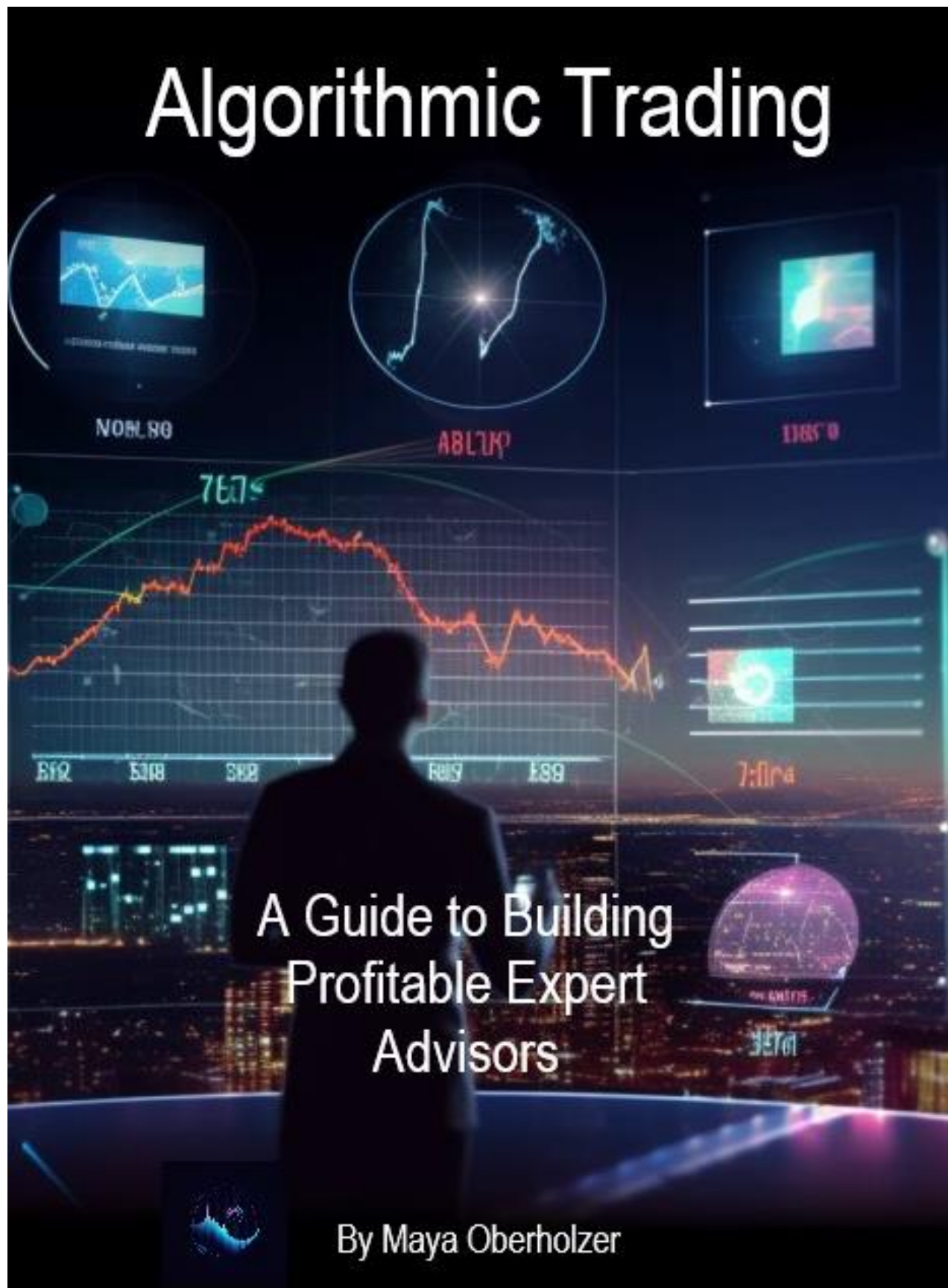




Algorithmic Trading

A Guide to Building Profitable Expert Advisors

By Maya Oberholzer



Algorithmic Trading

A Guide to Building Profitable Expert Advisors

By Maya Oberholzer

Table of Contents

Introduction	5
Understanding Algorithmic Trading	7
The Evolution of Trading	7
What is Algorithmic Trading?	9
The Role of Expert Advisors (EAs) and Indicators in Algorithmic Trading	11
Trading Strategies	13
Popular trading Strategies	16
Exploring MQL for Algorithmic Trading	22
Introduction to MQL	22
Creating Custom Indicators in MQL	23
Building Basic Expert Advisors in MQL	25
Converting Trading Strategies into Expert Advisors	29
Conclusion	31
About the Author	33
Free Downloads	35
Indicators:	35

Introduction

Welcome to the world of algorithmic trading, where the fusion of advanced technology and financial markets open a realm of exciting opportunities.

This e-book is a comprehensive guide that explores the intricacies of algorithmic trading, with a specific focus on the use of Expert Advisors (EAs) and rule-based automation in trading.

In recent years, algorithmic trading has revolutionized the way financial markets operate. Gone are the days of manual trading, where human limitations and emotions often influenced decision-making. Instead, traders now harness the power of algorithms and automation to execute trades swiftly and efficiently.

Within these digital realms, EAs play a pivotal role. They are computer programs that utilize predefined rules and algorithms to execute trades automatically. With the ability to analyze vast amounts of data, monitor market conditions, and respond instantaneously, EAs have become indispensable tools for traders seeking to capitalize on market opportunities.

In this e-book, we will delve into the core concepts of algorithmic trading, unravel the mysteries of EAs, and provide you with the knowledge to at least consider automating your manual trading systems.

We will begin by exploring the fundamentals of algorithmic trading, understanding how it has transformed the financial landscape and the advantages it offers over traditional trading methods.

We will then delve into the realm of EAs, exploring how to implement them using MQL, the language of choice for the popular MetaTrader platform. From creating custom indicators to developing complex trading strategies, you will gain insights into the process of converting your trading ideas into code.

