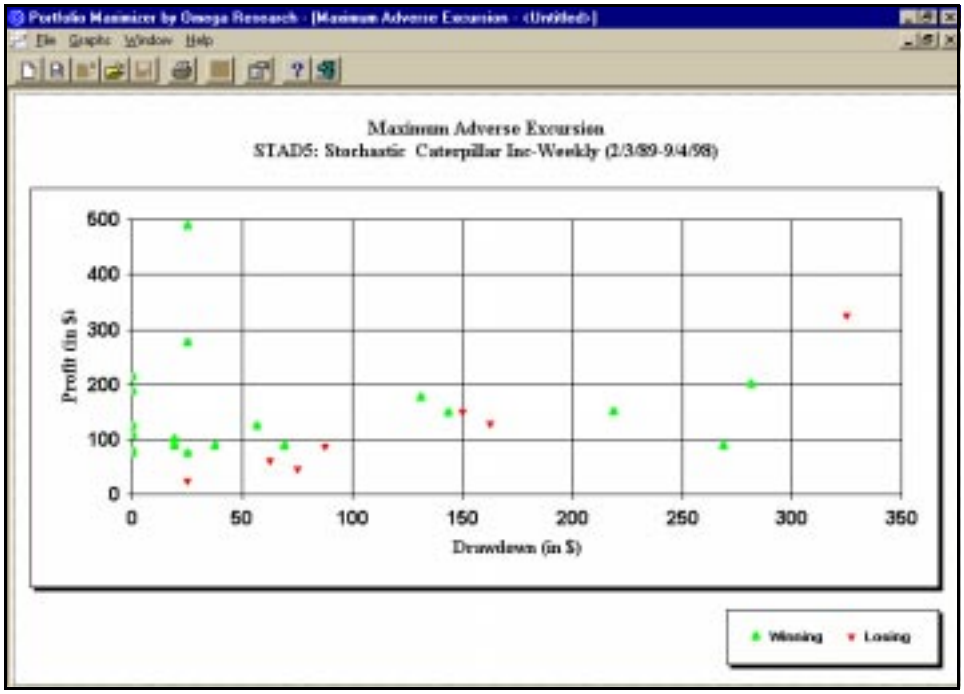


Figure 18. Maximum Adverse Excursion graph for the Stochastic Overbought/Oversold system

Therefore, we might want to set our initial protective stop no more than $275 from our entry point.

Suggestions for Improvement

Since we know that a support-and-resistance system will generally work well in a choppy market but perform poorly in a trending market, perhaps we could improve this system by adding a "trendiness" indicator such as ADX. When ADX is rising, suggesting a strong trend, we might want to ignore the Stochastic signals; when ADX is declining, suggesting a choppy market, we could act on our Stochastic signals.

Pivot Point Support & Resistance System

Support and resistance are among the most important concepts in all of technical analysis. Support is like a floor beneath current prices; resistance is like a ceiling above current prices. At support buyers become strong enough to overcome sellers and prices rise. At resistance sellers take control and overpower buyers and prices fall.

Pivot points (also referred to as swing highs and swing lows) are one useful way to determine levels of support and resistance. A pivot point high is a price bar with a lower high immediately before it and a lower high immediately after it. In other words, a pivot point high is a bar with a lower high to its left and to its right. A pivot point low is a price bar with a higher low immediately before it and a higher low immediate after it. In other words, a pivot point low has higher lows to its left and right.

A key concept in support and resistance theory is that support when broken becomes new resistance, and resistance when broken becomes new support. Think of it like this: support is the floor beneath the market. If the market falls through the floor, the old floor is now the ceiling

above the market. Resistance is like the ceiling above the market. If the market soars through the ceiling, the old ceiling becomes the new floor below the market.

We decided to write a system based on pivot points, support and resistance, and their changing roles when broken. Figure 19 shows the system applied to a daily chart of Treasury bonds.

Figure 19. The Pivot Point Support & Resistance system applied to the daily Treasruy Bond chart

Defining Your Trading Rules

In this system, we defined both long and short entries as well as exit orders. We also did some setup work to calculate an EMA (exponential moving average) and an ATR (average true range). The entries, exits, and setup are described next:

Setup

a) Calculate a nine-bar EMA.

b) Calculate a 10-bar ATR.

Long Entries

a) Check for a pivot-point low that is also the lowest low of the past eight bars.

b) Check for a price decline below the pivot-point low.

c) Check for a bar on which the following six conditions are true:

The bar closes above the previous bar’s close.

The bar closes above its open.

The bar closes in the top third of its true range.

The bar closes above the pivot-point low.

The bar has a true range greater than the 10-bar average true range.

The bar on which the five preceding conditions are true occurs within four bars of the pivot-point low.

d) Buy at one point above the high of the setup bar if the following two conditions are true:

The entry point is above EMA:9

The entry is within five bars of the setup bar.

Short Entries

a) Check for a pivot point high that is also the highest high of the past eight bars.

b) Check for a price rally above the pivot-point high.

c) Check for a bar on which the following six conditions are true:

The bar closes below the previous bar's close.

The bar closes below its open.

The bar closes in the bottom third of its true range.

The bar closes below the pivot-point high.

The bar has a true range greater than the 10-bar average true range.

The bar on which the five preceding conditions are true occurs within four bars of th pivot-point high.

d) Sell short at one point below the low of the setup bar if the following two conditions are true:

The entry point is below EMA:9

The entry is within five bars of the setup bar.

Exit Orders

a) An initial money management stop is written into the EasyLanguage code for this system. For a long position, the initial stop is one point below the low of the setup bar. For a short position, the initial stop is one point above the high of the setup bar.

b) A breakeven stop is also included in the EasyLanguage code for this system. For a long position, the stop is raised to breakeven when open profits on a closing basis are equal to or greater than two times the initial risk on the trade. For a short position, the stop is lowered to breakeven when open profits on a closing basis are equal to or greater than two times the initial risk on the trade.

c) We'll exit a long position when prices make a new 44-bar high; we'll exit a short position when prices make a new 44-bar low.

Designing & Formatting

This section presents the EasyLanguage instructions and formatting for the system, with the EasyLanguage instructions broken down and explained line by line.

EasyLanguage Instructions: Pivot Point Support & Resistance (STAD5: Pivot Point)

```
Inputs: HiLoLen(8), ATRLen(10), SetUpInt(4), XAvgLen(9), SetUpLen(24), RiskMult(2), HiLoStop(44);
Vars: LPivot(0), HPivot(0), EMA(0), BuySetupHigh(0), BuySetupCount(100), SellSetupLow(0),
SellSetupCount(100);
Vars: BuySetupLow(0), SellSetupHigh(0), LExitFlag(0), SExitFlag(0), LExitLow(0), SExitHigh(0),
PMult(2), TT(0);
```

```
{Assignment of basic system conditions}
Condition1 = SwingLow(1, Low, 1, 2) <> -1;
Condition2 = SwingHigh(1, High, 1, 2) <> -1;
Condition3 = Low[1] = Lowest(Low, HiLoLen)[1];
Condition4 = High[1] = Highest(High, HiLoLen)[1];
Condition5 = TrueRange > AvgTrueRange(ATRLen);

{Variables which represeent indicate the occurence of basic system criteria}
LPivot= MRO(Condition1 AND Condition3, SetUpInt, 1);
HPivot = MRO(Condition2 AND Condition4, SetUpInt, 1);
{Assignment of System variables}
EMA = XAverage(Close, XAvgLen);
TT = TotalTrades;

{Buy Setup}
IF LPivot > 0 AND Lowest(Low, LPivot) < Low[LPivot+1] Then Begin
        IF Condition5 AND Close > MaxList(Close[1], Open, Low +
        (2*(TrueRange/3)), Low[LPivot+1]) Then Begin
                BuySetupHigh = High;
                BuySetupLow = Low;
                BuySetupCount = 0;
        End;
End;

{Sell Setup}
IF HPivot > 0 AND Highest(High, HPivot) > High[HPivot+1] Then Begin
        IF Condition5 AND Close < MinList(Close[1], Open, Low + (TrueRange/3),
        High[HPivot+1]) Then Begin
                SellSetupLow = Low;
                SellSetupHigh = High;
                SellSetupCount = 0;
        End;
End;

{Breakeven Stops, above floor}
IF MarketPosition = 1 OR TT <> TT[1] Then Begin
        BuySetupCount = SetUpLen;
        IF Close > EntryPrice + (PMult*(EntryPrice - LExitLow)) OR
        LExitFlag = 0 Then Begin
                ExitLong Next Bar at EntryPrice Stop;
                LExitFlag = 0;
        End;
End;
IF MarketPosition = -1 OR TT <> TT[1] Then Begin
        SellSetupCount = SetUpLen;
```

```
        IF Close < EntryPrice - (PMult*(SExitHigh - EntryPrice)) OR
        SExitFlag = 0 Then Begin
                ExitShort Next Bar at EntryPrice Stop;
                SExitFlag = 0;
        End;
End;

{Varaible counters for the length of the Entry period}
BuySetupCount = BuySetupCount + 1;
SellSetupCount = SellSetupCount + 1;

{Long Entry}
IF BuySetupCount <= SetUpLen Then Begin
        IF BuySetupHigh + 1 Point > EMA Then Begin
                Buy Next Bar at BuySetupHigh + 1 Point Stop;
                LExitLow = BuySetupLow;
                LExitFlag = 1;
        End;
End;

{Short Entry}
IF SellSetupCount <= SetUpLen Then Begin
        IF SellSetupLow + 1 Point < EMA Then Begin
                Sell Next Bar at SellSetupLow - 1 Point Stop;
                SExitHigh = SellSetupHigh;
                SExitFlag = 1;
        End;
End;

{Initial Breakeven Stops}
IF LExitFlag = 1 ThenExitLong Next Bar at LExitLow - 1 Point Stop;
IF SExitFlag = 1 ThenExitShort Next Bar at SExitHigh + 1 Point Stop;

{System Exits}
ExitLong Next Bar at Highest(High, HiLoStop) Limit;
ExitShort Next Bar at Lowest(Low, HiLoStop) Limit;
```

Inputs

Following is the list of all the inputs we used in this system:

Input	Default	Description
HiLoLen	8	Length, expressed in bars, used in the calculation of the Highest High and the Lowest Low.
ATRLen	10	Length, expressed in bars, used to calculate the average true range.
SetUpInt	4	The Setup Initialization. A length, expressed in bars, in which the Buy/Sell Setup must be initiated after the pivot point.

Input	Default	Description
XavgLen	9	Length, expressed in bars, of the exponential moving average
SetUpLen	24	The Setup Length. A length, expressed in bars, of the Buy and Sell setups, within which an entry may occur.
RiskMult	2	The multiple of the initial risk, which defines the floor of the breakeven stop.
HiLoStop	44	Length, expressed in bars, of the Highest High and Lowest Low used in the system exits.

