

Міністерство освіти і науки України
Державний ВНЗ «Національний гірничий університет»

Факультет інформаційних технологій
(факультет)

Кафедра програмного забезпечення комп'ютерних систем
(повна назва)

ПОЯСНЮВАЛЬНА ЗАПИСКА
дипломної роботи

магістра
(назва освітньо-кваліфікаційного рівня)

галузь знань *12 Інформаційні технології*
(шифр і назва галузі знань)

спеціальність *121 Інженерія програмного забезпечення*
(код і назва спеціальності)

спеціалізація *Програмне забезпечення систем*
(код і назва спеціалізації)

освітній рівень *Магістр*
(назва освітнього рівня)

кваліфікація *інженер-програміст*
(назва кваліфікації)

на тему: *Удосконалення методики побудови автоматизованої торгової системи з використанням патернів «Price Action»*

Виконавець:

студент 2 курсу, групи 121М-16-1

Клімов Д. Г.
(підпис) (прізвище та ініціали)

Керівники	Посада, прізвище, ініціали	Оцінка	Підпис
проекту	<i>д.т.н., проф. Мещеряков Л. І.</i>		
розділів:			
Спеціальний	<i>д.т.н., проф. Мещеряков Л. І.</i>		
Економічний	<i>к.е.н., доц. Касьяненко Л. В.</i>		

Рецензент			
-----------	--	--	--

Нормоконтроль	<i>к.т.н., доц. Коротенко Л.М.</i>		
---------------	------------------------------------	--	--

Дніпропетровськ
2018

Практична цінність результатів полягає у: розробленні автоматизованої торгової системи, орієнтованої на широку аудиторію, яка буде здатна вести прибуткову торгівельну діяльність.

4 ВИМОГИ ДО РЕЗУЛЬТАТІВ ВИКОНАННЯ РОБОТИ

Автоматизована торгова система повинна тестуватися на історичних даних коливань торгових активів, а також в умовах поточного ціноутворення

5 ЕТАПИ ВИКОНАННЯ РОБІТ

Найменування етапів робіт	Строки виконання робіт (початок – кінець)
Формування та формалізація завдань магістерської роботи	01.06.2017 15.07.2017
Аналіз сфери застосування торгових систем та алгоритмів пошуку торгових сигналів	16.07.2017 31.07.2017
Проектування та розробка програмного продукту, що вирішує поставлені в дипломній роботі задачі	01.07.2017 10.09.2017
Тестування розробленої автоматизованої торгової системи на історичних та поточних даних і аналіз отриманих результатів	11.09.2017 14.01.2018

6 РЕАЛІЗАЦІЯ РЕЗУЛЬТАТІВ ТА ЕФЕКТИВНІСТЬ

Економічний ефект від реалізації результатів роботи очікується позитивним завдяки використанню комбінованого підходу до аналізу ринкової динаміки та виявленні торгових сигналів з високою вірогідністю виконання, що дає можливість отримувати прибуток від торгівельної діяльності.

Соціальний ефект від реалізації результатів роботи очікується позитивним завдяки створенню програмного продукту, що задовольняє потреби широкого можливих споживачів

7 ДОДАТКОВІ ВИМОГИ

Завдання видав

_____ (підпис)

Мещеряков Л. І.

_____ (прізвище, ініціали)

Завдання прийняв до виконання

_____ (підпис)

Клімов Д. Г.

_____ (прізвище, ініціали)

Дата видачі завдання: _____ .2017 р.

Термін подання дипломного проекту до ДЕК 19.01.2018

Реферат

Пояснительная записка: 202 с., 87 рис., 4 прил., 59 источников.

Объект исследования: технология построения автоматизированной системы для торговли на международных мировых рынках.

Цель магистерской работы: совершенствование методики построения автоматизированной торговой системы с использованием паттернов Price Action.

Методы исследования. При решении поставленной задачи использовались научные достижения в сфере технического и фундаментального анализа рыночной динамики и в области разработки программного обеспечения.

Научная новизна результатов, ожидается, заключается в проведении анализа и выявлении недостатков в сфере существующих автоматизированных торговых систем, ориентированных на широкого потребителя, и совершенствование методики их построения с помощью использования различных видов анализа, в частности, паттернов Price Action.

Практическая ценность исследования заключается в разработке автоматизированной торговой системы, ориентированной на широкого потребителя, учитывая особенности сферы ритейлерской торговли, которая будет способна вести прибыльную торговую деятельность.

Область применения. Разработанный программный продукт может использоваться в сфере анализа динамики мировых финансовых рынков с целью получения прибыли.

Значение работы и выводы. Усовершенствованная методика позволяет проектировать торговые системы, которые имеют высокую точность прогнозирования рыночных движений, благодаря использованию комбинации различных видов технического анализа, подтверждается разработанным программным продуктом в данной магистерской работе.

Прогнозы по развитию исследований. Разработать универсальные программные модули, каждый из которых может использоваться для отдельного компонента анализа рыночной динамики. Разработать систему управления капиталом и риском для определения возможностей проведения торговой деятельности.

В разделе «Экономика» проведены расчеты трудоемкости разработки программного обеспечения, затрат на создание ПО и длительности его разработки, а также проведены маркетинговые исследования рынка сбыта созданного программного продукта.

Список ключевых слов: FOREX, PRICE ACTION, ТЕХНИЧЕСКИЙ АНАЛИЗ, ТЕОРИЯ ДОУ, ТОРГОВЫЙ РОБОТ Трейдер маркетмейкеров, ликвидность, Ритейловую ИЛИ РОЗНИЧНАЯ FOREX.

Реферат

Пояснювальна записка: 202 с., 87 рис., 4 додатка., 59 джерел.

Об'єкт дослідження: технологія побудови автоматизованої системи для торгівлі на міжнародних світових ринках.

Мета магістерської роботи: вдосконалення методики побудови автоматизованої торгової системи з використанням патернів Price Action.

Методи дослідження. При рішенні поставленої задачі використовувалися наукові досягнення в сфері технічного і фундаментального аналізу ринкової динаміки та в області розробки програмного забезпечення.

Наукова новизна результатів, що очікуються, полягає у проведенні аналізу та виявленні недоліків у сфері існуючих автоматизованих торгових систем, орієнтованих на широкого споживача, і вдосконалення методики їх побудови за допомогою використання різноманітних видів аналізу, зокрема, патернів Price Action.

Практична цінність результатів полягає у розробленні автоматизованої торгової системи, орієнтованої на широкого споживача, враховуючи особливості сфери ритейлової торгівлі, яка буде здатна вести прибуткову торгівлю діяльність.

Область застосування. Розроблений програмний продукт може використовуватися в сфері аналізу динаміки світових фінансових ринків з метою отримання прибутку.

Значення роботи та висновки. Удосконалена методика дозволяє проектувати торгові системи, які мають високу точність прогнозування ринкових рухів, завдяки використанню комбінації різних видів технічного аналізу, що підтверджується розробленим програмним продуктом в даній магістерській роботі.

Прогнози щодо розвитку досліджень. Розробити універсальні програмні модулі, кожен з яких може використовуватися для окремого компоненту аналізу ринкової динаміки. Розробити систему управління капіталом та ризиком для визначення можливостей проведення торгової діяльності.

У розділі «Економіка» проведені розрахунки трудомісткості розробки програмного забезпечення, витрат на створення ПЗ й тривалості його розробки, а також проведені маркетингові дослідження ринку збуту створеного програмного продукту.

Список ключових слів: FOREX, PRICE ACTION, ТЕХНІЧНИЙ АНАЛІЗ, ТЕОРІЯ ДОУ, ТОРГОВИЙ РОБОТ, ТРЕЙДЕР, МАРКЕТМЕЙКЕР, ЛІКВІДНІСТЬ, РИТЕЙЛОВИЙ АБО РОЗДРІБНИЙ FOREX.

The abstract

Explanatory note: 202 p., 87 fig., 4 applications, 59 sources.

Object of research: the technology of building an automated system for trade in international world markets.

The purpose of the degree project: improving the methodology for building an automated trading system using the Price Action Patterns.

Methods of research. In solving the problem, scientific advances in the field of technical and fundamental analysis of market dynamics and in the field of software development were used.

The scientific novelty of the expected results is to analyze and identify shortcomings in the field of existing automated trading systems, aimed at the broad consumer, and to improve the methodology for constructing them using a variety of types of analysis, in particular, Price Action.

The practical value of work is the development of an automated trading system oriented to the broad consumer, taking into account the features of the retail trade, which will be able to conduct profitable trading activities

The scope. The developed software product can be used in the analysis of the dynamics of world financial markets in order to profit.

The value of the work and conclusions. The advanced technique allows designing trading systems that have high accuracy of forecasting of market movements, due to the use of a combination of different types of technical analysis, which is confirmed by the developed software product in the master's work.