Whether you are an experienced trader looking to automate your strategies or a novice seeking to enter the world of algorithmic trading, this e-book will serve as your comprehensive guide.

Embrace the power of algorithms, automation, and data-driven insights as you embark on your journey towards successful algorithmic trading. Let's dive in and unlock the endless possibilities that lie ahead!

Important links:

Website: <https://algotrade.co.za/>
Store: <https://store.algotrade.co.za/>
GitHub: <https://github.com/MayaOber>
YouTube: <https://www.youtube.com/@AlgoTradeMayaOber>
Facebook: <https://www.facebook.com/AlgoTradeMayaOber>
Instagram: <https://www.instagram.com/algotrademayaober>
LinkedIn: <https://www.linkedin.com/in/mayaoberholzer/>

Understanding Algorithmic Trading

The Evolution of Trading

Trading has undergone a remarkable evolution throughout history, transitioning from face-to-face open outcry trading to electronic trading platforms.

This evolution has been fueled by advancements in technology, particularly the development of computers and the internet. These technological advancements have revolutionized the way financial markets operate, leading to the rise of algorithmic trading.

It is important to understand the historical context and key milestones that have shaped algorithmic trading into what it is today.

The journey of algorithmic trading can be traced back to the 1970s when the first electronic trading platforms were introduced. These platforms allowed traders to submit orders electronically and have them executed automatically. Over time, trading algorithms became more sophisticated, incorporating complex mathematical models and technical indicators.

The 1980s saw the emergence of program trading, where large institutional investors used computer programs to execute trades based on predefined strategies. This laid the foundation for algorithmic trading as we know it

today. As technology continued to advance, algorithmic trading became more accessible to individual traders, leading to its widespread adoption.

With advancements in technology, trading has evolved into a highly automated and efficient process.

Algorithmic trading emerged as a prominent approach that leverages computer algorithms to execute trades in the financial markets.

Once a trading strategy has shown to be successful and profitable, it is a good candidate for an automated product such as an expert advisor that can trade automatically without any human intervention.