In addition to these inputs, we define the following variables:

```
Vars: LPivot(0), HPivot(0), EMA(0), BuySetupHigh(0), BuySetupCount(100), SellSetupLow(0),
SellSetupCount(100);
Vars: BuySetupLow(0), SellSetupHigh(0), LExitFlag(0), SExitFlag(0), LExitLow(0),
SExitHigh(0), PMult(2), TT(0);
```

Setup

In an effort to increase the readability of the system, we have specified important criteria in the conditional statements below. These statements, contingent on the outcome of the expressions, return either true or false. Condition1 is used to identify a swing low or low pivot. If a swing low does not exist, the SwingLow function will return a "-1," otherwise it will return the value of the pivot. If the function returns a "-1," Condition1 will be equal to true, otherwise it will be false. Condition2 is used to identify a swing high or high pivot. If a swing high does not exist, the SwingHigh function will return a "-1," otherwise it will return the value of the pivot. If the function returns a "-1," Condition2 will be equal to true, otherwise it will be False. Condition3 is True if the Low of the previous bar is the lowest low of the past 8 bars (HiLoLen), up to and including the previous bar. Condition4 is true if the high of the previous bar is the highest high of the past 8 bars (HiLoLen), up to and including the previous bar. Finally, Condition5 will be true if the True Range is greater than the Average True Range of the past 10 bars (ATRLen).

```
Condition1 = SwingLow(1, Low, 1, 2) <> -1;
Condition2 = SwingHigh(1, High, 1, 2) <> -1;
Condition3 = Low[1] = Lowest(Low, HiLoLen)[1];
Condition4 = High[1] = Highest(High, HiLoLen)[1];
Condition5 = TrueRange > AvgTrueRange(ATRLen);
```

The variable LPivot returns the number of bars since most recent occurrence (MRO) of a true value for both Condition1 and Condition3, within the setup initialization period. If such an occurrence does not exist, the variable is assigned a value of "-1." The variable HPivot serves the same purpose as was described above for Condition2 and Condition4.

```
Lpivot = MRO(Condition1 AND Condition3, SetUpInt, 1);
HPivot = MRO(Condition2 AND Condition4, SetUpInt, 1);
```

The variable EMA is assigned the value of the exponential moving average of the close. The variable TT, represents the Total number of Trades taken by the system. This value will be used later to determine the completion of a trade, after which the breakeven stop should no longer be generated.

```
EMA = XAverage(Close, XAvgLen);
TT = TotalTrades;
```

In order for a buy setup to occur, a series of criteria must be met. The first section verifies the existence of a low pivot, and that the low pivot has been exceeded to the downside by a lower

low. The LPivot variable, with a value greater than "0", indicates the occurrence of a low pivot. The value of the low pivot bar (Low[LPivot+1]) is then compared to the lowest low since the occurrence of the low pivot. If the lowest low is in fact lower than the low pivot, then that second criteria has also been met.

```
IF LPivot > 0 AND Lowest(Low, LPivot) < Low[LPivot+1] Then Begin
```

In addition to the criteria above, a buy setup requires that the True Range be greater than the Average True Range of the last ATRLen bars. This is established simply by using Condition5 (which implies Condition5 = True). Finally, the close must be greater than the close of the previous bar and the current open, it must be in the upper 1/3 of the bar, and be greater than the low pivot bar. The MaxList function is used to return the highest value of all the criteria mentioned above. If Condition5 is true and the current close is greater than all of the MaxList parameters, the high of the setup bar is assigned to the variable BuySetupHigh, the low of the setup bar is assigned to the variable BuySetupLow, and the variable BuySetupCount is reset, thus initiating the buy setup.

```
IF Condition5 AND Close > MaxList(Close[1], Open, Low + (2*(TrueRange/3)), Low[LPivot+1]) Then Begin
        BuySetupHigh = High;
        BuySetupLow = Low;
        BuySetupCount = 0;
End;
```

In order for a sell setup to occur, a series of criteria must also be met. The first section verifies the existence of a high pivot, and that the high pivot has been exceeded to the upside by a higher high. The HPivot variable, with a value greater than "0," indicates the occurrence of a high pivot. The value of the high pivot bar (Low[LPivot+1]) is then compared to the highest high since the occurrence of the high pivot. If the highest high is in fact higher than the high pivot, then that second criteria has also been met.

```
IF HPivot > 0 AND Highest(High, HPivot) > High[HPivot+1] Then Begin
```

In addition to the criteria above, a sell setup requires that the True Range be greater than the Average True Range of the past ATRLen bars. This is established simply by using Condition5 (which implies Condition5 = True). Finally, the close must be less than the close of the previous bar and the current open, it must be in the lower 1/3 of the bar, and be less than the high pivot bar. This time MinList Function is used to return the lowest value of all the criteria mentioned above. If Condition5 is True and the current close is greater than all of the MinList parameters, the low of the setup bar is assigned to the variable SellSetupLow, the high of the setup bar is assigned to the variable SellSetupHigh, and the variable SellSetupCount is reset, thus initiating the sell setup.

```
IF Condition5 AND Close < MinList(Close[1], Open, Low + (TrueRange/3), High[HPivot+1]) Then Begin
        SellSetupLow = Low;
        SellSetupHigh = High;
        SellSetupCount = 0;
End;
```

Breakeven Stops

This section of stops has been placed before the system entries in order to prevent multiple entries as a result of the same setup. Thus, once a long position is taken the current setup (as represented by BuySetupCount) is nullified. When the system is long, as indicated by MarketPosition equal to 1 or an increase in the total trades (TT) from the prior bar, as stated above, the buy setup is nullified.

```
IF MarketPosition = 1 OR TT <> TT[1] Then BeginBuySetupCount = SetUpLen;
```

In addition, if the system is long and the open profits on a closing basis are equal to or greater than 2 (RiskMult) times the initial risk (the initial risk for a long position is the entry price minus the low of the entry bar at which a stop is set), the initial stop is moved up to a breakeven stop.

The LExitFlag variable is set to 0 so that in the event that profits decline, the stop will remain at breakeven.

```
IF Close > EntryPrice + (RiskMult*(EntryPrice - LExitLow)) OR
        LExitFlag = 0 Then Begin
                ExitLong Next Bar at EntryPrice Stop;
                LExitFlag = 0;
        End;
End;
```

Once a short position is taken, the current sell setup (as represented by SellSetupCount) is nullified. When the system is short, as indicated by a MarketPosition equal to -1 or an increase in the total trades (TT) from the prior bar, as stated above, the sell setup is nullified.

```
IF MarketPosition = -1 OR TT <> TT[1] Then BeginSellSetupCount = SetUpLen;
```

In addition, if the system is short and the open profits on a closing basis are equal to or greater than 2 (RiskMult) times the initial risk (the initial risk for a short position is the low of the entry bar at which a stop is set minus the entry price), the initial stop is moved down to a breakeven stop. The SExitFlag variable is set to 0 so that in the event that profits decline, the stop will remain at breakeven.

```
IF Close < EntryPrice - (RiskMult*(SExitHigh - EntryPrice)) OR SExitFlag = 0 Then Begin
        ExitShort Next Bar at EntryPrice Stop;
        SExitFlag = 0;
End;
```

The two counters, BuySetupCount and SellSetupCount, are incremented on each bar in order to keep track of the number of bars in the buy/sell setups.

```
BuySetupCount = BuySetupCount + 1;
SellSetupCount = SellSetupCount + 1;
```

Long Entry

While the BuySetupCount is less than or equal to the SetUpLen, which means that we are within the specified setup period, the criteria for the buy signal is evaluated. Thus, if the above is true and the high of the setup bar (BuySetupHigh) is greater than the exponential moving average, then a buy stop order is placed at BuySetupHigh plus 1 point. When this occurs the BuySetupLow value is assigned to a new variable called LExitLow (low used for the long exit). This is done so that the stop will not change in the event that another buy setup is initiated before the long position is closed out. In addition, the LExitFlag variable is assigned a value of 1, which it will maintain until the position profit exceeds the RiskMult times the initial risk, at which point the breakeven stop will take effect.

```
IF BuySetupCount <= SetUpLen Then Begin
        IF BuySetupHigh + 1 Point > EMA Then Begin
                Buy Next Bar at BuySetupHigh + 1 Point Stop;
                LExitLow = BuySetupLow;
                LExitFlag = 1;
        End;
End;
```

Short Entry

While the SellSetupCount is less than or equal to the SetUpLen, which means that we are within the specified Setup period, the criteria for the sell signal is evaluated. Thus, if the above is true and the low of the setup bar (SellSetupLow) is less than the exponential moving average, then a sell stop order is placed at SellSetupLow minus 1 point. When this occurs the SellSetupHigh value is assigned to a new variable called SExitHigh (high used for the short exit). This is done so that

the stop will not change in the event that another sell setup is initiated before the short position is closed out. In addition, the SExitFlag variable is assigned a value of 1, which it will maintain until the position profit exceeds the RiskMult times the initial risk, at which point the breakeven stop will take effect.

```
IF SellSetupCount <= SetUpLen Then Begin
        IF SellSetupLow + 1 Point < EMA Then Begin
                Sell Next Bar at SellSetupLow - 1 Point Stop;
                SExitHigh = SellSetupHigh;
                SExitFlag = 1;
        End;
End;
```

Initial Long & Short Exits

These are the initial protective stops for the system. The protective stops are in effect while LExitFlag or SExitFlag respectively have a value of 1. Once the position profit exceeds the RiskMult times the initial risk, LExitFlag and SExitFlag respectively will be set to 0 and the breakeven stop will be used instead.

```
IF LExitFlag = 1 Then ExitLong Next Bar at LExitLow - 1 Point Stop;
IF SExitFlag = 1 Then ExitShort Next Bar at SExitHigh + 1 Point Stop;
```

System Exits

These system exits use the highest high and lowest low of the last HiLoStop bars to liquidate the existing position at the new high/low value.

```
ExitLong Next Bar at Highest(High, HiLoStop) Limit;
ExitShort Next Bar at Lowest(Low, HiLoStop) Limit;
```

General System Format

When we apply a system to a chart, we can use the options in the **Format** dialog box to format costs, stops, and properties. We did not enter an amount for slippage and commission, although those costs must be taken into account before a system is traded.

Note: *Remember that commissions are calculated on a per contract/share basis. When you are trading stocks, you would enter the average commission you are charged divided by the number of shares the system is buying and selling.*

In the **Stops** tab, we enabled a Breakeven stop with a floor of $200. Also, a money management stop and a different kind of breakeven stop were written into the EasyLanguage code for this system, as was discussed in the previous section. The money management stop holds the dollar amount per position or per contract/share we are willing to risk before exiting from the position. The breakeven stop specifies the dollar amount of open profit that our position must achieve before we move our stop to our entry price (breakeven).

Note: *When you are trading stocks and you choose the stop option* ***Apply on a per share (contract) basis****, you will type in the number of points you are willing to lose before you exit. When you are trading futures or any other instrument that has a different dollar-point value, you would type the maximum number of dollars you are willing to risk per contract traded.*

In the **Properties** tab, we selected the option **Do not allow multiple entries in the same direction**. If the system is in a long position, and market conditions generate another long entry order, the order is ignored; if the system is in a short position, and market conditions generate another short entry order, the order is ignored.

Testing & Improving

We tested our Pivot Point Support & Resistance System on daily data of U.S. Treasury Bonds from June, 1993, to September, 1998. The system signaled 17 trades, of which 70% were profitable. Figure 20 shows the Portfolio Maximizer Individual Report.

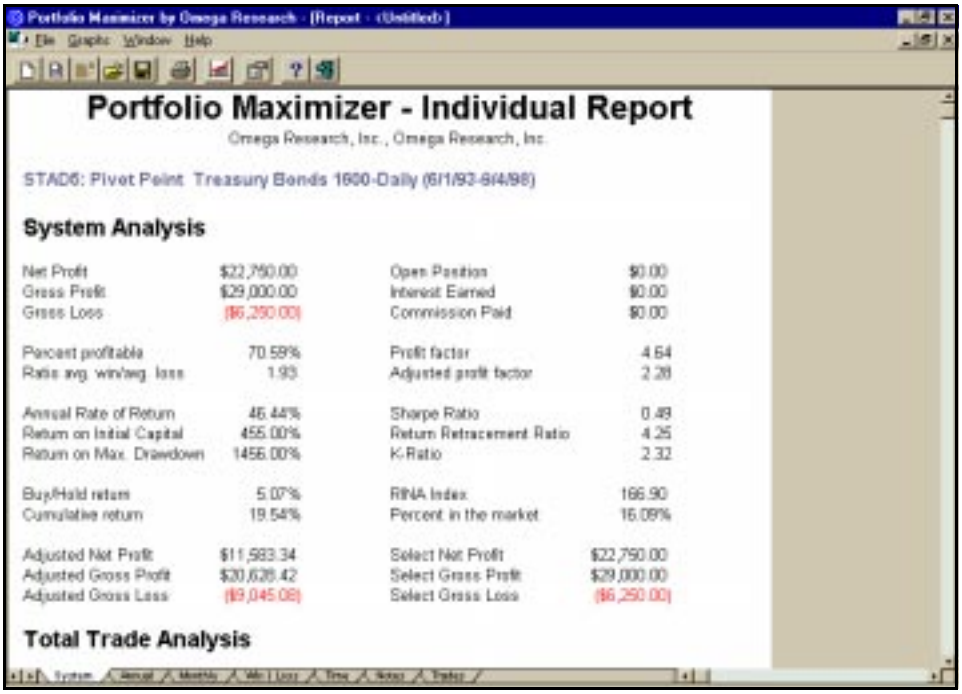


Figure 20. The Portfolio Maximizer Individual Report for thePivot Point Support & Resistance system

The ratio of average win to average loss was 1.93, which is exceptional in a system with 70% winners. The profit factor was an excellent 4.64, indicating that the system earned $4.64 for each dollar it lost.