Projections on development research. Develop universal program modules, each of which can be used for a separate component of the analysis of market dynamics. Develop a capital and risk management system to determine trading activity opportunities.

In section "Economics" calculated the complexity of software development, the cost of creating the software and the duration of its development, and marketing studies market created by the software.

List of keywords: FOREX, PRICE ACTION, TECHNICAL ANALYSIS, DOW THEORY, TRADING ROBOT, TRADER, MARKETMAKER, LIQUIDITY, RETAIL FOREX.

Зміст

	Перелік скорочень	10
	Вступ	11
1	РОЗДІЛ 1. АНАЛІЗ РИНКУ FOREX, ЙОГО ФУНКЦІЇ В СИСТЕМІ СВІТОВИХ ЕКОНОМІЧНИХ ВІДНОСИН	14
1.1	Поняття світового фінансового ринку	14
1.2	FOREX	15
1.2.1	Визначення ринку FOREX	15
1.2.2	Причини появи FOREX в сучасному вигляді	15
1.2.3	Зростання обороту на валютних ринках	17
1.2.4	Ритейловий FOREX	18
1.2.5	Учасники торгів на валютному ринку	19
1.2.6	Маркетмейкери, їх роль в системі ринкового ціноутворення	21
1.3	Використання автоматизованих торгових систем	22
1.3.1	Алгоритмічна торгівля	24
1.3.2	Вплив алгоритмічних систем на ліквідність	25
1.3.3	Види спекулятивної торгівлі	26
1.3.4	Способи підключення автоматизованих систем до ринкових торгів	27
1.3.5	Торгові роботи на ритейловому FOREX	30
1.3.6	Особливості автоматизованої торгової системи з використанням патернів Price Action	31
	Висновки	32
2	РОЗДІЛ 2. МЕТОДИ АНАЛІЗУ І ПРОГНОЗУВАННЯ РУХУ ЦІНИ НА СВІТОВИХ РИНКАХ З МЕТОЮ ОТРИМАННЯ ПРИБУТКУ	33
2.1	Існуючі методи аналізу торгових інструментів	33
2.2	Фундаментальний аналіз	33
2.3	Технічний аналіз	39
2.3.1	Поняття технічного аналізу	39
2.3.2	Принципи технічного аналізу	41
2.4	Методи та інструменти технічного аналізу	42
2.4.1	Класичний аналіз	42
2.4.2	Графічний аналіз	47
2.4.2.1	Рівні підтримки та опору	48
2.4.2.2	Механіка попиту та пропозиції	49
2.4.2.3	Рівень підтримки	52
2.4.2.4	Рівень опору	54

2.4.2.5	Торгові інтерпретації	54
2.4.2.6	Динамічні рівні підтримки та опору	60
2.4.3	Індикаторний аналіз	65
2.4.3.1	Канальні індикатори	66
2.4.3.2	Осцилятори	67
2.4.3.3	MACD	68
2.4.3.4	Stochastic	72
2.4.3.5	RSI	73
2.4.3.6	Хвилі Боллинджера	81
2.5	Опціонний аналіз	86
2.5.1	Опис ринку ф'ючерсів	87
2.5.2	Сутність опціонного аналізу	87
2.6	Патерни Price Action	92
2.6.1	Школа Price Action. Поняття патернів та моделей	92
2.6.2	Аналіз ефективності патернів Price Action	95
	Висновки...	106
3	РОЗДІЛ 3. РЕАЛІЗАЦІЯ АЛГОРИТМУ І ЗАСОБІВ УПРАВЛІННЯ РАХУНКОМ АВТОМАТИЗОВАНОЇ ТОРГОВОЇ СИСТЕМИ	107
3.1	Засоби розробки	107
3.1.1	Опис компонентів торгової платформи MetaTrader 4	107
3.1.2	Опис інтерфейсу клієнтського терміналу MT4	110
3.1.3	Опис MT4 Editor	112
3.1.4	Опис програмних можливостей мови MQL 4	115
3.2	Загальні відомості про управління торговим рахунком	118
3.2.1	Поняття балансу та вільних коштів	119
3.2.2	Визначення термінів «Лот» і «обсяг»	119
3.2.3	Поняття маржи, вільної маржи та «рівень»	121
3.2.4	Margin Call та Stop Out	122
3.2.5	Ордер, типи виконання ордерів	123
3.2.6	Валютний SWAP	125
3.3	Методика пошуку точок входу в ринок	126
3.3.1	Періодичність аналізу	127
3.3.2	Торговий сигнал, як основний елемент аналізу	128
3.3.3	Сила точки входу	128
3.4	Структура і компоненти торгової системи	131
3.4.1	Реалізація компоненту аналізу торгового інструменту	135
3.4.2	Реалізація компоненту управління рахунком	140
3.5	Аналіз результатів роботи автоматизованої торгової системи	141
	Висновки	147

4	РОЗДІЛ 4. ЕКОНОМІЧНА ЧАСТИНА	148
4.1	Визначення трудомісткості розробки програмного забезпечення	148
4.2	Витрати на створення програмного забезпечення	150
4.3	Тривалість розробки програмного забезпечення	151
4.4	Маркетингові дослідження ринку збуту розробленого програмного продукту	151
4.5	Оцінка економічної ефективності впровадження програмного забезпечення	155
	ВИСНОВКИ	157
	СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	159
	Додаток А. Текст програми	164
	Додаток Б. Відгук на дипломну роботу магістра	186
	Додаток В. Рецензія на дипломну роботу магістра	187
	Додаток Г. Результати аналізу ефективності патернів Price Action на таймфреймах H1 та H4	188

Перелік скорочень

FOREX –	Foreign Exchange (ринок міжбанківського обміну валюти)
ДЦ –	дилінговий центр
МВФ –	Міжнародний валютний фонд
CFD –	contracts for difference (контракти на різницю)
СНД –	Співдружність незалежних держав
ECN –	Electronic Communication Network (електронна торгівельна мережа)
ММ –	Маркетмейкер, «спеціаліст», «фахівець»
MT4 –	MetaTrader 4
MQL4 –	MetaQuotes Language 4
MA –	Moving Average (ковзне середнє)

Вступ

Актуальність роботи. У наш час благополуччя населення майже будь-якої країни на планеті визначається великою кількістю різних чинників, таких як геополітична ситуація в країні і в світі, внутрішня політика уряду щодо економіки і підтримання її в здоровому стані, соціальна стабільність та інше. Ці та багато інших факторів впливають на ще один показник кожної окремо взятої держави – інвестиційна привабливість. Саме великі фінансові компанії і інвестиційні інститути відіграють велику роль у розвитку науки і економіки різних країн, адже більшість людей не мають засобів для реалізації тих чи інших ідей. І лише за допомогою зовнішнього, приватного фінансування втілюються ідеї та проекти, які змінюють життя на краще.

Інвестори перебувають в постійному пошуку нових варіантів вкладення коштів і оскільки ситуація постійно змінюється (одні компанії замінюють інші, відбуваються різні катаклізми, як людського характеру, так і незалежні від нас), рухається і міжнародний капітал. А капітал, в свою чергу, виражається в грошах. Звідси з'являється потреба в новому вигляді фінансових ринків – валютний або більш відомий як FOREX, на яких працюють трейдери.

Завдання трейдера просте – купити дешево і продати дорого. Для прогнозування руху котирувань валютних пар використовуються сотні різних видів аналізу, однак у всьому цьому є один нюанс – людський фактор. Люди піддаються емоціям і в їх пориві можуть приймати неправильні рішення, які призводять до втрати депозиту або навіть капіталу. Саме тому постійно створюються, розробляються й удосконалюються різні програмні рішення, які виключають людський фактор – торгові роботи.

Мета та завдання дослідження. Метою даної магістерської роботи є створення вдосконаленої методики розробки автоматизованих торгових систем для здійснення торгівельної діяльності на світових ринках.

Для досягнення поставленої мети в роботі сформульовані і вирішені такі завдання:

1. Проведення аналізу та виявлення недоліків існуючих торгових систем, орієнтованих на широкого споживача.
2. Створення методики побудови автоматизованої торгової системи з використанням різноманітних видів аналізу, зокрема, патернів Price Action.

Об'єкт дослідження – технологія побудови автоматизованих торгових систем.

Предмет дослідження – методика розробки торгових систем з використанням комбінованого аналізу.

Ідея роботи полягає в удосконаленні методики розробки автоматизованих торгових систем за рахунок використання різноманітних видів аналізу для досягнення високої точності прогнозування ринкової динаміки.

Методи дослідження. При вирішенні поставленого завдання використовувалися наукові досягнення в сфері технічного і фундаментального аналізу ринкової динаміки та в області розробки програмного забезпечення

Наукові положення, очікувані наукові результати.