What is Algorithmic Trading?

Algorithmic trading involves the use of pre-defined instructions or algorithms to execute trades in an automated manner.

These algorithms are based on various factors such as market conditions, technical indicators, quantitative models, and a set of rules.

Algorithmic trading offers several advantages over traditional manual trading, including speed, accuracy, the ability to execute complex strategies, risk management, trading psychology and much more.

The key advantage of algorithmic trading is its ability to eliminate human emotion from the trading process. It allows for objective decision-making based on quantitative analysis rather than subjective judgment.

By removing the emotional element, algorithmic trading aims to optimize trading outcomes and allows for the implementation of sophisticated trading strategies. It minimizes human biases that can lead to poor trading decisions.

Once implemented, it is then possible to run various scenarios and optimizations. This tremendously increase profitability and lower the risks compared to manual trading.

Key Notes:

- Algorithmic trading combines computer programming and financial markets to execute trades at precise moments.
- Algorithmic trading attempts to strip emotions out of trades, ensures the most efficient execution of a trade, places orders instantaneously and may lower trading fees.
- Common trading strategies include trend-following strategies, arbitrage opportunities, and index fund rebalancing.
- Algorithmic trading is also executed based on trading volume (volume-weighted average price) or the passage of time (time-weighted average price).
- To get started with algorithmic trading, you must have computer access, network access, financial market knowledge, and coding capabilities.

The Role of Expert Advisors (EAs) and Indicators in Algorithmic Trading

Expert Advisors (EAs) and indicators play a crucial role in algorithmic trading, enabling traders to automate their strategies and make informed trading decisions.

EAs are computer programs that execute trades automatically based on predefined rules and algorithms, while indicators provide valuable insights into market trends and conditions. Together, they form the backbone of algorithmic trading systems, empowering traders with efficiency, accuracy, and consistency.

EAs act as intelligent assistants, tirelessly monitoring the markets, analyzing data, and executing trades without the need for human intervention.

By programming specific rules and conditions into an EA, traders can ensure that their trading strategies are executed with precision and discipline. EAs can perform complex calculations, manage multiple positions simultaneously, and react swiftly to market changes, which would be challenging for manual traders.

Indicators, on the other hand, provide traders with visual representations of market data, helping them identify patterns, trends, and potential trading opportunities. Indicators can be customized to suit specific trading strategies, allowing traders to monitor price movements, volatility, momentum, and other key market factors.

By interpreting the information provided by indicators, traders can make well-informed decisions, whether it be entering or exiting trades, adjusting risk parameters, or applying specific trading rules.

The combination of EAs and indicators enables traders to automate their trading strategies while benefiting from real-time market analysis. By leveraging the power of technology and data-driven insights, traders can eliminate emotional biases, reduce human errors, and optimize their trading performance. Additionally, EAs and indicators facilitate back testing and optimization, allowing traders to assess the historical performance of their strategies and fine-tune them for optimal results.

In conclusion, EAs and indicators are indispensable tools in algorithmic trading, providing traders with the ability to automate their strategies, analyze market data, and make informed trading decisions. Through their seamless integration into trading platforms, EAs and indicators empower traders to navigate the complexities of the financial markets with precision and confidence.

Trading Strategies



In this chapter, we delve into the fascinating world of trading strategies. Whether you are a novice or an experienced trader, understanding different trading strategies is crucial for success in financial markets. We will explore the fundamentals of trading strategies, their objectives, and the key factors that influence their effectiveness.

We can define trading strategies as systematic approaches that traders employ to make informed decisions about buying or selling financial instruments. These strategies are developed based on analysis, research, and a deep understanding of market dynamics.

Every trading strategy aims to achieve specific objectives. It could be maximizing profits, managing risk, generating consistent returns, or a combination of these goals. Visualize a trader strategizing to achieve their desired outcome amidst a sea of market fluctuations.

Trading strategies can be broadly classified into two categories: fundamental analysis and technical analysis. Fundamental analysis focuses on analyzing the intrinsic value of an asset, whereas technical analysis relies on historical price data and statistical indicators to predict future price movements.