Five of the six ratios we're emphasizing in this volume far surpassed the reference points for a good system. The Sharpe Ratio was .49 (reference .25), Return Retracement Ratio was 4.25 (reference 3.0), and the RINA Index was 166.90 (reference 30). Net Profit/Largest Loss was 14.56 (reference 7.0) and Net Profit/Maximum Drawdown was also 14.56 (reference 5.0). Only the Return Retracement Ratio at 2.32 failed to achieve its reference point of 3.0.

The Equity Curve is displayed in Figure 21.

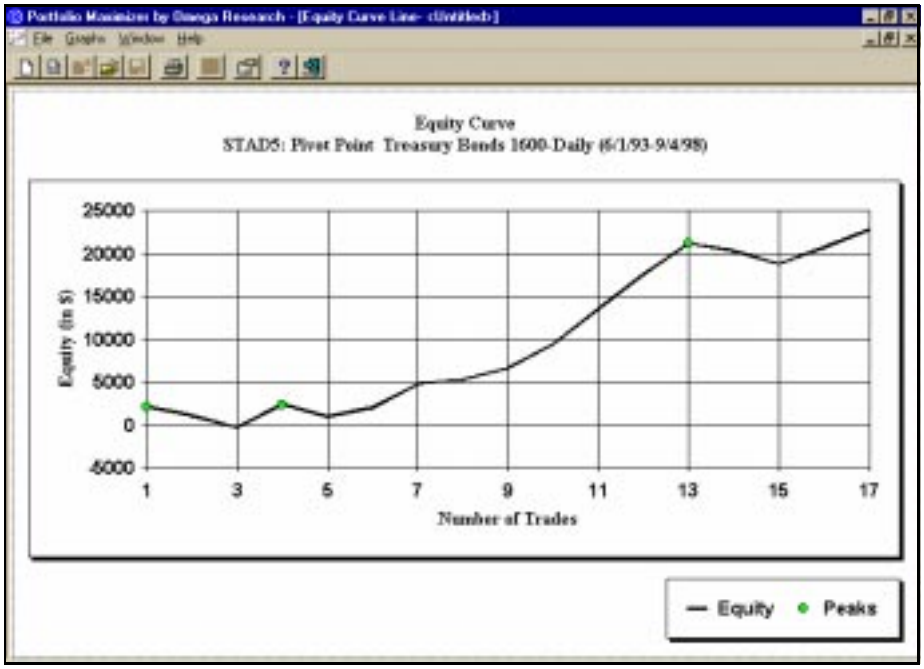


Figure 21. Equity Curve for the Pivot Point Support & Resistance system applied to the daily Treasury Bond chart

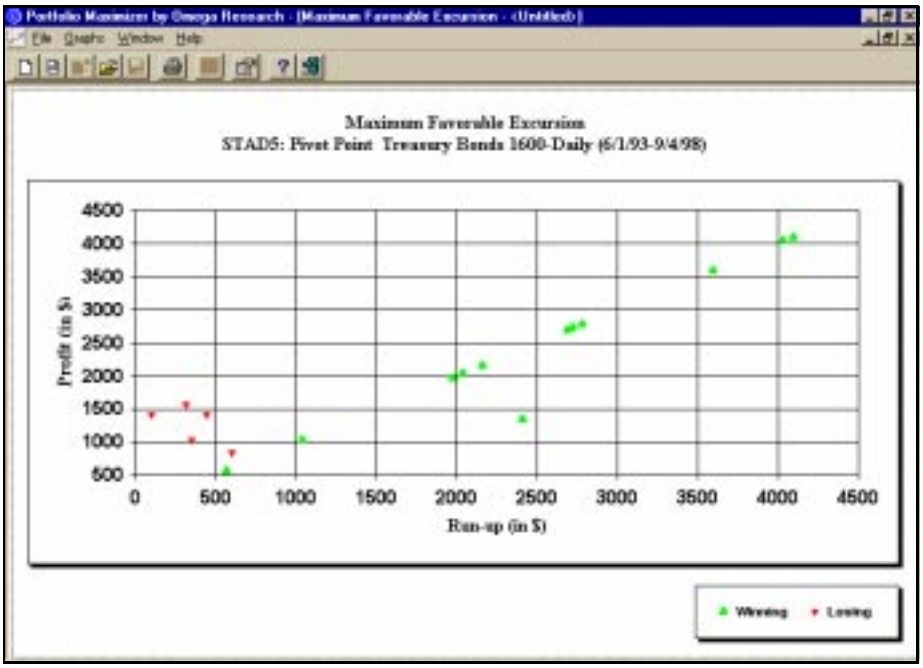
Note that the equity curve has risen to a new peak on trade 17, the last trade in the test period. The equity curve looks strong with profits of almost $25,000 per contract in only 17 trades.

Figure 22 graphs the system's Monthly Rolling Net Profit, which increased sharply from early 1995 to early 1996.

Figure 22. Monthly Rolling Net Profit graph for the Pivot Point Support & Resistance system

This measure of equity growth fluctuated in a disappointingly narrow range (between about $21,000 and $23,000) from early 1996 to the end of August, 1998.

Maximum Favorable Excursion is the topic of Figure 23.



igure 23. Maximum Favorable Excursion graph for the Pivot Point Support & Resistance system

Note that only one losing trade had a run-up of over $500. No other trades that experienced a run-up of greater than $500 reversed into losses. Figure 24 is a chart of Maximum Adverse Excursion.

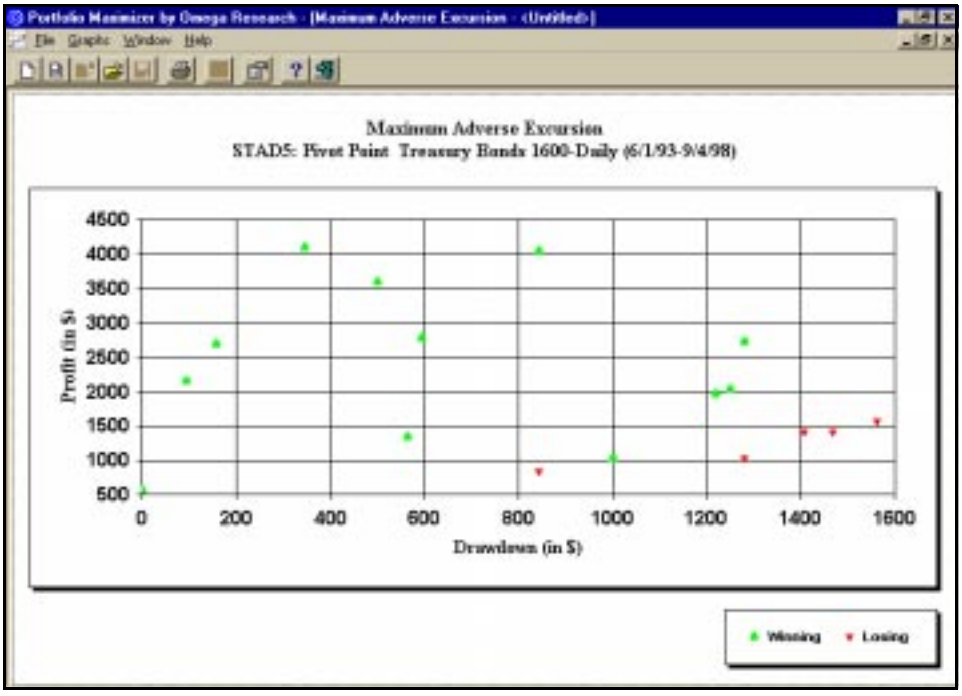


Figure 24. Maximum Adverse Excursion graph for the Pivot Point Support & Resistance system

We weren't surprised that four losing trades had adverse excursions (open losses) of more than $1,200. We *were* surprised, however, that three trades that were closed out with profits of $2,000 or more each endured adverse excursions of more than $1,200.

Average Profit By Month is shown in Figure 25.

Figure 25. Average Profit by Month graph for the Pivot Point Support & Resistance system

Note that when the monthly returns are averaged over the five-year test period, only July and November suffered a measurable loss.

Suggestions for Improvement

This system was tested without either a $ Risk Trailing stop or a % Risk Trailing stop. In effect, each trade could end in one of only three ways: a trade could be exited at the initial protective stop, it could be exited at breakeven, or it could be exited when prices reach the 44-bar high (if long) or the 44-bar low (if short). In other words, the system's strategy as it stands is all or nothing.

It is at least possible that the addition of a trailing stop would improve this system's performance, locking in some profit while still giving the market room to move.

CHAPTER 4

Volatility Breakout Systems

Volatile markets are characterized by sharp jumps in price, and volatility breakout systems are designed to take advantage of this type of change in volatility. Volatility breakout systems generally have the following characteristics:

- Substantial amount of time out of the market.
- High percentage of winning trades, but with a small profit per trade.
- Don't take advantage of big moves.
- Exciting to trade because trades are quick and short-term.
- Based solely on price movement.

When designing this type of system, the key is to effectively anticipate and take advantage of a significant change in volatility and then exit the position before a loss of profit. In this chapter, we present three volatility breakout systems, which, like all volatility breakout systems, are designed to capture significant change in volatility and limit losses during directionless and/or trending phases.

In This Chapter

Currency Four-Day System

Our Currency Four-Day System is based on the premise that if a currency is strong on Friday there will usually be some bullish follow through on Monday. Conversely, if a currency is weak on Friday we should expect some bearish follow through on Monday.

To determine the trend, we'll use a very slow 50-day RSI. If RSI is above 50 on Friday's day-session open, the trend is up; if it's below 50, the trend is down. Our setup to buy, therefore, is that today is Friday, and that the 50-day RSI is greater than 50. To establish our entry point to buy, we'll add 20% of the 10-day average true range (ATR) to Friday's day-session open. Our setup to sell is that today is Friday, and the 50-day RSI is less than 50. To establish our entry point to sell short, we'll subtract 20% of the 10-day ATR from Friday's day-session open.

When we enter a new long position, we'll place a money management stop. Our objective will be to exit on Monday at a return to Friday's day-session high. The protective stop for Monday will be one point below Friday's day-session low. If we're still in a long position after the close on Monday, we'll exit on Tuesday's open.

When we enter a new short position, we'll place a money management stop. Our objective will be to exit on Monday at a return to Friday's day-session low. The protective stop for Monday will be one point above Friday's day-session high. If we're still in a short position after the close on Monday, we'll exit on Tuesday's day-session open. Figure 1 shows the Currency Four-Day system applied to a daily chart of the Deutsche Mark.

Figure 1. The Currency Four-Day system applied to a daily chart of the Deutsche Mark

Defining Your Trading Rules

In this system, we defined both long and short entries as well as exit orders. We also did some setup work to calculate the RSI, the ATR, and 20% of the ATR. The setups, entries, and exits are defined next.

Setup

d) Calculate a 50-day RSI with an overbought level at 70 and an oversold level at 30.

e) Calculate a 10-day ATR (average true range) and 20% of ATR:10.

Long Entries

a) Today is Friday.

b) Check for RSI to be greater than 50.

c) Place a buy stop at the open plus 20% of ATR:10.

Short Entries

a) Today is Friday.

b) Check for RSI to be less than 50.

c) Place a sell stop at the open minus 20% of ATR:10.