1. Сформований аналіз недоліків існуючих торгових систем.
2. Створення автоматизованої торгової системи з використанням різноманітних видів аналізу, зокрема, патернів Price Action.

Обґрунтованість і достовірність наукових положень.

Обґрунтованість і достовірність наукових положень, висновків і рекомендацій магістерської роботи обґрунтована коректністю поставлених проблем і прийнятих припущень при математичному описі процесів, обґрунтованістю вихідних посилок, достатнім обсягом вибірки даних і верифікованими на модельних об'єктах результатами обчислень.

Наукова новизна отриманих результатів полягає в створеній методиці побудови автоматизованих торгових систем з використанням різних видів аналізу.

Практичне значення отриманих результатів полягає в розробці програмних модулів, програмного продукту, що дозволяють оцінити переваги

розробки алгоритмізації різних видів аналізу, використання їх у комбінації з патернами Price Action.

Зв'язок роботи з державними програмами, планами науково-дослідних робіт.

Результати дипломної роботи можуть бути використані підприємствами, фірмами, розробниками або приватними особами для проектування торгових систем з метою отримання прибутку під час торгівельної діяльності на фінансових ринках.

Особливий внесок магістра складається в:

- використанні практичного досвіду торгової діяльності при алгоритмізації різних видів аналізу ринкової динаміки;
- виборі методів досліджень і технологій реалізації;
- створенні автоматизованої торгової системи, яка реалізує комбінований підхід до аналізу та прогнозування ринкового ціноутворення;
- розробці теоретичної частини роботи, в якій досліджені і систематизовані знання про існуючі види технічного та фундаментального аналізу, механізми роботи на фінансових ринках;
- оцінці отриманих результатів.

Апробація результатів магістерської роботи.

Основні положення і результати були докладені та обговорені на студентській науковій конференції.

Структура і обсяг роботи. Робота складається з вступу, чотирьох розділів і висновків. Містить 202 сторінок друкованого тексту, в тому числі 134 сторінки тексту основної частини з 87 рисунками, списку використаних джерел з 30 найменуваннями на 3 сторінках, 4 додатків на 38 сторінках.

РОЗДІЛ 1. АНАЛІЗ РИНКУ FOREX, ЙОГО ФУНКЦІЇ В СИСТЕМІ СВІТОВИХ ЕКОНОМІЧНИХ ВІДНОСИН

1. 1. Поняття світового фінансового ринку

У процесі глобалізації та установки тісних економічних відносин між різними країнами, в першу чергу між США і Європою, появи міжнародних компаній і корпорацій, виникає і світовий фінансовий ринок, як величезна торгова площадка для продавців і покупців.

Фінансовий ринок – являє собою систему відносин, що виникає в процесі обміну економічних благ з використанням грошей в якості активу-посередника.

На фінансовому ринку відбувається мобілізація капіталу, надання кредиту, здійснення обмінних грошових операцій і розміщення фінансових коштів у виробництві. А сукупність попиту і пропозиції на капітал кредиторів і позичальників різних країн утворює світовий фінансовий ринок.

Світовий фінансовий ринок – це частина світового ринку позичкового капіталу, сукупність попиту і пропозиції на капітал кредиторів і позичальників різних країн. Одним із сегментів світового фінансового ринку є фондовий ринок або ринок цінних паперів.

Світовий фінансовий ринок почав розвиватися з початком вивезення (міграції) капіталу з кінця XIX століття.

Фінансові ринки поділяються на:

- Фондовий ринок;
- Терміновий ринок;
- Грошовий ринок;
- Ринок капіталів;
- Валютний ринок (FOREX і спотовий).

Валютний ринок – це система стійких економічних та організаційних відносин, що виникають під час здійснення операцій з купівлі або продажу іноземної валюти, платіжних документів в іноземних валютах, а також операцій по руху капіталу іноземних інвесторів.

На валютному ринку відбувається узгодження інтересів інвесторів, продавців і покупців валютних цінностей. Західні економісти характеризують валютний ринок з організаційно-технічної точки зору як сукупну мережу сучасних засобів зв'язку, що з'єднують національні та іноземні банки і брокерські фірми.

1. 2. FOREX

1. 2. 1. Визначення ринку FOREX

FOREX – це ринок міжбанківського обміну валюти за вільними цінами (котирування формуються без обмежень або фіксованих значень). В англomовному середовищі слово «FOREX» може означати не тільки взаємний обмін валюти, але і всю сукупність валютних операцій, а також іноземні валюти як такі.

У СНД просторі термін FOREX зазвичай використовують в більш вузькому сенсі – мається на увазі виключно спекулятивна торгівля валютою через комерційні банки або дилінгові центри (далі – ДЦ), яка ведеться з використанням кредитного плеча, тобто маржинальна торгівля валютою.

Операції на ринку FOREX за програмними цілями можуть бути торговими, спекулятивними, хеджуючими та регулюючими (валютні інтервенції центробанків).

1. 2. 2. Причини появи FOREX в сучасному вигляді

15 серпня 1971 року, президент США Річард Ніксон оголосив рішення про скасування вільної конвертованості долара в золото (відмовився від золотого стандарту), відмовившись таким чином в односторонньому порядку від виконання Бреттон-Вудських угод (згідно з якими долар забезпечувався золотом, а всі інші валюти долларом). У грудні 1971 року у Вашингтоні було досягнуто Смітсонівської угоди, за якою замість 1% коливань курсу валюти щодо долара США стали допускатися коливання в 4,5% (на 9% для недоларових валютних пар). Це зруйнувало систему стійких валютних курсів і

стало кульмінаційною подією в кризі післявоєнної Бреттон-Вудської валютної системи. На зміну прийшла Ямайська валютна система, принципи якої були закладені в березні 1971 року на острові Ямайка за участю 20 найбільш розвинених держав некомуністичного блоку.

Суть змін, що відбулися, зводилася до більш вільної політики щодо цін на золото. Якщо раніше курси валют були стійкі в силу дії золотого стандарту, то після таких рішень плаваючий курс золота призвів до неминучих коливань курсів обміну між валютами. Це породило відносно нову галузь діяльності – валютну торгівлю, коли курс обміну почав залежати не тільки від золотого еквівалента валюти, але і від ринкового попиту та пропозиції на неї.

8 січня 1976 року на засіданні міністрів країн-членів МВФ в м Кінгстон (Ямайка) було прийнято нову угоду про устрій міжнародної валютної системи, яка мала вигляд поправок до статуту МВФ. Система замінила Бреттон-Вудську валютну систему. Багато країн фактично відмовилися від прив'язки національної валюти до долара або до золота. Однак лише в 1978 році МВФ офіційно дозволив таку відмову. Починаючи з цього моменту вільно плаваючі курси стали основним способом обміну валюти.

У новій валютній системі остаточно відбулася відмова від принципу визначення купівельної спроможності грошей на підставі вартості їх золотого еквівалента. Гроші країн-учасниць угоди перестали мати офіційний золотий вміст, обмін почав відбуватися на вільному валютному ринку за вільними цінами.

Становлення системи плаваючих курсів призвело до трьох істотних підсумків:

1. Імпорттери, експорттери та обслуговуючі їх банківські структури були змушені стати регулярними учасниками валютного ринку, оскільки зміни курсів валюти можуть позначитися на фінансових результатах їх роботи як з позитивного, так і з негативного боку.

2. Центральні банки отримали можливість впливати на курси національної валюти і на економічну ситуацію в країні ринковими методами, а не тільки адміністративними.
3. Курси найбільш ліквідних національних валют формуються на основі пошуку ринком точки рівноваги між поточним попитом і наявною пропозицією, а зміна попиту і пропозиції на ринку викликає зміщення валютного курсу в ту або іншу сторону.

1. 2. 3. Зростання обороту на валютних ринках

Вважається, що щоденний оборот на ринку FOREX становив:

- в 1977 році - 5 млрд. доларів;
- в 1987 році - 600 млрд. доларів;
- в кінці 1992 року - 1 трлн. доларів;
- в 1997 році - 1,2 трлн. доларів;
- у 2000 році - 1,5 трлн. доларів;
- в 2005-2006 роках обсяг денного обороту на ринку FOREX вагався, за різними оцінками, від 2 до 4.5 трлн. доларів;
- в 2010 році - 4 трлн. доларів;
- в 2013 році - 5,3 трлн. доларів;
- в 2016 році - 5,1 трлн. доларів.

При цьому, прискорення зростання щоденного обороту в 2013 р. склало 9% (зростання 2010 р. по відношенню до 2007 р. склало 21%, а з 2013 до 2010 року – вже 32,5%).

До 2020 року подальше зростання внутрішньоденного обороту на ринку FOREX може досягти 10 трлн. доларів.