Fundamental analysis strategies involve studying economic indicators, financial statements, industry trends, and market conditions to identify undervalued or overvalued assets. Investors who employ fundamental analysis seek to determine the true value of an asset and make trading decisions accordingly.

Technical analysis strategies rely on patterns, trends, and statistical indicators derived from historical price data. Traders using technical analysis aim to identify market trends, support and resistance levels, and entry and exit points for trades. Visualize a trader recognizing patterns on a price chart and executing trades based on technical signals.

Developing a trading strategy requires careful planning, research, and testing. Traders must consider their risk tolerance, time horizon, market conditions, and available resources. A well-defined trading plan includes

entry and exit rules, position sizing, risk management techniques, and performance evaluation.

Several factors can impact the effectiveness of a trading strategy. These include market conditions, volatility, liquidity, transaction costs, and psychological factors such as emotions and biases. Successful traders adapt their strategies to changing market dynamics and maintain discipline in executing their plan.

Popular trading Strategies

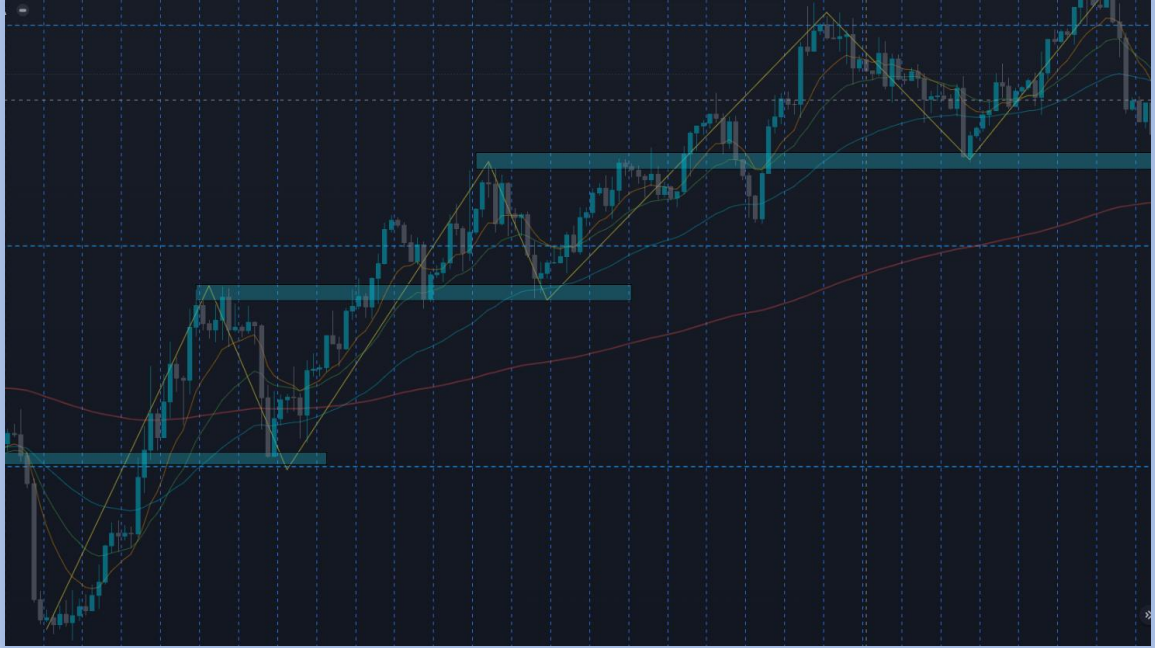
Trend Following or Momentum trading

Trend following is a strategy that aims to capture long-term price trends in financial markets. Traders using this strategy believe that the trend will continue and ride the wave until it shows signs of reversal.

Momentum trading strategy focuses on trading assets that have shown strong upward or downward momentum. Traders using this strategy believe that the momentum will continue and look for confirmation signals to enter trades in the direction of the prevailing trend.