Exit Orders

a) When we enter a new position, we'll place a money management stop.

b) Then we'll exit a long position on Monday at a return to Friday's day-session high. We'll exit a short position on Monday at a return to Friday's day-session low.

c) If unable to take profits on Monday at a return to Friday's extreme point, we'll exit Tuesday on the day-session open.

Designing & Formatting

This section presents the EasyLanguage instructions and formatting for the system, with the EasyLanguage instructions broken down and explained line by line.

EasyLanguage Instructions: Currency 4 Day (STAD5: Currency4Day)

```
Input: RSILen(50), Pcnt(.2);

If DayOfWeek(Date) = 5 then begin
        {Long Entry & Exits}
        If RSI(Close, RSILen) > 50 thenBuy next bar at Open next bar + Pcnt*Average(TrueRange, 10) stop;
        If Open of next bar > High then ExitLong next bar at High Stop;
        ExitLong next bar at Low Stop;

        {Short Entry & Exits}
        If RSI(Close, RSILen) < 50 then Sell next bar at Open next bar - Pcnt*Average(TrueRange, 10) stop;
        If Open of next bar < Low thenExitShort next bar at Low Stop;
        ExitShort next bar at High stop;
End;

{Exits}
If MarketPosition <> 0 then Begin
        ExitLong next bar at market;
        ExitShort next bar at market;
End;
```

Inputs

Following is the list of all the inputs we used in this system:

Input	Default	Description
RSILen	50	A length value, expressed in bars, for the RSI
Pcnt	.2	A percentage value of the Average True Range (.2 = 20%)

There are no variables defined for this system.

Setup

This initial criterion determines that the day of week must be Friday in order for the system entries to be evaluated.

```
If DayOfWeek(Date) = 5 then Begin
```

Long Entry

If the RSI is greater than 50, a buy stop order is generated for the next bar. The buy stop order is placed at the open of the next bar plus 20% (Pcnt) of the average true range of the last 10 bars.

```
If RSI(Close, RSILen) > 50 then
        Buy next bar at Open next bar + Pcnt * Average(TrueRange, 10) stop;
```

Long Exit

If the open of the next bar is greater than the high of the current bar, an exit long order will be placed at the high of the current bar. An exit long order will also be placed at the low of the current bar (although it will be meaningless if the exit long order based on the current high is placed).

```
If Open of next bar > High then ExitLong next bar at High Stop;
ExitLong next bar at Low Stop;
```

Short Entry

If the RSI is less than 50, a sell stop order is generated for the next bar. The sell stop order is placed at the open of the next bar minus 20% (Pcnt) of the average true range of the last 10 bars.

```
If RSI(Close, RSILen) < 50 then
        Sell next bar at Open next bar - Pcnt *Average(TrueRange, 10) stop;
```

Short Exit

If the open of the next bar is less than the low of the current bar, an exit short order will be placed at the low of the current bar. An exit short order will also be placed at the high of the current bar (although it will be meaningless if the exit short order based on the current low is placed).

```
        If Open of next bar < Low then ExitShort next bar at Low Stop;
        ExitShort next bar at High stop;
End;
```

Exits

The system will generate a long and short exit for the open of the next bar if the current position is long or short. This exit would executed on the second bar into the position.

```
If MarketPosition <> 0 then Begin
        ExitLong next bar at market;
        ExitShort next bar at market;
End;
```

General System Format

When we apply a system to a chart, we can use the options in the **Format** dialog box to format costs, stops, and properties. We did not enter an amount for slippage and commission, although those costs must be taken into account before a system is traded.

Note: *Remember that commissions are calculated on a per contract/share basis. When you are trading stocks, you would enter the average commission you are charged divided by the number of shares the system is buying and selling.*

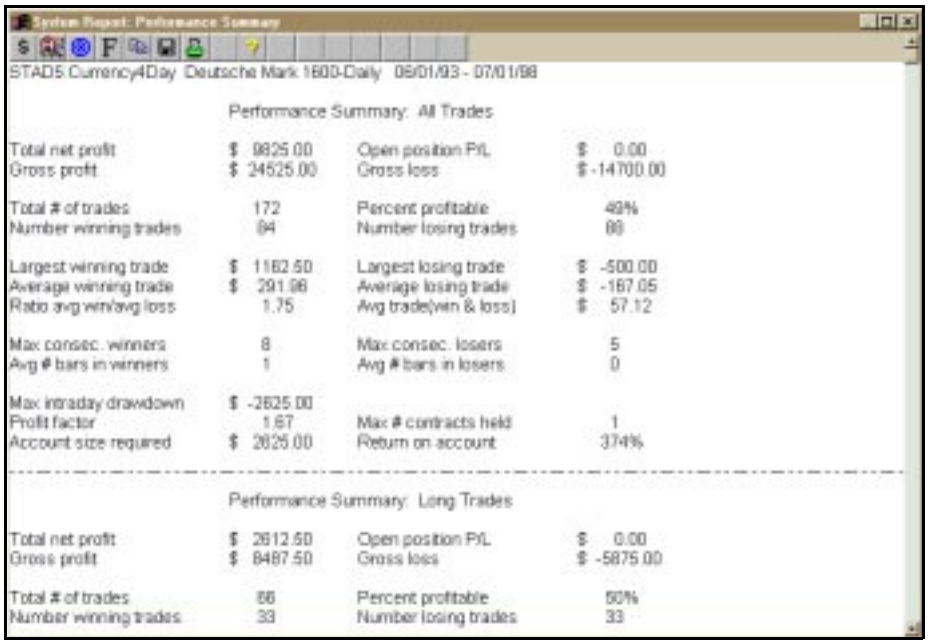
Under the **Stops** tab, we enabled a $200 money management stop for the day of entry. The money management stop option holds the dollar amount per position or per contract/share we are willing to risk before exiting from the position.

Note: *When you are trading stocks and you choose the stop option* ***Apply on a per share (contract) basis****, you will type in the number of points you are willing to lose before you exit. When you are trading futures or any other instrument that has a different dollar-point value, you would type the maximum number of dollars you are willing to risk per contract traded.*

In the **Properties** tab, we selected the option **Do not allow multiple entries in the same direction.** If the system is in a long position, and market conditions generate another long entry order, the order is ignored; if the system is in a short position, and market conditions generate another short entry order, the order is ignored.

Testing & Improving

We tested the Currency Four-Day System on the Deutsche Mark from June, 1993, to July, 1998. Figure 2 shows the System Report; as you can see, the system signaled 172 trades, of which 49% were winners.


System Report: Performance Summary

STAD5 Currency4Day Deutsche Mark 1600-Daily 06/01/93 - 07/01/98

Performance Summary: All Trades

Total net profit	$ 9825.00	Open position P/L	$ 0.00
Gross profit	$ 24525.00	Gross loss	$ -14700.00
Total # of trades	172	Percent profitable	49%
Number winning trades	84	Number losing trades	88
Largest winning trade	$ 1162.50	Largest losing trade	$ -500.00
Average winning trade	$ 291.96	Average losing trade	$ -167.05
Ratio avg win/avg loss	1.75	Avg trade(win & loss)	$ 57.12
Max consec. winners	8	Max consec. losers	5
Avg # bars in winners	1	Avg # bars in losers	0
Max intraday drawdown	$ -2625.00		
Profit factor	1.67	Max # contracts held	1
Account size required	$ 2625.00	Return on account	374%

Performance Summary: Long Trades

Total net profit	$ 2812.50	Open position P/L	$ 0.00
Gross profit	$ 8487.50	Gross loss	$ -5675.00
Total # of trades	66	Percent profitable	50%
Number winning trades	33	Number losing trades	33

Figure 2. System Report for the Currency Four-Day system applied to a daily chart of the Deutsche Mark

The system provided a net profit of $9,825 per contract. The ratio of average win to average loss was 1.75, and the profit factor (the number of dollars won per dollar lost) was 1.67. The Sharpe Ratio, the Return Retracement Ratio, and the K-Ratio fell a little below our reference points, but the RINA Index at 514 (compared to a reference point of 30), the Net Profit/Largest Loss at 19.65 (compared to a guideline of 7.0), and Net Profit/Maximum Drawdown at 12.28 (against

a guideline of 5.0) were excellent. The Equity Curve for our Currency Four-Day System is shown in Figure 3.

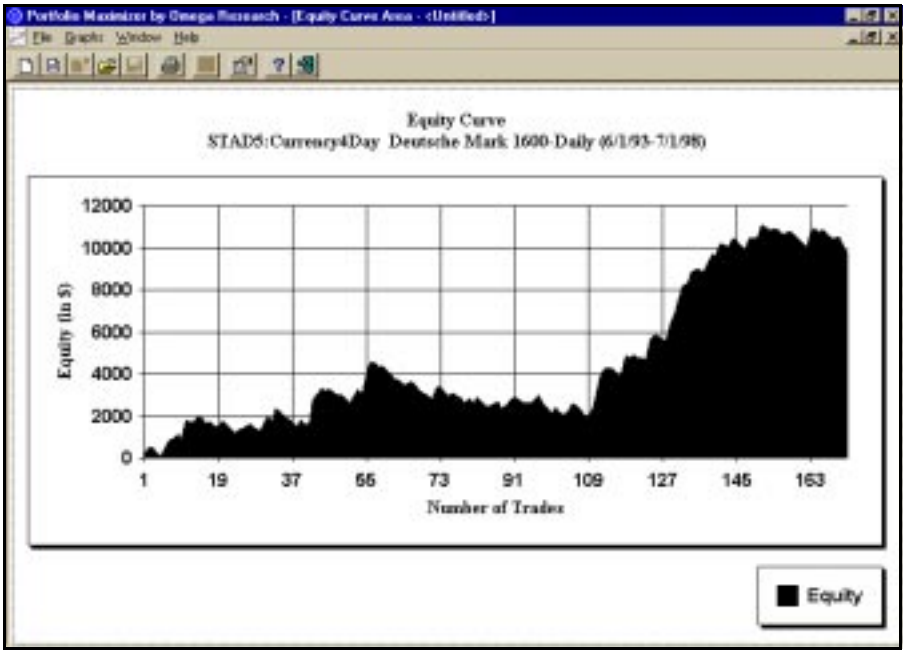


Figure 3. Portfolio Maximizer's Equity Curve for the Currency Four-Day system

Profits increased gradually to about $4,000 after 55 trades, but decreased to about $2,000 after 109 trades. Then the system began to perform well, earning profits of almost $12,000 per contract by trade 172.

Figure 4, the Monthly Rolling Net Profit, shows the accumulation of profits (both realized and unrealized) on a mark-to-market end-of-month basis.