Банк міжнародних розрахунків періодично проводить масштабне дослідження ринку FOREX кожні три роки, починаючи з 1989 р. Підсумковий звіт містить інформацію про обороти ринку, структуру і його динаміку.

Проте точні дані відсутні, так як це позабіржовий ринок, і немає вимоги обов'язкової реєстрації та публікації даних про угоди. Частина цього обсягу

забезпечує маржинальна торгівля, за умовами якої дозволяється укладати контракти на суми, які істотно перевищують дійсний капітал учасника угоди. Незалежно від характеру і мети угод, великий денний оборот є гарантією високої ліквідності цього ринку.

1. 2. 4. Ритейловий FOREX

У цій роботі зроблено акцент на фондовий і валютний ринок, зокрема, буде виділено FOREX, так як це найбільш поширена аббревіатура, яка знаходиться на слуху у значної частини населення. Багато в чому це заслуга «агресивної реклами», яка проводиться в СНД просторі.

Насправді, є велика різниця між реальним ринком міжбанківського обміну, і псевдо-ринком, який доступний більшості населення і реалізований за допомогою ритейлового FOREX. Роздрібний FOREX працює за принципом спред-бетінга і реалізований за допомогою CFD (contracts for difference) – що є ставкою на різницю між ціною відкриття позиції і її закриття, при цьому угоди трейдера дуже рідко виводяться на реальний ринок і за фактом укладаються проти свого брокера, який виступає в ролі контрагента.

Роздрібний (ритейловий) FOREX – це ринок позабіржових фінансових операцій з валютними контрактами на маржинальних умовах. FOREX для приватного інвестора.

Особливості ритейлового FOREX:

1. Маржинальна торгівля, при якій обсяги операцій значно (в десятки і сотні разів) перевищують наявні засоби.
2. Торгові інструменти (контракти на зміну валютних курсів) є розрахунковими, операції з ними зазвичай не припускають реальної поставки валют. Відкрита позиція закривається на дату розрахунків протилежною угодою.
3. FX компанії використовують дві основні технології роботи: власний дилінг і прямий доступ до електронних торговельних мереж (далі ECN).

У першому випадку компанія (дилер або брокер), в якій трейдер відкриває торговий рахунок, виступає контрагентом по операціях: виставляє котирування валют на основі курсів міжбанківського ринку FOREX, купує і продає інструменти (найбільш поширений та доступний варіант).

У другому випадку FX компанія забезпечує доступ до ECN.

4. Переважно спекулятивний характер угод.
5. Основний обсяг операцій здійснюється в електронному режимі (інтернет-трейдинг) – через інформаційно-торговельні термінали.

У різних країнах операції на роздрібному ринку FOREX мають свою правову кваліфікацію і можуть розглядатися, як валютні операції, похідні фінансові інструменти (деривативи) або угоди-парі (спред-беттінг).

1. 2. 5. Учасники торгів на валютному ринку

FOREX є міжнародним міжбанківським ринком. Операції проводяться через систему установ: центральні банки, комерційні банки, інвестиційні банки, брокери та дилери, пенсійні фонди, страхові компанії, транснаціональні корпорації тощо. Обсяг однієї угоди з реальним постачанням валюти на другий робочий день зазвичай становить близько 5 млн доларів США або їх еквівалент. Вартість одного конвертаційного платежу становить від 60 до 300 доларів. Крім цього, доведеться нести витрати до 6 тис. доларів на місяць за міжбанківський інформаційно-торгівельний термінал. У зв'язку з цим на FOREX не проводять прямих конвертацій невеликих грошових сум. Для цього дешевше звернутися до фінансових посередників (банку або валютного брокера), які проведуть конвертацію за певний відсоток від суми угоди. При великій кількості клієнтів і різноспрямованих заявок у посередників регулярно виникають ситуації внутрішнього клірингу (брокерської «кухні»), через що далеко не завжди потрібно проводити реальну конвертацію через FOREX. Але свої комісійні вони отримують з клієнтів завжди. Саме через те, що на FOREX потрапляють не всі клієнтські заявки, посередники можуть запропонувати клієнтам

комісійні, які істотно нижче вартості прямих операцій на FOREX. У той же час, якщо усунути посередників, вартість конвертації для кінцевого клієнта неминуче зросте.

Поточні котирування валюти використовуються для великої кількості операцій, які не обов'язково мають безпосередній вихід на FOREX. Прикладом може служити зміна курсу національної валюти державним банком, який змушений зберігати пропорції курсу між іноземними валютами відповідно до їх пропорцій на FOREX, навіть якщо справжні попит і пропозиція в країні не відповідає тенденціям на FOREX. Наприклад, якщо на внутрішньому ринку є надлишкова пропозиція євро, але при цьому на FOREX ціна євро проти долара збільшується, то центральний банк змушений буде піднімати ціну, а не знижувати під тиском надмірної пропозиції.

Інший яскравий приклад – маржинальна спекулятивна торгівля валютою, яка орієнтована на фіксацію поточних котирувань FOREX, але за своїми умовами проходить без реального постачання. Майже всі посередники на валютному ринку пропонують для клієнтів не тільки послуги з прямої конвертації, але і спекулятивну торгівлю з кредитним плечем. У більшості випадків, комісійні для таких операцій ще нижче, ніж для прямої конвертації, так як за рахунок масовості і короткочасності угод необхідність в укладенні реальних угод на поставку виникає ще рідше. Дуже часто комісійні набувають форми спреда – встановленої брокером різниці між ціною придбання валюти і ціною продажу в один і той же момент часу. У більшості випадків між FOREX і спекулянтом вибудовується ланцюжок з декількох посередників, кожен з яких бере свою комісію.

Маржинальні операції можуть призводити (але не обов'язково) до виникнення реального додаткового попиту або пропозиції на валютному ринку, особливо на короткостроковому відрізку часу. Але загальну тенденцію руху валютних курсів вони не формують.

Останнім часом на FOREX почали отримувати поширення системи торгівлі, спрямовані на зменшення впливу брокера (наприклад, ECN).

1. 2. 6. Маркетмейкери, їх роль в системі ринкового ціноутворення

Брокер і ДЦ може встановлювати власні курси валют для своїх клієнтів, оскільки на FOREX немає єдиного міжнародного розрахункового центру для їх утворення. Зрештою, угода виводиться до маркетмейкерів (далі – ММ, на заході – «спеціаліст»; в СНД слензі – «ляльковод») – найбільш великих учасників ринку, які забезпечують його ліквідність. Тобто це деякі фінансові установи, фірми або інститути, які мають обов'язок більшу частину часу перед іншими рівноцінними учасниками найвищого рівня або біржею (у випадку з фондовим ринком) бути і покупцем, і продавцем будь-якої акції або певного обсягу валюти.

Наприклад, здійснюється торгівля акцією компанії «АБВ». Маркетмейкер виставив заявки одночасно і на покупку, і на продаж цієї акції, припустимо за такими цінами 100,99 і 100,10. Різниця між ціною купівлі та продажу утворює **спред**. Якщо покупців і продавців буде приблизно порівну, то ціна суттєво не зміниться, а ММ отримає прибуток. Для більшості людей маркетмейкером умовно можна назвати банк, у якому вони проводять операції з конвертації валюти. Кожен банк виставляє ціни на покупку та продаж валют. Різниця між ними і є спред.

Маркетмейкери забезпечують ліквідність – можливість здійснювати операції іншим учасникам ринку (простим трейдерам) за прийнятими цінами. Фактично ММ більшу частину часу згладжує сильні цінові коливання. Стратегія маркетмейкерів полягає в заробітку на спреді.

Якщо ціна знаходиться в невеликому діапазоні (консолідація), то ММ буде мати прибуток. Якщо виникає сильний односпрямований рух (тренд), то можливі значні збитки для «фахівців». Внаслідок того, що більшу частину часу ринок перебуває в консолідації, ММ мають можливість трохи зміщувати ціни від загальноринкових, тому що це їм може бути вигідно. Звідси і з'явився термін «ляльковод».

Недосвідчені трейдери мають одну спільну рису з маркетмейкерами. І ті, і інші грають проти тренда і при сильному ціновому русі зазнають збитків. Тільки «фахівці» роблять свою справу професійно (за допомогою торгових роботів), а новачки керуються інтуїцією. Як правило, торгові роботи надійніші інтуїції.

1.3. Використання автоматизованих торгових систем

В останнє десятиріччя на світових біржах брокерів-людей багато в чому потіснили так звані «торгові роботи» – комп'ютерні програми, спеціалізовані для біржової торгівлі.

Торговий робот – це автоматизована торгова система, яка частково або повністю замінює людину під час роботи на біржі, при цьому робот може керуватися трейдером (наприклад, прийняття рішень про відкриття/закриття позиції приймає сам трейдер) або працювати за заздалегідь складеною програмою.