The most common algorithmic trading strategies follow trends in moving averages, channel breakouts, price level movements, and related technical indicators. These are the easiest and simplest strategies to implement through algorithmic trading because these strategies do not involve making any predictions or price forecasts.

Trades are initiated based on the occurrence of desirable trends, which are easy and straightforward to implement through algorithms without getting into the complexity of predictive analysis. Using 50- and 200-day moving averages is a popular trend-following strategy.



Mean Reversion or Reversals

Mean reversion strategy is based on the concept that prices tend to revert to their average or mean value over time. Traders employing this strategy identify overbought or oversold conditions and take trades in the opposite direction, expecting prices to move back towards the mean.

Reversals normally happen after false breakouts and reverses at least back to the mean or a previous support or resistance level.

Ranging markets are most suitable for this strategy.



Breakouts

The breakout strategy involves identifying key levels of support or resistance and taking trades when the price breaks out of these levels. Traders using this strategy anticipate that the breakout will lead to a significant price movement in the direction of the breakout. This normally happens after the market have been ranging for a while and then breaks out to continue the trend after manipulation.



Scalping/Range Trading

Scalping is a short-term trading strategy that aims to profit from small price movements. Traders using this strategy enter and exit trades quickly, often within minutes or seconds, capturing small gains multiple times throughout the day. Ranges are a good example for this.



Swing Trading

Swing trading is a medium-term strategy that aims to capture short-to-medium-term price swings within an established trend. Traders using this strategy look for price retracements or pullbacks within the trend and enter trades when they anticipate the trend to resume.

Some more advanced strategies

There are some more advanced and refined strategies. These normally develop the more you study trading in depth and learn more from other experienced traders.

Some of these are:

- Intra day market movements at certain times of the day.
- Smart money concepts
- ICT Concepts and the models he teaches.
 - 2022 Models (OB's, FVG's and OTE, etc.)
 - Silver bullet trades and entries based on certain timings and programmed algo's
 - Many more...
- Pattern trading, Elliot Wave Theory and much much more...

Exploring MQL for Algorithmic Trading

Introduction to MQL

The MetaQuotes Language (MQL) is a programming language specifically designed for developing trading strategies and custom indicators for MetaTrader platforms.

There are two different versions of MQL, such as MQL4 for MT4 and MQL5 for MT5. There are similarities and differences.

It is important to cater for compatibility between these versions and platforms.

Our framework also provides the ability to develop once and being extendable for both versions. (So are all our current products and future developments)

Creating Custom Indicators in MQL

Custom indicators play a significant role in technical analysis and decision-making in algorithmic trading. A combination of standard and custom indicators can be used to better decision making and implementation for our rule-based algorithms as expert advisors. See example code and of the structure of such:

```
//+-----+
//|                                     MarketViewer.mq5 |
//| Market Viewer gives a quick glance on correlation and direction. |
//|                                     Copyright 2023, AlgoTrade |
//|                                     https://www.algotrade.co.za |
//+-----+
#property copyright "Copyright 2023, AlgoTrade,"
#property link      "https://www.algotrade.co.za"
#property version   "1.00"
#property indicator_chart_window

#property icon "MyIcon.ico"
#property strict

/* ...

//Include for MQL4/MQL5 compatibility
#include "MQL4.mqh"

input string symbols = "----- Symbols list -----"; //Symbols list
input string Symbol1 = "GBPUSD";
input string Symbol2 = "EURUSD";
input string Symbol3 = "AUDUSD";
input string Symbol4 = "NZDUSD";
input string Symbol5 = "XAUUSD";
input string Symbol6 = "";
input string Symbol7 = "";
input string Symbol8 = "";

input ENUM_BASE_CORNER corner = CORNER_RIGHT_UPPER; //Upper right

int dateTimeMin = 0;

string listPairs[]; // = { "GBPUSD","EURUSD","AUDUSD","NZDUSD","XAUUSD"}; //,"USDJPY","USDCHE","USDCAD","US30","US500","USTEC"};

color headerColor = clrWhite;
color buyColor = clrLimeGreen;
color sellColor = clrRed;

string arrowUp = CharToStr(233);
string arrowDown = CharToStr(234);

int OnInit() ...

void OnDeinit(const int reason) ...

int OnCalculate(const int rates_total, const int prev_calculated, const datetime &time[], const double &open[], const double &high[],
               const double &low[], const double &close[], const long &tick_volume[], const long &volume[], const int &spread[]) ...

bool BullCandle(string symbol, ENUM_TIMEFRAMES TF) ...

bool BearCandle(string symbol, ENUM_TIMEFRAMES TF) ...

void OnChartEvent(const int id, const long &lparam, const double &dparam, const string &sparam) ...
```