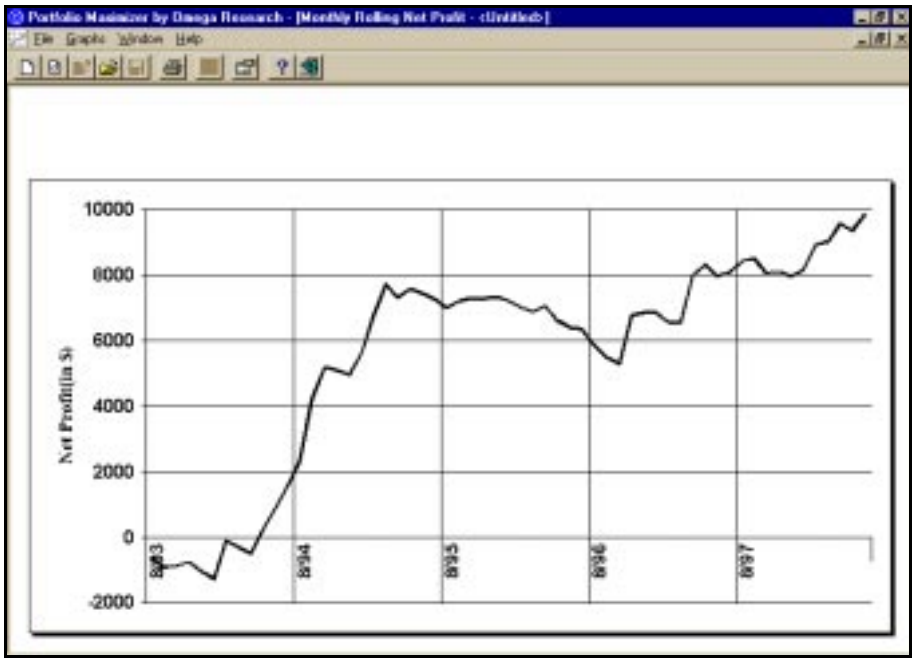
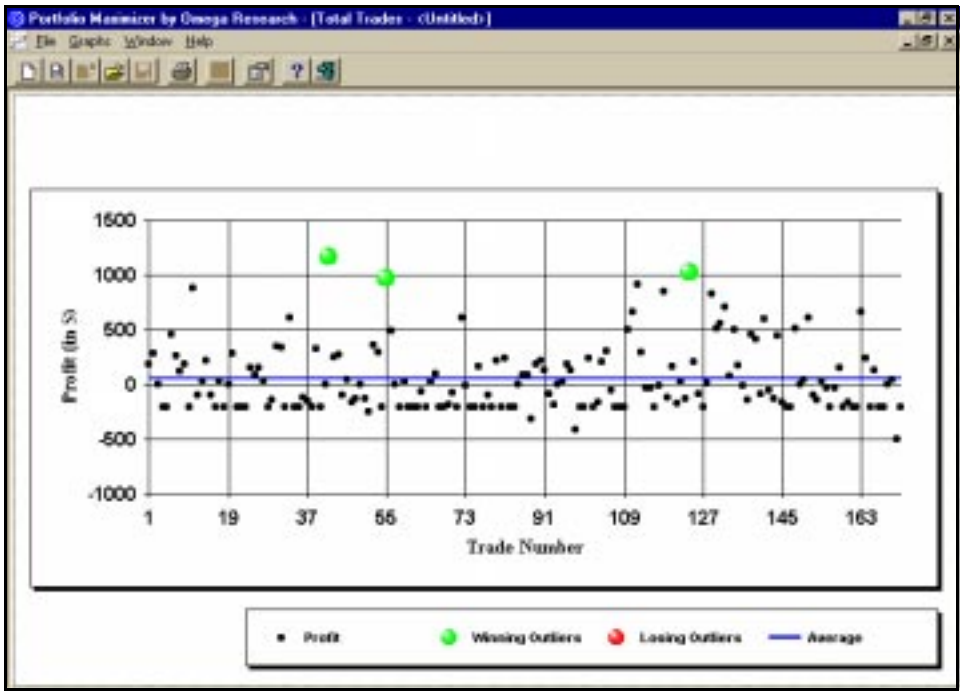


Figure 4. The Monthly Rolling Net Profit graph for the Currency Four-Day system

The equity run-up from August, 1993, to July, 1998, was interrupted by a drawdown period of almost two years. Monthly Rolling Net Profit finished powerfully, however, with an equity increase from about $5,500 per contract to almost $10,000 at the end of June, 1998.

Total Trades are shown in a scatter graph format in Figure 5.



igure 5. Total Trades grsph for the Currency Four-Day system

Note that there were three positive outliers and no negative outliers. The three winning outliers made a huge contribution to the profitability of this system, illustrating the absolute necessity of following the system with perfect discipline. Missing even one of the winning outliers would have been very detrimental to the system's bottom line. A system trader must take *all* the signals generated by his or her system because it is impossible to know in advance which signals will result in the winning outliers.

Also note that many of the trades show a loss of $200. Our optimization for the most profitable initial protective stop indicated that $200 was the best amount to risk. Although many trades were stopped out with a loss of $200, most of those trades would have been stopped out even if we had applied a wider stop.

Suggestions for Improvement

We designed the Currency Four-Day System to initiate a position in the direction of the major trend (RSI:50) and the minor trend (a 20% move off the open) on a Friday to hold the position over the weekend and to take a "business man's" profit (a reasonable, quick profit) on Monday, if possible, or on Tuesday at the latest. Maybe the system could be improved by letting profits run with a trailing stop instead of exiting automatically on Monday or Tuesday.

Channel Breakout with Pullback System

One of the best known and most profitable trading systems is the price channel breakout. Let's use a 10-bar price channel as an example. The top line of the channel is simply the high of the last 10 bars; the bottom line of the channel is the low of the last 10 bars. Buy on a close above the top line and sell short on a close below the bottom line. The price channel can be used in this manner as a stop-and-reverse system, meaning you would always have a position in the market, either long or short.

Of course, the price channel breakout system comes with a disadvantage. Not all closes above the 10-bar high or below the 10-bar low follow through in the direction of the breakout. Frequently, the breakout of the 10-bar high or low is about as far as the price swing carries, resulting in many buys at short-term tops and many sells at short-term bottoms.

Our Channel Breakout with Pullback System attempts to mitigate this disadvantage by waiting for a pullback (a countertrend reaction) after a breakout to gain a more favorable entry price. For example, in the case of a bullish breakout, we would wait to enter on an ensuing decline rather than buying automatically after a close above the previous 10-bar high. The downside to waiting for a pullback is that a market may not pull back after a breakout, causing us to miss a trading opportunity. Figure 6 shows the Channel Breakout with Pullback system applied to a daily Treasury bond chart.

Figure 6. The Channel Breakout with Pullback system applied to a daily Treasury Bond chart

Defining Your Trading Rules

In this system, we defined both long and short entries as well as exit orders. We also did some setup work to construct the price channel. The setups, entries, and exits are defined next.

Setup

a) Calculate a 10-bar price channel of highs and lows.

Long Entries

a) Check for a close above the top line of a 10-bar price channel.

b) Check for a new four-bar low within 10 bars of the breakout.

c) Buy on the next open.

Short Entries

a) Check for a close below the bottom line of a 10-bar price channel.

b) Check for a new four-bar high within 10 bars of the breakout.

c) Sell short on the next open.

Exit Orders

a) We'll enable a money management stop, a breakeven stop, and a % Risk Trailing stop.

b) We'll also exit a long position on a rally to a new eight-bar high and a short position on a decline to a new eight-bar low.

Designing & Formatting

This section presents the EasyLanguage instructions and formatting for the system, with the EasyLanguage instructions broken down and explained line by line.

EasyLanguage Instructions: Channel Breakout with Pullback (STAD5: ChanBrkOut)

```
Input: Channel(10), RetrBars(10), NewLo_Hi(4), TrailStp(8);
Vars: HiChannel(0), LoChannel(0);

{Assignment of Channel calculations}
HiChannel = Highest(Close, Channel)[1];
LoChannel = Lowest(Close, Channel)[1];

{Long Entry}
If MRO(Close > HiChannel, RetrBars, 1)>-1 AND Low < Lowest(Low, NewLo_Hi)[1] then
        Buy next bar at Market;

{Short Entry}
If MRO(Close < LoChannel, RetrBars, 1)>-1 AND High > Highest(High, NewLo_Hi)[1] then
        Sell next bar at Market;

{Exits}
If High > Highest(High,TrailStp)[1] then Exitlong this bar on close;
If Low < Lowest(Low,TrailStp)[1] then Exitshort this bar on close;
```

Inputs

Following is the list of all the inputs we used in this system:

Input	Default	Description
Channel	10	A length value, expressed in bars, used in the Highest and Lowest functions which form the basis of the channels.
RetrBars	10	Retracement bars is a length value, expressed in bars, used to define the maximum number of bars after the retracement at which a signal can occur.
NewLo_Hi	4	A length value, expressed in bars, for the new Highest/Lowest value following a valid retracement.
TrailStop	8	A length value, expressed in bars, for the trailing stop.

In addition to these inputs, we define the following variables:

```
Vars: HiChannel(0), LoChannel(0);
```

Setup

The channels are calculated and assigned to the HiChannel and LoChannel variables. Notice that the channel calculations are offset by 1 bar. This is because we are evaluating breakouts beyond the upper/lower channel boundaries. Since the Highest and Lowest functions include the current bar in their calculations, and the high will never be higher than itself, me must include the offset.

```
HiChannel = Highest(Close, Channel)[1];
LoChannel = Lowest(Close, Channel)[1];
```

Long Entry

The system will generate a long entry at the open of the next bar if the close has exceeded the upper channel boundary, within the specified retracement period (RetrBars), and the current low qualifies as a new low. The MRO Function is used to perform the evaluation for a close that has exceeded the upper boundary within the specified retracement period. In order to identify a new low, the current low is compared to the most recent lowest low for the previous NewLo_Hi bars. Note that we are again referring to previous low values.

```
If MRO(Close > HiChannel, RetrBars, 1)>-1 AND
        Low < Lowest(Low, NewLo_Hi)[1] thenBuy next bar at Market;
```

Short Entry

The system will generate a short entry at the open of the next bar if the close has broken below the lower channel boundary, within the specified retracement period (RetrBars), and the current high qualifies as a Nnew High. The MRO Function is used to perform the evaluation for a close that has broken below the lower boundary within the specified retracement period. In order to identify a new high, the current high is compared to the most recent highest high for the previous NewLo_Hi bars. Note that we are again referring to previous high values.

```
If MRO(Close < LoChannel, RetrBars, 1)>-1 AND
High > Highest(High, NewLo_Hi)[1] thenSell next bar at Market;
```

Trailing Stop

We will exit any long position on the close if the current high exceeds the highest high for the previous TrailStp bars. We will exit any short position on the close if the current low exceeds the lowest low ofr the previous TrailStp bars.

```
If High > Highest(High,TrailStp)[1] then Exitlong this bar on close;
If Low < Lowest(Low,TrailStp)[1] thenExitShort this bar on close;
```

General System Format

When we apply a system to a chart, we can use the options in the **Format** dialog box to format costs, stops, and properties. We did not enter an amount for slippage and commission, although those costs must be taken into account before a system is traded.

Note: *Remember that commissions are calculated on a per contract/share basis. When you are trading stocks, you would enter the average commission you are charged divided by the number of shares the system is buying and selling.*

Under the **Stops** tab, we enabled a $1,000 money management stop, and a breakeven stop with a floor of $200. The money management stop option holds the dollar amount per position or per contract/share we are willing to risk before exiting from the position. The Breakeven stop allows us to specify the dollar amount of open profit that our position must achieve before we move our stop to our entry price (breakeven).

Note: *When you are trading stocks and you choose the stop option* ***Apply on a per share (contract) basis****, you will type in the number of points you are willing to lose before you exit.*

When you are trading futures or any other instrument that has a different dollar-point value, you would type the maximum number of dollars you are willing to risk per contract traded.

In the **Properties** tab, we selected the option **Do not allow multiple entries in the same direction.** If the system is in a long position, and market conditions generate another long entry order, the order is ignored; if the system is in a short position, and market conditions generate another short entry order, the order is ignored.

Testing & Improving

We tested our Channel Breakout with Pullback System on daily data of U.S. Treasury Bonds from June, 1993, to September, 1998. The system generated 220 trades, with 69% of them profitable. Figure 7 shows the resulting Portfolio Maximizer Individual Report.