Роботи, які самостійно ведуть торги на біржі – це спеціально розроблені програми. Вони будуються на математичних алгоритмах, і можуть відстежувати показники різних індексів на фондовій біржі та на основі отриманих даних здійснювати операції з купівлі або продажу. Звичайний обсяг угод автоматизованих систем в кілька разів перевищує кількість угод, які здійснюють люди.

Фахівці зазвичай ділять біржових роботів на три групи: трендові (дирекційні або спрямовані), контртрендові та арбітражні. Вони відповідають різним типам торгових стратегій.

Звичайне завдання трендового робота – якнайшвидше вловити тенденцію зростання або падіння котирувань та відкрити позицію. Після чого своєчасно «відчути» розворот тренда і встигнути зафіксувати прибуток (тобто продати акції або валюту). Контртрендові роботи намагаються «зловити» всі відкати ціни, особливо вони добре працюють у флетовому стані ринку. У свою чергу,

арбітражний робот повинен отримувати прибуток, виявляючи перекося в цінах на ідентичні або тісно пов'язані активи на різних ринках.

Зокрема, роботів можна поділити на індикаторні (використовують в якості сигналів на відкриття і закриття угод індикатори) і свічкові (використовують в якості сигналів на відкриття і закриття угод комбінації свічкових моделей).

Сьогодні, за різними оцінками, програми-роботи проводять від 30% до 50% угод на фондових, термінових та валютних ринках планети. Автоматизовані торгові системи прийнято вважати одними з найбільш ефективних і надійних методів торгівлі. Перш за все, автоматична торгівля допомагає трейдерам у продажу великих пакетів цінних паперів. Заявка розбивається на кілька дрібних, і вони поступово виводяться на біржу за певним алгоритмом (алгоритмічна торгівля).

У звичайні торгові дні робот приймає рішення автоматично і проводить операції, не дозволяючи суб'єктивній оцінці взяти верх над об'єктивним технічним розрахунком. Вважається, що роботи більш ефективні в короткостроковій внутрішньоденній торгівлі, яка зазвичай будується на індикаторах технічного аналізу. Відсутність емоцій допомагає і коли справа доходить до фіксації збитків: продажу падаючих в ціні активів.

Якщо механізм здійснення операцій на даній конкретній біржі можна алгоритмізувати, а сам трейдер не в змозі (або не хоче) особисто обробити весь обсяг даних, має сенс передати торгівлю роботу.

До переваг торгових роботів можна сміливо віднести:

- Повна відсутність емоцій.
- Швидкість реакції – від надходження сигналу до відкриття (закриття) позиції та виконання заявки витрачаються частки секунд.
- Повне дотримання правил торгової стратегії без будь-яких відхилень.
- Цілодобова працездатність торгового робота, на відміну від людини.
- Швидкість обробки даних, що надходять. Робот може відстежувати дані по сотні інструментів водночас.

- Роботи можуть торгувати одночасно сотні алгоритмів, чого звичайна людина ніколи не зможе зробити.

До недоліків же можна віднести:

- Ризик помилки в кодї програми. Усувається ця помилка тестуванням.
- Переважне використання технічного аналізу.

Кожен трейдер має право самостійно вирішувати про встановлення торговельного робота на своєму терміналі. Можна придбати вже готовий робот, але ціни при цьому будуть чималі.

Торгові роботи і системи можуть виконувати велику кількість завдань, спрощуючи різноманітні аспекти трейдингу. Але має значення: хто і яким чином торгує і для чого займається торговельною діяльністю. Наприклад, звичайна людина, що торгує через брокера в його кліринговій системі, з свого домашнього ПК, та інституціональний інвестор у вигляді хедж-фонду, який має велику кількість людей і засобів, володіють різними можливостями взаємодії з ринком. Саме тому раніше був відзначений такий феномен, як «роздрібний» FOREX, де основна аудиторія трейдерів – це приватні особи, які не мають нічого спільного зі світом великих грошей. І якщо у таких людей основне завдання – це додатковий заробіток, а торговий робот – потенційно легка можливість досягти цієї мети; то на реальних світових ринках необхідність автоматизувати торги з'явилася не тільки заради отримання прибутку. Великі гравці використовують високочастотних роботів задля реалізації своєї торговельної діяльності, як такової.

1.3.1. Алгоритмічна торгівля

Алгоритмічна торгівля, або Алгоритмічний трейдинг – це метод виконання великої заявки (занадто великої, щоб бути виконаною за один раз). За допомогою особливих алгоритмічних інструкцій така заявка (parent order) ділиться на кілька під-заявок (child orders) зі своїми характеристиками ціни та обсягу і кожна з під-заявок відправляється в певний час на ринок для виконання. Такі алгоритми були придумані для того, щоб трейдерам не

доводилося постійно стежити за котируваннями і ділити велику заявку на менші вручну. Популярні алгоритми носять назви "Percentage of Volume", "Pegged", "VWAP", "TWAP", "Implementation Shortfall", "Target Close".

Алгоритмічна торгівля не ставить за мету отримати прибуток. Її мета – зменшити вартість виконання великої заявки (transaction cost), мінімізувати її вплив на ринок (market impact) і зменшити ризик її невиконання.

Термін "алгоритмічна торгівля" часто помилково використовується в тих випадках, коли мова йде про автоматизовані торгові системи. Перед такими системами дійсно ставиться мета отримати прибуток.

1. 3. 2. Вплив алгоритмічних систем на ліквідність

Ліквідність фінансових інструментів зазвичай оцінюють за обсягом і кількістю здійснюваних операцій (обсяг торгів), величиною спреду між кращими цінами попиту та пропозиції (максимальними цінами заявок на покупку і мінімальними цінами заявок на продаж) і сумарного обсягу заявок поблизу кращих цін попиту і пропозиції (ціни та обсяг поточних заявок можна побачити в склянці торгового терміналу). Чим більше обсяг і кількість угод по інструменту, тим більше його торгова ліквідність, в свою чергу, чим менше різниця між кращими цінами попиту та пропозиції і чим більше обсяг заявок поблизу цих цін, тим більше моментальна ліквідність.

Існує два основних принципи виставлення заявок:

- котировочний (instant execution) – виставлення заявок з метою здійснення угоди за вигіднішою ціною, ніж поточні кращі ціни попиту або пропозиції;
- ринковий (market execution) – виставлення заявок з метою моментального здійснення операції за поточними цінами попиту або пропозиції.

Заявки, виставлені котировочним принципом, формують моментальну ліквідність ринку, дозволяючи іншим учасникам торгів у будь-який момент часу купити або продати певну кількість активу.

Заявки, виставлені за ринковим принципом, формують торгову ліквідність ринку, дозволяючи іншим учасникам торгів купити або продати певну кількість активу за бажаною ціною.

Алгоритмічні торгові системи, що використовують котировочний принцип, є одними із основних постачальників моментальної ліквідності, а ті що використовують ринковий принцип – одними з основних постачальників торгової ліквідності. Велика кількість алгоритмічних систем одночасно використовують обидва ці принципи.

1. 3. 3. Види спекулятивної торгівлі

Основною метою спекулятивних стратегій є отримання доходу в короткостроковому періоді за рахунок коливань ринкових цін фінансових інструментів. З метою класифікації, можна виділити вісім основних груп спекулятивних стратегій, деякі з яких використовують принципи і алгоритми інших груп, або є їх похідними (рис. 1.1).



Рис. 1.1. Діаграма залежностей між різними спекулятивними алгоритмічними стратегіями

1.3.4. Способи підключення автоматизованих систем до ринкових торгів

Для більшості алгоритмічних систем швидкість отримання ринкових даних і швидкість виставлення заявок є найважливішими факторами, що впливають на ефективність роботи системи. На ринку історично склалися шість різних варіантів підключення роботів до біржових торгових систем. Як приклад розглянемо варіанти доступу до торгового майданчику FORTS (рис. 1.2).