The result of such an indicator:



Note: See our free download section to get this free indicator and many more...

Building Basic Expert Advisors in MQL

The implementation of Expert Advisors (EAs) is a critical step in harnessing the power of algorithmic trading. EAs are computer programs that execute trades automatically based on predefined rules and algorithms, allowing traders to automate their trading strategies and take advantage of market opportunities. The successful implementation of EAs requires careful planning, consideration of various factors, and a thorough understanding of the underlying trading strategy.

To implement an EA effectively, traders must first define the rules and conditions that govern their trading strategy. This involves specifying entry and exit criteria, position sizing, risk management parameters, and any other relevant rules. Clear and unambiguous rules ensure that the EA can execute trades consistently and accurately.

Next, traders need to program the defined rules into the EA using a suitable programming language, such as MQL for MetaTrader platforms. This involves writing code that reflects the desired trading logic, including conditions for trade execution, stop-loss and take-profit levels, and any additional functions or calculations required.

Once the EA is programmed, it needs to be tested and optimized. This involves backtesting the EA using historical market data to assess its performance and fine-tune its parameters. Traders can analyze performance metrics such as profitability, drawdown, and risk-adjusted returns to evaluate the effectiveness of the EA. Optimization techniques, such as adjusting parameters or using genetic algorithms, can be applied to enhance the EA's performance.

After testing and optimizing the EA, traders can deploy it in live trading conditions. It is crucial to monitor the EA's performance in real-time, ensuring that it executes trades as expected and adheres to the predefined rules. Ongoing monitoring allows traders to identify any issues or necessary adjustments and maintain the optimal performance of the EA.

In summary, the implementation of EAs involves defining the trading strategy, programming the rules into the EA, testing and optimizing its performance, and monitoring its execution in live trading. By following a systematic approach and leveraging technology, traders can harness the power of EAs to automate their trading strategies and improve their trading efficiency and consistency.

Example code and layout of such:

```
//+-----+
//|                                     Extreme.mq4 |
//| Market Viewer gives a quick glance on correlation and direction. |
//|                                     Copyright 2023, AlgoTrade |
//|                                     https://www.algotrade.co.za |
//+-----+
#property copyright "Copyright 2023, AlgoTrade."
#property link      "https://www.algotrade.co.za"
#property version   "7.00"
#property icon "MyIcon.ico" //This Icon should be in the same folder as expert at compile time to be included
#property strict

input int MABase = 200; //Base MA - 200 Default
input int MABaseTP = 189; //Base MA TP - 189 Default
input double MABasedeviation = 2.2; //Std Dev - 2.2 Default
input double BBgrace = 10; //10 BB grace - 10 Default
input double MAgance = 7; //7 MA grace = 7 Default

extern int NoOfPivots = 10; //Number of pivots to show - 10 Default
extern double GridSize = 5; //GridSize 5 Default
extern bool showGrid = true; // Show grid

extern bool showWaves = true; // Show waves
extern color WaveColor = clrDarkGray; //Color for waves

extern bool lotSizeFixed = false; //Use Fixed lot Size
extern double lotSize = 0.01; //Fixed lot size
extern double riskFactor = 1; //Risk Factor for non fixed lot size
extern int magicNumber = 55555; //Magic number
extern bool alertsActive = true; //Alerts Active
extern double distanceMeasure = 172; //Distance rule to 172SMA

//Internal properties

string AppName = "ExtremeV7";
string AppVersion = "7.00";

int candleHist = 2500;
double HH = 0;
double LL = 0;
datetime HHT;
datetime LLT;

int WaveIndex = 0;
double WavePrices[];
datetime WaveTimes[];

bool BuyMode = false;
bool SellMode = false;

bool didBuy = false;
bool didSell = false;

int dateTimeMin = 0;

bool initDone = false;
```

```

bool initDone = false;

int OnInit() ...

void deinit() ...

void RemoveExpert() ...

void OnTick() ...

int State_WaveBand = 0;
void CheckWaves(int candles) ...

void UpdateGUIOutputs() ...

double lastBoughtPrice = 0;
double lastSoldPrice = 0;
void Trade() ...

int orderCount = 0;
int orderIndex = 0;
int orderRetries = 5;
extern int slippage = 50;
void Buy(double sl, double tp, string comment, int magicNumber) ...

void Sell(double sl, double tp, string comment, int magicNumber) ...

// *****
// Helper Methods
// *****

int noOfBarsOnChart = 0;
datetime LastActionTime_TF = dateTimeMin;
datetime LastActionTime_sTF = dateTimeMin;
bool isNewCandle(int tf) ...

double PipSize(string symbol) ...

double PipsToPrice(double pips, string symbol) ...

double PriceToPips(double price, string symbol) ...

void ResizeArrays() ...

int maxPivots = 15;
double pivotsHighs[15];
datetime pivotsHighTimes[15];
void SetPivotHighs() ...

double pivotsLows[15];
datetime pivotsLowTimes[15];
void SetPivotLows()
{