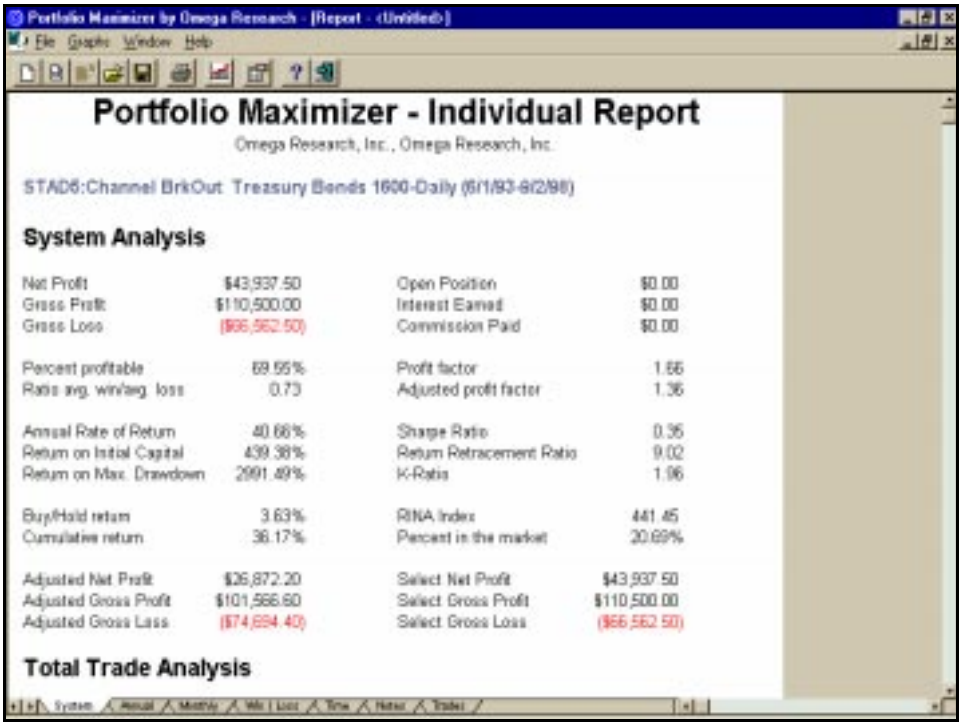


Figure 7. Portfolio Maximizer Individual Report for the Channel Breakout with Pullback system

The ratio of average win to average loss was only .73, meaning that the average winning trade was only 73% as large as the average losing trade. That's not too bad, though, for a support and resistance system with a high winning percentage. The profit factor of 1.66 indicates that the system made $1.66 in profits for each dollar it lost.

Five of the six ratios we're studying in this volume exceeded our reference points for a good system. Only the K-Ratio (at 1.96) was less than its guideline of 2.5.

A Sharpe Ratio of .35, Return Retracement Ratio of 9.02, and RINA Index of 441.45 significantly eclipsed their standards of .25, 3.0, and 30.0, respectively.

A Net Profit/Largest Loss of 29.91 and Net Profit/Maximum Drawdown (also 29.91) were much better than our reference points of 7.0 and 5.0.

Figure 8 is an Equity Chart by Bar for our Channel Breakout with Pullback System.

Figure 8. Equity Chart by Bar for the Channel Breakout with Pullback system

The chart presents a day-by-day depiction of our equity growth. It's no wonder that five of our six performance ratios trounced their guidelines. This equity curve looks great! Monthly Net Profit is displayed in Figure 9.

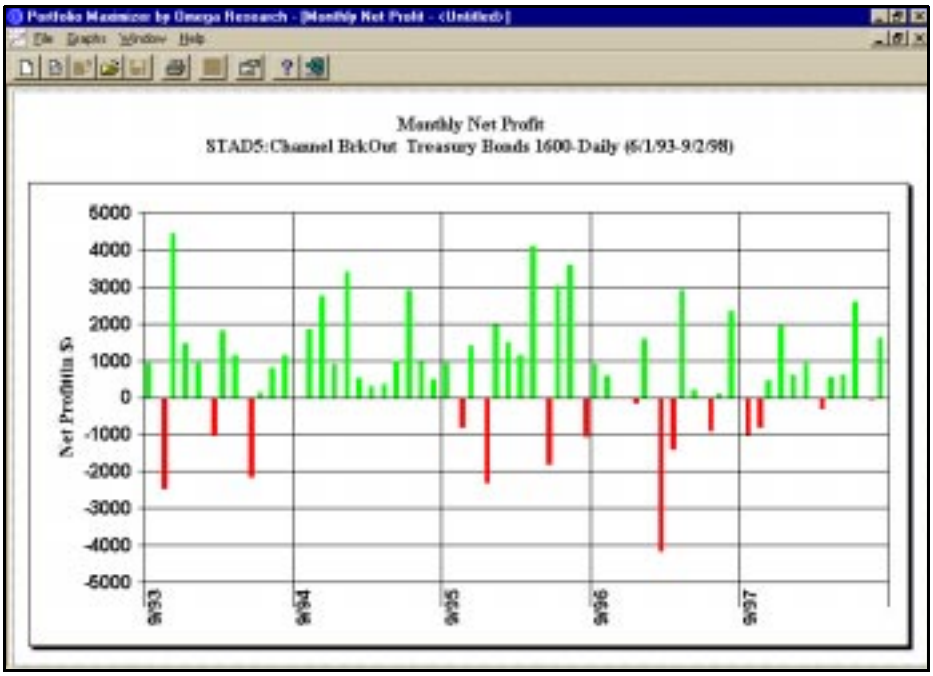


Figure 9. Monthly Net Profit graph for the Channel Breakout with Pullback system

The Monthly Net Profit shows that over the test period 43 months were profitable while only 13 months lost money. The system produced 15 consecutive winning months versus only two consecutive losing months.

Figure 10 depicts Average Profit By Month.

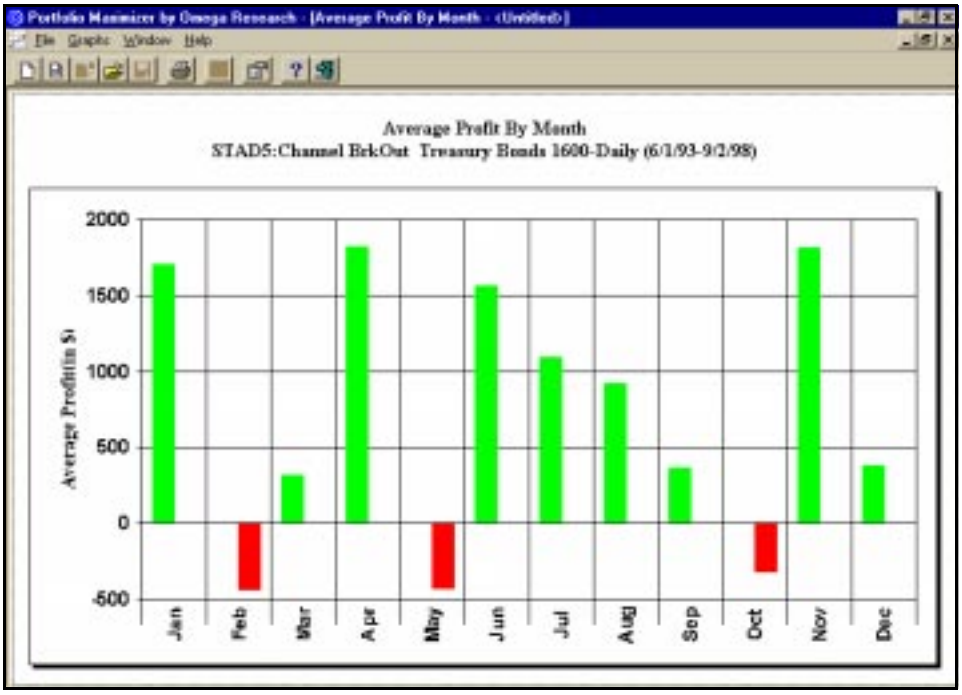


Figure 10. Average Profit by Month graph for the Channel Breakout with Pullback system

Averaged over the five-year test period, nine months were profitable, and only three were not. Also note that our system earned more than $1,500 per contract in four of the months, while losing less than $500 per contract in the three negative months.

The Underwater Equity Curve is graphed in Figure 11.

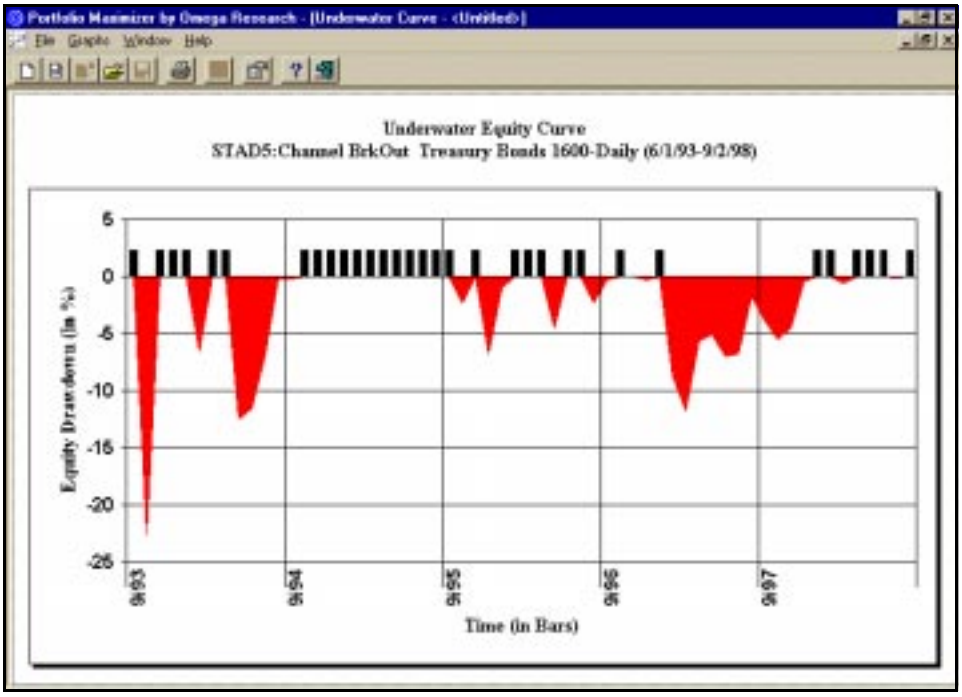


Figure 11. Underwater Equity curve for the Channel Breakout with Pullback system

It shows a drawdown of about 22% between the first and second equity peaks. The other drawdowns never exceeded 12%. An amazing feature of the chart is that the system produced 12 consecutive new monthly equity highs between October, 1994, and September, 1995.

Figure 12 shows our system's drawdown on a trade-by-trade basis.

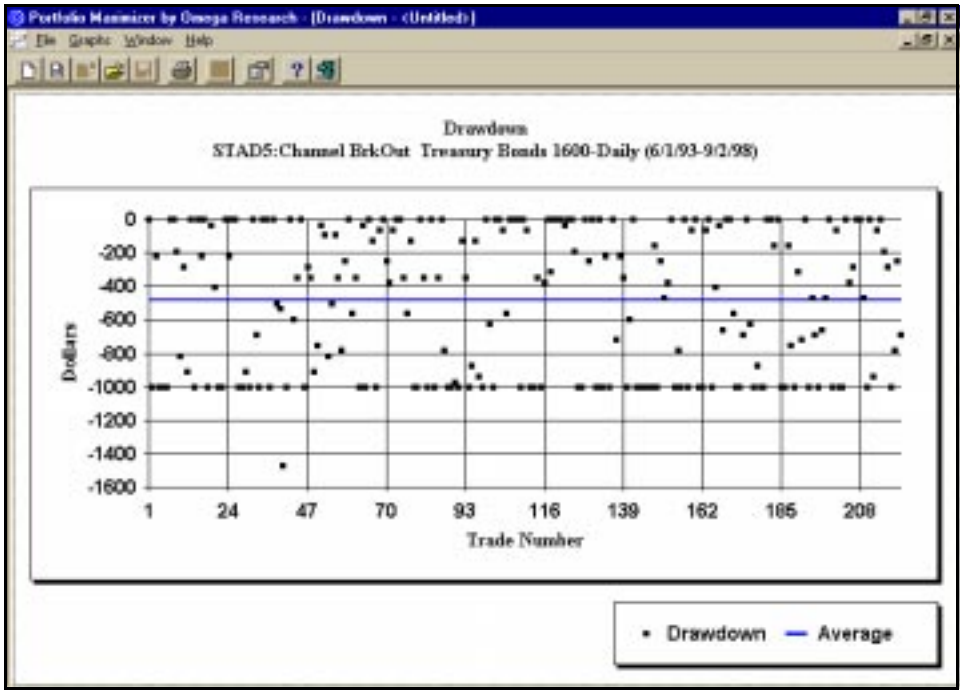


Figure 12. Drawdown graph for the Channel Breakout with Pullback system

Many of the trades posted zero drawdown, and the average drawdown of all trades (both winners and losers) was only about $450. Note that many trades incurred a drawdown of $1,000, which was the amount of our money management stop.