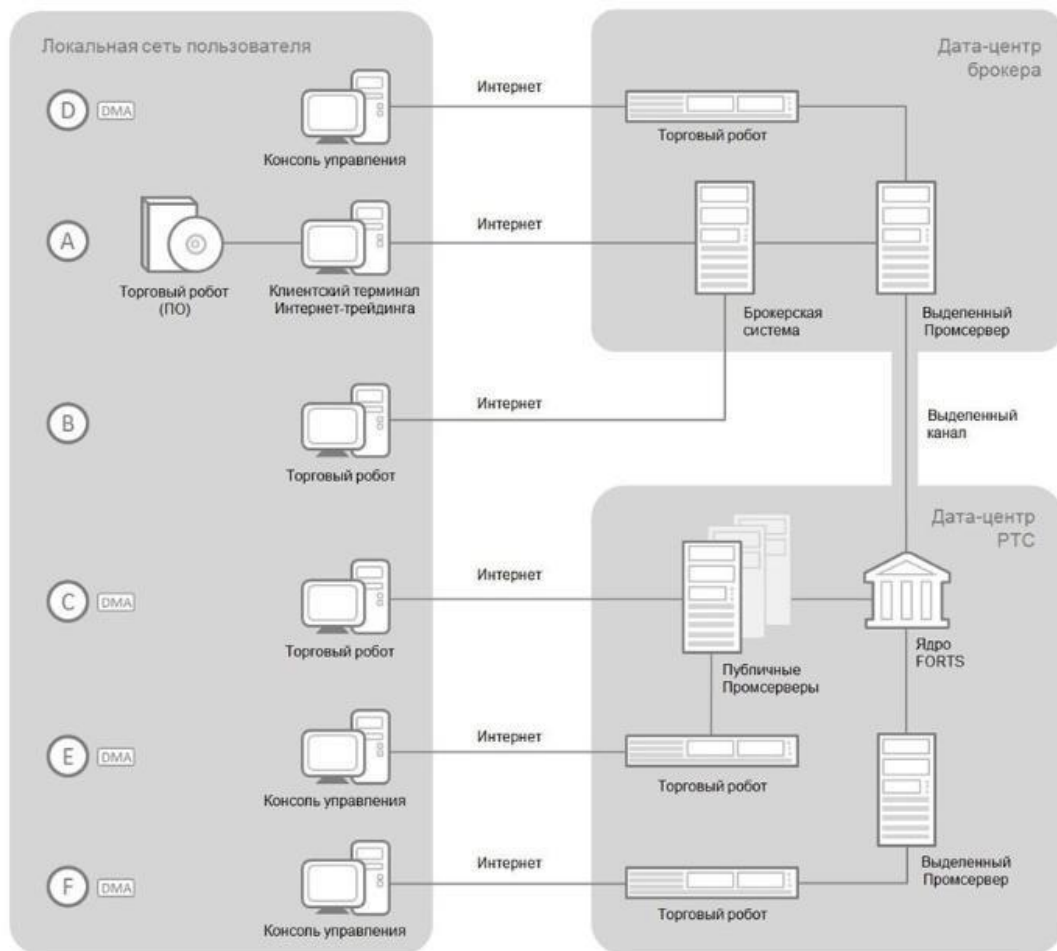


Рис. 1.2. Способи взаємодії клієнтського торгового робота з біржовим або брокерським API

Варіант А – є найекономічнішим з усіх існуючих, так як не вимагає практично ніяких витрат на інфраструктуру. Недоліками цього варіанта доступу є максимально можливі затримки в отриманні ринкових даних і низька швидкість виставлення заявок, що пов'язано з великою кількістю проміжних

ланок між торговим роботом і ядром торгової системи. Крім того, існує цілий ряд зовнішніх ризиків, які можуть призвести до нестабільної роботи алгоритмічної системи. Наприклад, обрив інтернет з'єднання або збій брокерської системи. У зв'язку з цим, даний варіант доступу не рекомендується застосовувати для високочастотної торгівлі.

Варіант В – практично ідентичний попередньому, з тією лише відмінністю, що для підключення торгового робота до брокерської системи не потрібна наявність торгового терміналу. Незважаючи на виключення однієї проміжної ланки, цей варіант доступу майже не відрізняється за величиною затримок даних, швидкості роботи з заявками, і зовнішніх ризиків від Варіанта А, що теж робить його непридатним для високочастотного трейдингу.

Варіанти С, D, E і F відносяться до групи варіантів прямого доступу (далі DMA) до ринків і характеризуються підключенням торгового робота безпосередньо до ланки біржової торгівельної інфраструктури, в даному випадку до проміжного сервера (промсерверу) FORTS. У зв'язку з мінімальною кількістю ланок, DMA є оптимальним рішенням для алгоритмічних систем високочастотної торгівлі.

Варіант С – найпростіший в реалізації і найекономічніший варіант DMA. В даному випадку, всі інфраструктурні витрати обмежуються платою за доступ до промсерверу, через який торговий робот отримує ринкові дані і виставляє заявки. Цей варіант доступу, істотно перевершує за всіма характеристиками варіанти А і В, проте у нього є один дуже важливий недолік, пов'язаний з використанням інтернет-з'єднання для взаємодії з біржовою торгівельною інфраструктурою. Справа в тому, що підключення через інтернет не гарантує якості передачі даних, оскільки на шляху від торгового робота до біржі знаходиться безліч маршрутизаторів, на кожному з яких можуть виникати черги з пакетів даних, що призводять до істотного зниження швидкості їх передачі або до їх втрати.

Варіант D – дозволяє усунути ризики, пов'язані з Інтернет-з'єднанням, за допомогою передачі торгових даних через виділений канал зв'язку, який

забезпечує стабільну швидкість передачі з мінімальними втратами. Для цього необхідно розмістити торгового робота в дата-центрі брокера, що передбачає додаткові витрати на придбання апаратного забезпечення (сервера) для торгового робота і на оплату послуги розміщення сервера в дата-центрі (колокейшн). Незважаючи на всі переваги цього варіанту, у нього залишається ряд зовнішніх ризиків, обумовлених загальним використанням виділеного каналу і промсервера усіма клієнтами брокера, що може призвести до «забивання» виділеного каналу або перевантаження промсервера.

Варіант Е – дозволяє виключити ризик, пов'язаний з виділеним каналом зв'язку і домогтися мінімальних затримок в отриманні ринкових даних, за допомогою розміщення торгового робота в дата-центрі РТС. Але максимальна близькість до біржової торгівельної інфраструктури, зажадає додаткових витрат, серед яких оплата отримання даних з локальної мережі РТС і оплата доступу в інтернет, для можливості віддаленого управління торговим роботом.

Варіант F – по праву вважається самим надійним і ефективним з усіх існуючих варіантів доступу, завдяки виключенню останнього зовнішнього ризику, обумовленого використанням промсервера декількома учасниками торгів. Для реалізації цього варіанту будуть потрібні витрати на придбання апаратного забезпечення для промсервера, оплату ліцензії та інсталяції програмного забезпечення для нього, а також на оплату послуги його розміщення в дата-центрі РТС.

Варто зауважити, що вищевикладені способи взаємодії клієнтського робота з ринком актуальні лише для реального FOREX. У випадку з роздрібним FOREX, в більшості варіантів використовується спосіб А, за умови, що брокер виводить великі угоди за межі клірингової системи. Інакше, варіант А набуває наступного вигляду (рисунок 1.3)



Рис. 1.3. Операції всередині «FOREX кухонь» не потрапляють на реальний ринок, а обробляються силами самого брокера.

1. 3. 5. Торгові роботи на ритейловому FOREX

У пункті 1.2.4. був відзначений той факт, що основна аудиторія ритейлового FOREX це люди, які дуже рідко мають спеціальні знання про світові ринки, про ті економічні та геополітичні процеси, які стоять за рухами на ринках, і про те, як торгувати, використовуючи легальні та законні способи.

В Інтернеті існує достатня кількість ресурсів, де поширюються торгові радники, як безкоштовні, так і платні. Ціни на платних роботів можуть досягати 2000 USD, але зазвичай вони коштують в діапазоні 500-800 USD.

В обох випадках є ряд недоліків:

- безкоштовні роботи і радники не ефективні. Вони вільно поширюються на тематичних форумах і порталах, проте вони реалізують 1-2 торгові стратегії (наприклад, торгівля з використанням показань індикатора MACD). Найчастіше «сліпе» виконання угод без комплексного аналізу призведе до збитків. До того ж цьому є просто логічне пояснення – жоден хороший продукт не буде поширюватися безкоштовно;
- платні роботи для більшості є дорогим «задоволенням». До того ж, ця сфера ніяк не регулюється чинним законодавством, тому, купуючи радника в інтернеті, покупець діє на свій страх та ризик. Відсутність можливості тестування ефективності робота, а також законодавчого регулювання дозволяє шахраям продавати «повітря».

1. 3. 6. Особливості автоматизованої торгової системи з використанням патернів Price Action

Вдосконалення методики побудови торгової системи полягає в комбінуванні різних типів аналізу ринкової ситуації, де різноманітні торгові сигнали підтверджують або спростовують один одного. Комбінація сигналів, яка підтверджує єдину поведінку ринку в короткочасній перспективі, вважається достатньою надійною в співвідношенні прибуток/ризик. Так як даний програмний продукт націлений на найбільшу кількість потенційних споживачів, отже сфера його реалізації – роздрібний FOREX. Знаючи особливості даної області, можна конкретизувати основні завдання та вимоги, яким повинен задовольняти готовий продукт, а саме:

- простота використання. Неосвідчений користувач матиме можливість інсталиувати радника з-під свого терміналу, при цьому, не вносячи жодних змін в його налаштування для коректної роботи;
- гнучкість настройки. Готовий робот буде містити гнучку торговельну систему, що складається з декількох стратегій на основі технічного та свічкового аналізу (в першу чергу, патернів Price Action);
- можливість тестування. Користувач зможе протестувати ефективність радника протягом одного місяця на демо-рахунку.