```

Converting Trading Strategies into Expert Advisors

Converting trading strategies into Expert Advisors (EAs) is a transformative process that allows traders to automate their strategies and execute trades seamlessly. By translating trading rules and logic into code, traders can harness the power of technology to execute trades with precision and efficiency.

The first step in converting a trading strategy into an EA is to clearly define the rules and conditions that govern the strategy. This involves specifying the entry and exit criteria, position sizing, risk management parameters, and any other relevant rules. Clarity and specificity are key to ensuring accurate translation into code.

Next, traders need to program the defined rules into the EA using a suitable programming language, such as MQL for MetaTrader platforms. This involves writing code that reflects the trading logic, including the conditions for trade execution, stop-loss and take-profit levels, and any additional functions or calculations required.

During the conversion process, traders should ensure that the code is robust and error-free. Careful consideration should be given to handling exceptional scenarios and incorporating appropriate error handling mechanisms.

Once the code is written, it needs to be tested thoroughly. Backtesting the EA using historical market data is crucial to assess its performance and identify any potential issues or areas for improvement. Traders can analyze various performance metrics, such as profitability, drawdown, and risk-adjusted returns, to evaluate the effectiveness of the EA.

Additionally, optimization techniques can be applied to fine-tune the EA's parameters and maximize its performance. By adjusting variables, testing different parameter combinations, and employing optimization algorithms, traders can optimize the EA for improved results.

Converting trading strategies into EAs empowers traders to automate their strategies, reduce emotional biases, and execute trades consistently. By following a systematic approach and ensuring the accuracy and robustness of the code, traders can unlock the full potential of their trading strategies and enhance their trading experience.

Conclusion

We have explored the fascinating world of algorithmic trading and the role of Expert Advisors (EAs) in automating trading strategies.

We started by understanding the fundamentals of algorithmic trading and its advantages over traditional manual trading approaches. We delved into the programming skills required to develop EAs using MQL, the language of choice for popular trading platforms like MetaTrader.

We explored the implementation of EAs, emphasizing the importance of defining clear trading rules, programming the rules into the EA, and testing and optimizing its performance.

We highlighted the power of algorithmic trading and automation in enhancing trading efficiency, accuracy, and consistency. EAs, coupled with indicators and advanced technologies like Artificial Intelligence (AI), enable traders to make data-driven decisions, monitor market conditions, and execute trades swiftly.

By combining our years of software development experience with years of trading expertise, we have acquired a unique skill set to write EAs for MT4 and MT5.