Suggestions for Improvement

This system looks very promising, but there is a least one area where we can probably make it even better. Notice once again that we tested a system with a good setup but didn't specify a separate entry technique.

As it stands, the system simply enters a new trade at the market on the next open after the conditions for the setup have been met. It is very unusual to see a system that can not be improved by adding a carefully designed entry technique to the setup.

Since our Channel Breakout with Pullback System is classified as a volatility system because of the breakout requirement, let's add another volatility requirement as the entry technique. After a buy setup, add 125% of the 10-bar ATR (average true range) to the close of the setup bar. That volatility expansion is the entry point. After a setup to sell, subtract 125% of ATR:10 from the setup bar's close to determine our volatility-expansion entry point. For both long and short trades, keep the entry point active for four bars. If not stopped into a trade by the fourth bar after the setup, the signal is cancelled.

Displaced Standard Deviation Bands System

Bands and channels are popular and effective tools for the technical analyst. There are several different types of bands and channels in wide use: moving average channels of highs and lows, moving average envelopes, Keltner Channels, price channels (*n*-day high, *n*-day low), Bollinger Bands, etc.

Bands and channels can be used to identify support and resistance or to identify breakouts from trading ranges. For example, Bollinger Bands, which are generally plotted two standard deviations above and below a moving average, serve as support and resistance, containing more than 90% of a market's fluctuations. In other words, almost all price action occurs between the top and bottom Bollinger Bands.

Price channels, on the other hand, are usually employed to indicate breakouts. For example, a 20-bar price channel plots the highest high and lowest low of the past 20 bars. Although a price channel can be used in different ways, most traders buy breakouts to new *n*-bar highs and sell breakouts to new *n*-bar lows.

Our system begins with a 3-bar simple moving average of closes and a 15-bar standard deviation of closes. Then we multiply the standard deviation by 2, adding that amount to the moving average to construct the top band and subtracting it from the moving average to construct the bottom band. Our next step is to displace the bands forward in time by 16 bars. In other words, we take the standard deviation bands that we've calculated bar-by-bar and slide the bands 16 bars to the right. Displacement may reduce the number of "whipsaws" (false signals) and may contain price action better than non-displaced indicators.

In our Displaced Standard Deviation Bands System, we enter long on a penetration of the top band and sell short on a penetration of the bottom band. This is a simple stop-and-reverse system, always in the market, either long or short. Figure 13 shows the Displaced Standard Deviation Bands system applied to a daily chart of Johnson & Johnson.

Figure 13. Daily chart of Johnson & Johnson with the Displaced Standard Deviation Bands system applied

Defining Your Trading Rules

In this system, we defined both long and short entries as well as exit orders. We also did some setup work to calculate the moving average of closes and the standard deviation of closes and to construct the displaced standard deviation bands. The setups, entries, and exits are defined next.

Setup

a) Calculate a 3-bar simple moving average of closes.

b) Calculate a 15-bar standard deviation of closes.

c) Multiply the standard deviation by two.

d) Add two times the standard deviation to the moving average to form the top band and subtract it from the moving average to form the bottom band.

e) Displace the bands forward in time by 16 bars.

Long Entries

a) Check for a penetration of the top band.

b) Buy at the market.

Short Entries

a) Check for a penetration of the bottom band.

b) Sell short at the market.

Exit Orders

c) This is a stop-and-reverse system. We exit a long position and sell short when the short entry is reached, and we exit a short position and go long when the buy entry is reached.

Designing & Formatting

This section presents the EasyLanguage instructions and formatting for the system, with the EasyLanguage instructions broken down and explained line by line.

EasyLanguage Instructions: Displaced Standard Deviation Bands (STAD5: Disp Std Dev)

```
Inputs: Price(Close), AvgLen(3), Disp(16), SDLen(12), SDev(2);
Vars: AvgVal(0), SDmult(0), DispTop(0), DispBottom(0);

{Assignment of Displaced Channel Calculations}
AvgVal = Average(Price, AvgLen);
SDmult = StdDev(Price, SDLen) * SDev;
DispTop = AvgVal[Disp] + SDmult;
DispBottom = AvgVal[Disp] - SDmult;

{System Entries}
IF CurrentBar > Disp Then Begin
        Buy Next Bar at DispTop Stop;
        Sell Next Bar at DispBottom Stop;
End;
```

Inputs

Following is the list of all the inputs we used in this system:

Input	Default	Description
Price	Close	Price used to calculate the moving average and the standard deviation (on which the channels will be based).
AvgLen	3	Length, expressed in bars, used to calculate the moving average.
Disp	16	Number of bars that the moving average is displaced.
SDLen	12	Length, expressed in bars, used to calculate the standard deviation.

In addtion to these inputs, we define the following variables:

```
Vars: AvgVal(0), SDmult(0), DispTop(0), DispBottom(0);
```

Setup

We calculate the moving average and store the resulting value in the variable AvgVal. The standard deviation of the closing prices is calculated and multiplied by 2, and the resulting value is stored in the variable SDmult.

```
AvgVal = Average(Price, AvgLen);
SDmult = StdDev(Price, SDLen) * SDev;
```

The SDmult is added to AvgVal and assigned to the variable DispTop. Conversely, the SDmult value is subtracted from AvgVal and assigned to the variable DispBottom.

```
DispTop = AvgVal[Disp] + SDmult;
DispBottom = AvgVal[Disp] - SDmult;
```

Long & Short Entries

Once there are more than 16 bars (as specified by the Disp input), the system will generate a buy stop order at the DispTop value and a sell stop order at the DispBottom value.

```
IF CurrentBar > Disp Then Begin
        Buy Next Bar at DispTop Stop;
        Sell Next Bar at DispBottom Stop;
End;
```

General System Format

When we apply a system to a chart, we can use the options in the **Format** dialog box to format costs, stops, and properties. We did not enter an amount for slippage and commission, although those costs must be taken into account before a system is traded, nor did we enable any stops for this system.

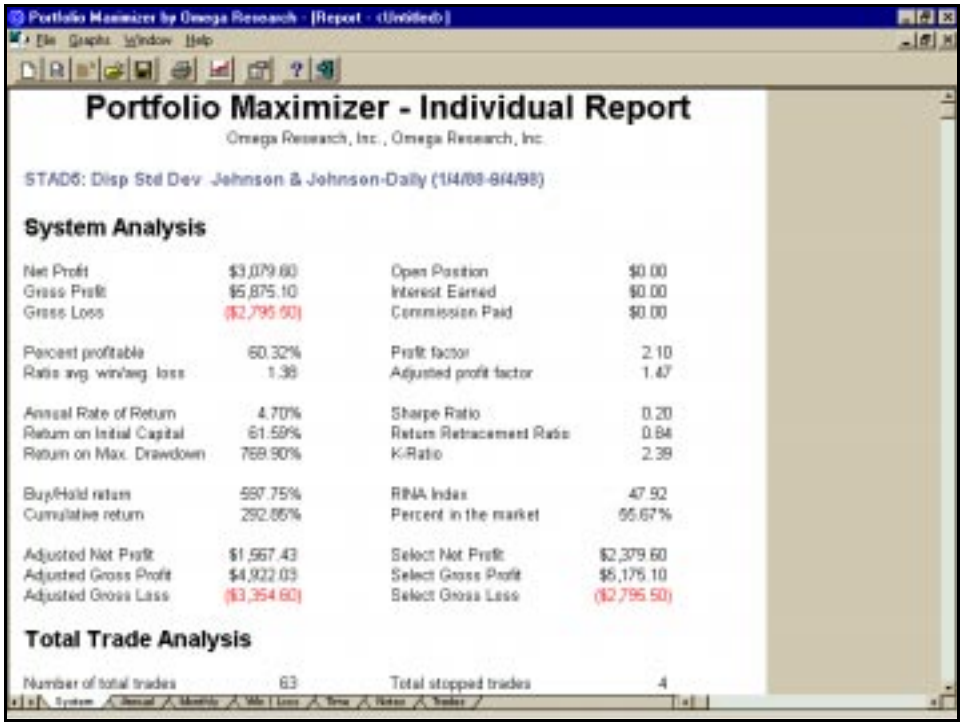
Note: *Remember that commissions are calculated on a per contract/share basis. When you are trading stocks, you would enter the average commission you are charged divided by the number of shares the system is buying and selling.*

We did not enable any stops for this system.

In the **Properties** tab, we selected the option **Do not allow multiple entries in the same direction.** If the system is in a long position, and market conditions generate another long entry order, the order is ignored; if the system is in a short position, and market conditions generate another short entry order, the order is ignored.

Testing & Improving

We tested our Displaced Standard Deviation Bands system on daily data of Johnson & Johnson from January, 1988, to September, 1998. The system generated 63 trades, of which 60% were profitable. Figure 14 shows the Portfolio Maximizer Individual Report.



Portfolio Maximizer - Individual Report

Omega Research, Inc., Omega Research, Inc.

STAD6: Disp Std Dev Johnson & Johnson-Daily (1/4/88-8/4/98)

System Analysis

Net Profit	$3,079.60	Open Position	$0.00
Gross Profit	$5,875.10	Interest Earned	$0.00
Gross Loss	($2,795.50)	Commission Paid	$0.00
Percent profitable	60.32%	Profit factor	2.10
Ratio avg. win/avg. loss	1.38	Adjusted profit factor	1.47
Annual Rate of Return	4.70%	Sharpe Ratio	0.20
Return on Initial Capital	61.59%	Return Retracement Ratio	0.84
Return on Max. Drawdown	769.90%	K-Ratio	2.39
Buy/Hold return	597.75%	RINA Index	47.92
Cumulative return	292.66%	Percent in the market	66.67%
Adjusted Net Profit	$1,567.43	Select Net Profit	$2,379.60
Adjusted Gross Profit	$4,922.03	Select Gross Profit	$5,175.10
Adjusted Gross Loss	($3,354.60)	Select Gross Loss	($2,795.50)

Total Trade Analysis

Number of total trades	63	Total stopped trades	4

Figure 14. The Portfolio Maximizer Individual Report for the Displaced Standard Deviation Bands system

The ratio of average win to average loss was 1.38, and the profit factor (the number of dollars won for each dollar lost) was 2.10.

The Sharpe Ratio was a fraction low at .20 (compared to a reference point of .25), and the Return Retracement Ratio fell below its guideline at .84 (reference 3.0). The K-Ratio just missed reaching its reference point at 2.39 (reference 2.5), while the RINA Index was strong at 47.92 (reference 30). Net Profit/Largest Loss and Net Profit/Maximum Drawdown both exceeded our guidelines of 7.0 and 5.0, respectively.

The system's Equity Curve is shown in Figure 15.

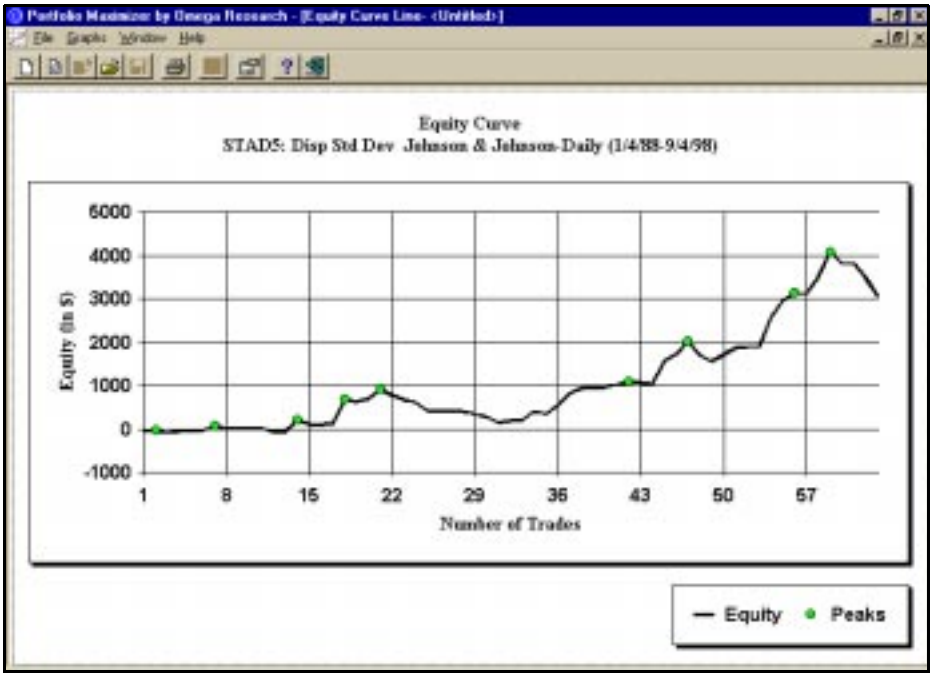


Figure 15. Equity Curve for the Displaced Standard Deviation Bands system

Although the equity curve was strong between trades 35 and 59, it was flat throughout the first 14 trades and declined from trades 21 to 31. It appears that only an historic bull market enabled this system to earn significant profits. Figure 16 displays the Monthly Rolling Net Profit.