Висновки

Ознайомившись з вищевказаною інформацією про світові ринки та FOREX, а також про індустрію торгових роботів для широкого споживача, можна виділити такі моменти:

1. Світові ринки виникли в результаті необхідності створення стійкого механізму для міжнародних економічних відносин і руху різного капіталу. Він, в свою чергу, складається з декількох сегментів, одним з яких є FOREX – ринок міжбанківського обміну валютою.
2. Основними учасниками торгів на FOREX є центральні, комерційні та інвестиційні банки, брокери і дилери, пенсійні фонди, страхові компанії, транснаціональні корпорації тощо.
3. Угоди на FOREX проходять через систему фінансових установ від менших до більших і виводяться на реальний ринок при необхідності.
4. Обіг коштів на FOREX щорічно зростає, що робить його самим високоліквідним ринком в світі.
5. Велика частина населення в СНД стикається з агресивною рекламою роздрібного FOREX. Під час торгівлі в організаціях, які доступні для більшості, люди мають справу з FOREX «кухнями» – угоди в таких компаніях ніколи не виводяться на реальний ринок, а виконуються всередині клірингової системи брокера, де контрагентом виступає сам брокер.
6. Маркетмейкери – найбільші учасники ринку, які зобов'язуються згладжувати цінові коливання і заробляють за рахунок спреда – різниці між ціною покупки і продажу торгового інструменту.
7. Торгові роботи з'явилися, в першу чергу, для забезпечення можливості торгів великими капіталами на ринках. Використовувати їх з метою заробітку – ідея останнього десятиліття.
8. Для широкого споживача доступні варіанти автоматизації торгівлі пов'язані з великими ризиками.

РОЗДІЛ 2. МЕТОДИ АНАЛІЗУ І ПРОГНОЗУВАННЯ РУХУ ЦІНИ НА СВІТОВИХ РИНКАХ З МЕТОЮ ОТРИМАННЯ ПРИБУТКУ

2. 1. Існуючі методи аналізу торгових інструментів

З появою ринку та інвестицій було розроблено велику кількість методик прогнозування руху ціни найрізноманітніших торгових інструментів: від акцій і валютних пар, до сировинних товарів, ф'ючерсів і індексів. Всим цим можна торгувати заради отримання прибутку або хеджування ризиків вже відкритих ринкових позицій. Кожен тип ринку (валютний, фондовий, ф'ючерсний тощо) має свої особливості ціноутворення, а тому одні і ті ж види аналізу можуть бути по-різному ефективні.

Історично так склалося, що всі види аналізу можна зарахувати по двом основним категоріям – технічний і фундаментальний аналіз. В їх основі лежить дві абсолютно різні характеристики, які відображає поточна ринкова ситуація. Технічний аналіз передбачає роботу з історичними даними, що формують графік руху котирувань. Фундаментальний аналіз передбачає роботу з економічними (і не тільки) показниками, які допоможуть знайти недооцінені активи та інструменти.

2. 2. Фундаментальний аналіз

Завдання по розробці торгового робота ставить питання про те, чи можна використовувати фундаментальний аналіз і його методики при програмуванні механізмів торгівлі, адже це невід'ємний тип прогнозування руху активів і ціни. В цьому пункті коротко надано опис даної школи і відповідь на питання: чому його дуже складно використовувати для автоматичної торгівлі.

Фундаментальний аналіз – термін для позначення методів прогнозування ринкової (біржової) вартості торгового інструменту, заснованих на аналізі фінансових і виробничих показників, що характеризують його потенційний прибуток.

Фундаментальний аналіз використовується інвесторами для оцінки вартості компанії (або її акцій), яка відображає стан справ в компанії, рентабельність її діяльності. При цьому аналізі піддаються вивченню фінансові показники компанії: виручка, EBITDA (Earnings Before Interests, Taxes, Depreciation and Amortization), чистий прибуток, чиста вартість компанії, зобов'язання, грошовий потік, величина виплачуваних дивідендів і виробничі показники компанії.

«Внутрішня вартість» в більшості випадків не збігається з ціною акцій компанії, яка визначається співвідношенням попиту і пропозиції на фондовому ринку. Інвесторів, які використовують у своїй діяльності фундаментальний аналіз, цікавлять в першу чергу ситуації, коли «внутрішня вартість» акцій компанії перевищує ціну акцій на біржі. Такі акції вважаються недооціненими і є потенційними об'єктами інвестицій. Купуючи недооцінені акції, інвестори розраховують, що в умовах неефективності ринку ціна акцій на фондовому ринку буде прагнути до «внутрішньої вартості», тобто в разі недооцінених акцій – буде рости. Це твердження є зворотним до постулату технічного аналізу, який стверджує, що вся суттєва інформація негайно і в повній мірі відбивається на ринковій вартості цінних паперів. А цей принцип зводить нанівець ідею фундаментального аналізу.

В основі американської школи фундаментального аналізу лежить класична праця Бенджаміна Грема і Девіда Додда «Аналіз цінних паперів», опублікований ними в 1934 році. Сам Грем користувався фундаментальним аналізом на практиці і був успішним інвестором. Один з найбільш відомих послідовників Грема, що використовують фундаментальний аналіз – Уоррен Баффетт.

Фундаментальний аналіз спирається на макроекономічні індикатори та індекси ділової активності. Використовувати фундаментальний аналіз на ринку FOREX можна наступним чином.

Вивчається макроекономічна статистика за останні півроку. Наприклад, якщо трейдер хоче працювати з парою EUR/USD, йому необхідно вивчити дані,

що надходять з Єврозони та США, оцінити геополітичну обстановку і перспективи монетарної політики цих країн, а саме – рівень процентної ставки. Чим вона вища, тим вигідніше буде зберігати капітал в якійсь з валют.

Блоки економічних новин постійно публікуються відомствами різних країн, які характеризують ті чи інші сторони економічного розвитку і благополуччя держави. Як правило, більшість інформації публікується раз на місяць, наприклад, звіт про стан ринку праці в США (Non-Farm Payrolls) викладається в першу п'ятницю місяця о 15:30 за Києвом. Так як інформація про дати публікації відома заздалегідь, то на багатьох інтернет-ресурсах, пов'язаних з торгівлею на FOREX є економічний календар (рис. 2.1).

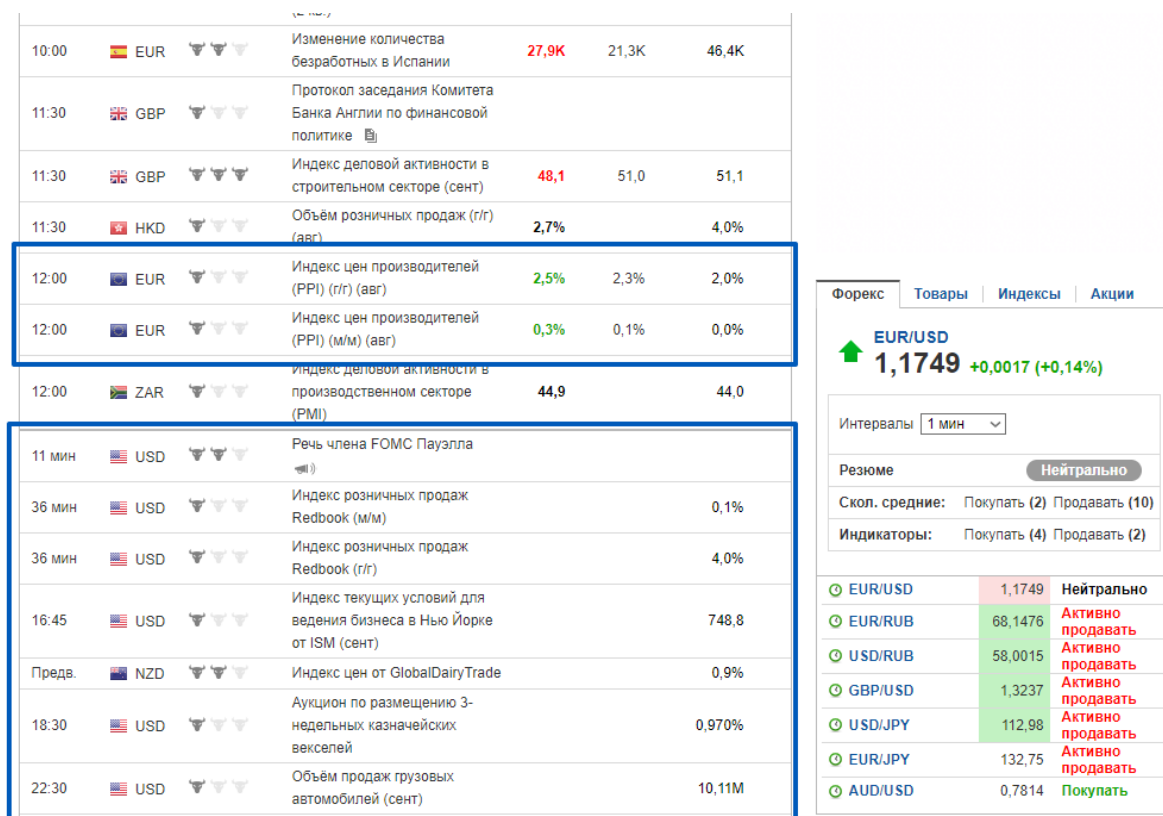


Рис. 2.1. Приклад економічного календаря з сайту investing.com

На скріншоті вище виділені блоки економічних новин з Єврозони (які вже опубліковані) і США (які ще будуть опубліковані). Індикатор поруч з назвою валюти вказує на «силу новини» – чим вище, тим більш значуща статистика публікується, а значить ринок може сильніше відреагувати.