This e-book, with its multiple chapters and a brief exploration of algorithmic trading, serves as a valuable resource for both experienced traders and newcomers looking to venture into the world of automated trading.

As you embark on your algorithmic trading journey, remember to continuously learn, adapt, and stay informed about the latest trends and innovations. Algorithmic trading offers tremendous potential, but it also requires discipline, rigorous testing, and sound risk management practices.

May this knowledge empower you to unlock the endless possibilities of algorithmic trading and guide you towards success in the dynamic world of financial markets. Good luck!

About the Author

The author of this e-book, Maya Oberholzer is a seasoned Senior Software Analyst Programmer with over 20 years of experience in the IT industry and technologies. With a strong background in software solutions architecture, they have demonstrated expertise in delivering robust and scalable software solutions.

From 2003 to 2006, the author worked as a contractor in the UK, providing their services to renowned corporate companies and the UK government. Their entrepreneurial spirit, leadership abilities, effective communication skills, endurance, and unwavering commitment have been instrumental in their success throughout their career.

The author possesses a diverse skill set, including expertise in software architecture and design, system integration, and agile development methodologies such as Agile, Scrum, and Waterfall. They are well-versed in emerging technologies such as .NET Core, Azure, AI, Machine Learning, Deep Learning, Angular, Python, and more. Their proficiency extends to various programming languages, including C#, VB.NET, ASP.NET, SQL Server, and web technologies like HTML/XML/XSLT.

In addition to their extensive software development experience, the author has a deep understanding of the financial markets and trading. They have acquired expertise in MQL4/MQL5 programming, particularly in developing Expert Advisors (EAs) for the popular MT4 platform.

Throughout their career, the author has demonstrated a commitment to delivering high-quality software solutions, following best practices and industry standards. They are skilled in utilizing software modeling and specification tools like UML, ERD, and DFD, and have worked with clients across various sectors, covering the entire development life cycle.

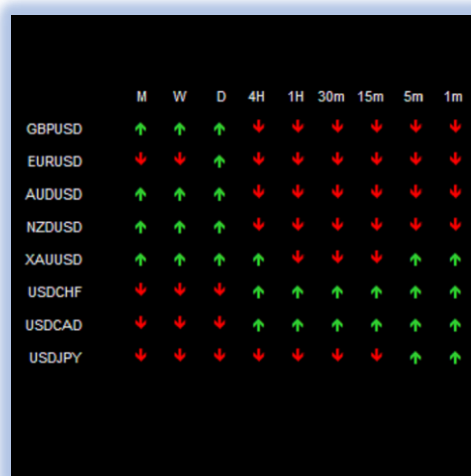
With a passion for staying up-to-date with the latest technologies and trends in the IT industry, the author brings a wealth of knowledge and experience to the world of algorithmic trading and EA development.

Free Downloads

Free downloads: [GitHub](#)

Indicators:

Market Viewer



	M	W	D	4H	1H	30m	15m	5m	1m
GBPUSD	↑	↑	↑	↓	↓	↓	↓	↓	↓
EURUSD	↓	↓	↑	↓	↓	↓	↓	↓	↓
AUDUSD	↑	↑	↑	↓	↓	↓	↓	↓	↓
NZDUSD	↑	↑	↑	↓	↓	↓	↓	↓	↓
XAUUSD	↑	↑	↑	↑	↓	↓	↓	↑	↑
USDCHF	↓	↓	↓	↑	↑	↑	↑	↑	↑
USDCAD	↓	↓	↓	↑	↑	↑	↑	↑	↑
USDJPY	↓	↓	↓	↓	↓	↓	↓	↑	↑

Market Viewer gives a quick glance on correlation and direction.

This indicator takes a list of symbols (max 8). Leave a symbol setting empty if you wish not to use that many - like in the defaults...

It will check for each timeframe (Monthly, Weekly, Daily, H4, H1, 30min, 15min, 5min and 1min) if the current candle on that timeframe is bullish or bearish.

[Download Market Viewer for MT5](#) / [Download Market Viewer for MT4](#)