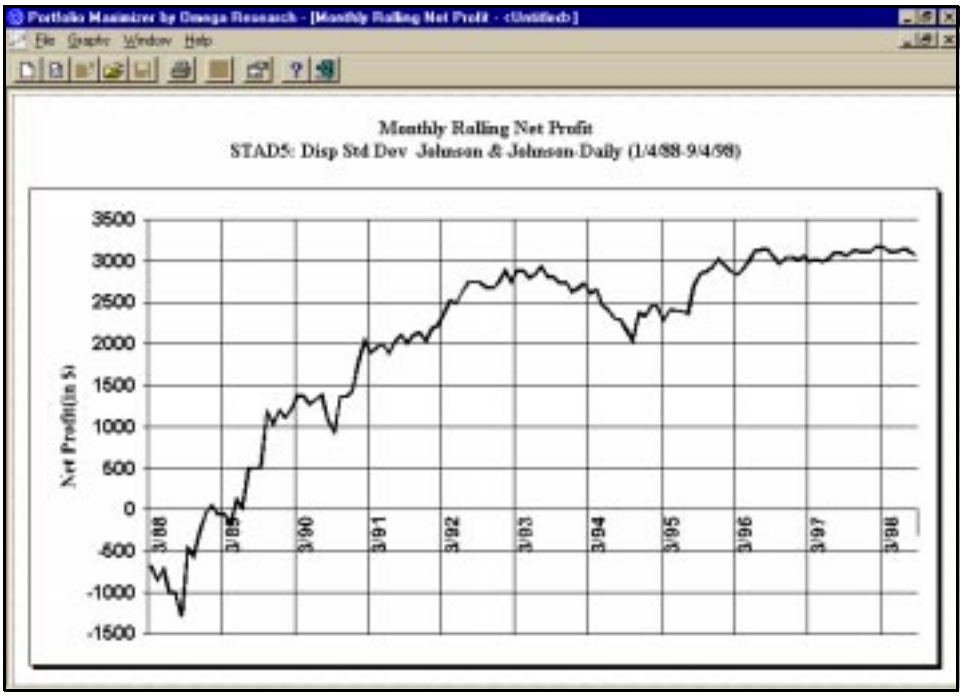


Figure 16. Monthly Rolling Net Profit graph for the Displaced Standard Deviation Bands system

On a mark-to-market at end-of-month basis, the system faltered at first, performed well until mid-1993, declined for about a year, rallied to a new equity high in mid-1996, and flattened out through mid-1998. We're not pleased with the year-long drawdown or the two-year flat period.

Average Profit By Month is graphed in Figure 17.

Figure 17. Average Profit By Month graph for the Displaced Standard Deviation Bands system

When the monthly equity is averaged over the ten-year test period, only three months failed to return a positive result (March, May, and August). The amount of money lost in August, the worst-performing month, was equaled or exceeded in each of the three best months (January, April, and October).

Figure 18 shows Maximum Favorable Excursion.

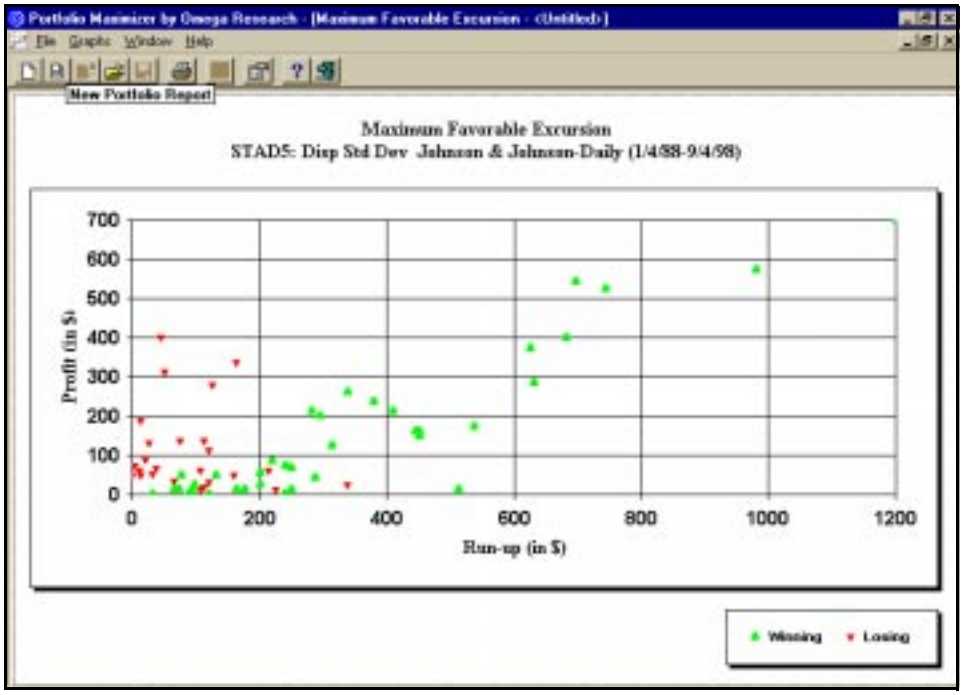


Figure 18. The Maximum Favorable Excursion graph for the Displaced Standard Deviation Bands system

Note that no losing trades had a run-up of more than approximately $325 (per hundred shares traded). All the trades that reached an open profit of $325 or more remained winners.

Maximum Adverse Excursion is depicted in Figure 19.

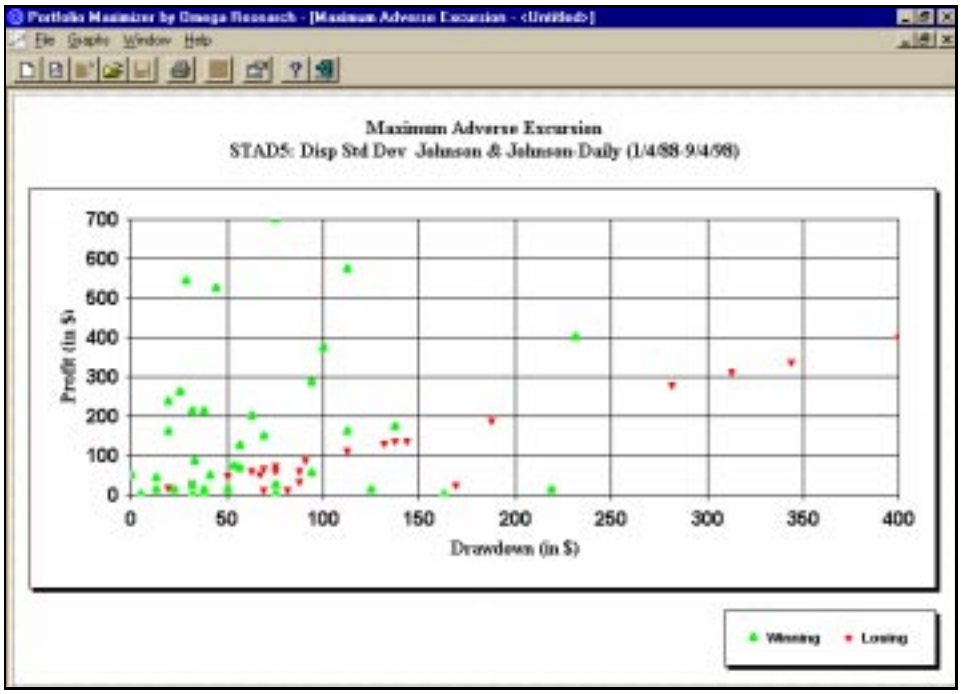


Figure 19. The Maximum Adverse Excursion graph for the Displaced Standard Deviation Bands system

The interesting feature here is that all trades that posted an open loss of more than $275 (per hundred shares traded) ended up as losers. No trades that were closed out with a profit exceeded a drawdown of about $235. A money management stop of $250 would have worked extremely well during the test period.

Suggestions for Improvement

This system was written and tested without stops or an exit strategy (other than reversing the position from long to short or from short to long). The system also lacked an entry technique that would require some confirmation of the setup (the penetration of one of the bands) before a new trade was entered. This is, in fact, how we usually test a setup to determine if our trading idea has enough promise to pursue further. We think the performance of this setup warrants additional time and effort.

Can you improve this system's performance by adding an entry requirement, a money management stop, a breakeven stop, a trailing stop, and an exit strategy? (We bet you can!)

APPENDIX A

Volume in Review

Each volume of the *System Trading and Development Club* provides feedback on the previous volume. This appendix provides feedback on Volume 4.

Feedback can include general comments from clients as well as answers to specific questions clients may have asked regarding one of the systems. This appendix will also address any discrepancies or ommissions, if any, made from the systems in the previous volume.

Please feel free to send us your comments and/or questions on the current volume, and we will include them in the next volume. Send your comments via e-mail to:

stadclub@omegaresearch.com

Thank you for your continued support!

ADX-DMI System (Page 14)

On page 14, under the commentary **{Exit Criteria}**, the second set of instructions reads:

```
If MarketPosition = -1 then Begin
        SellSetup = True;
        Exitshort next bar at ExitSPrc Stop;
End;
```

This is incorrect. The second line should be read SellSetup = False. The correct set of instructions,then, will read:

```
If MarketPosition = -1 then Begin
        SellSetup = False;
        Exitshort next bar at ExitSPrc Stop;
End;
```

The incorrect instructions are also listed on page 16. The corresponding explanation for the instructions is correct; the instructions themselves (and therefore the system) are incorrect. For the system to work correctly, you will have to edit the system itself in the PowerEditor.

ADX-DMI System (Page 15)

On page 15, the first sentence of the last paragraph reads:

"If the DMI- is lower than the DMI- and the ADX crosses over the DMI - and we are not in a short position already, we will set our entry price for a short position to the low of hte current bar minus 1 point (stored in the variable SellPrice)."

This is incorrect. The first sentence should instead read:

"If the **DMI+** is lower…"

The system itself is correct.

Momentum Retracement (Page 54)

The set of instructions described at the top of this page discusses the exit criteria. When we are in a long position OR when the other criteria is met, we reset our variables to false. Also, we want to exit a long position when the other criteria is met. The system is incorrectly written. It currently reads:

```
If MarketPosition = 1 or (PriceXMA < LowXMA OR MACDSigLine < 0) then Begin
        BuySwitch = False;
        Fase1 = False;
        Fast2 = False;
        ExitLong next bar at Market;
End;
```

Since we want to exit the long position only when the other critieria is met, we have to include an additional line. This segment of the system should read:

```
Condition1 = (PriceXMA < LowXMA OR MACDSigLine < 0);
If MarketPosition = 1 or Condition1 then Begin
        BuySwitch = False;
        Fase1 = False;
        Fast2 = False;
        If Condition1 thenExitLong next bar at Market;
End;
```

For the system to work correctly, you will have to edit the system in the PowerEditor.

INDEX