Більшість економічної статистики використовується центробанком при формуванні монетарної політики. Коли економіка зростає, банк стає на шлях монетарної рестрикції – підвищення ключової процентної ставки. Чим вища відсоткова ставка, тим вигідніше купувати дану валюту, тримати в ній капітал і видавати її в кредит третім особам.

При фундаментальному аналізі велику роль має соціальна і геополітична стабільність і плани уряду щодо поліпшення економічної ситуації в державі.

Розглянемо два приклади глобальних подій і їх наслідків, представлених на рисунках 2.2-2.3.

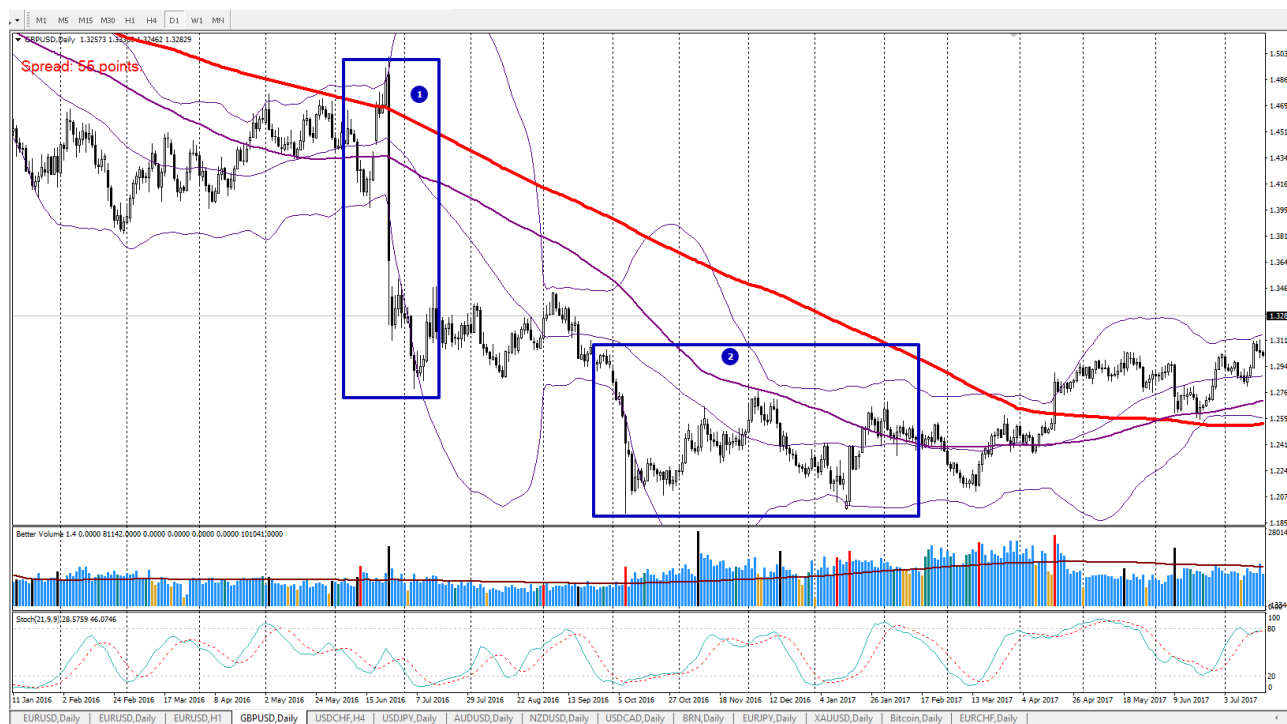


Рис. 2.2. Графік GBP/USD

На малюнку 2.2 показаний графік вартості фунта стерлінгів проти долара США. Під цифрою 1 відзначено внутрішньоденне падіння фунта в день проведення і оприлюднення результатів референдуму Вrexit за підсумками якого Великобританія відокремилася від ЄС. Під цифрою 2 відзначено подальше ослаблення фунта, на тлі різкого погіршення економічних настроїв і падіння рівня життя в Великобританії через обвал національної валюти, яке

відбулося влітку раніше. І якщо першу ситуацію неможливо було передбачити за допомогою фундаментального аналізу, то подальший несприятливий розвиток подій для фунта стерлінгів цілком можна було спрогнозувати, використовуючи загальний аналіз ситуації і економічні показники.

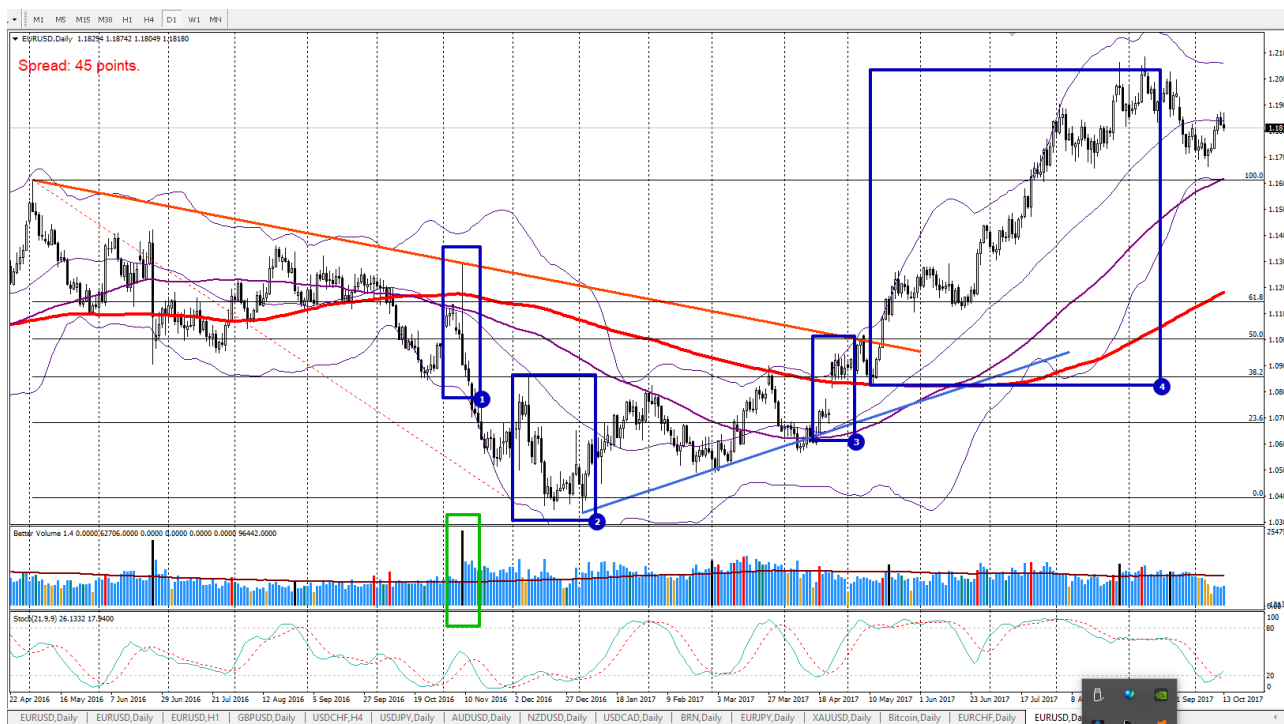


Рис. 2.3. Графік EUR/USD

Рисунок 2.3. показує, як зміни політичної ситуації в США та ЄС впливають на монетарну політику їх центробанків. Даний графік чітко показує ланцюжок типу «причина та наслідок», де одні події, які складно передбачити з певною часткою впевненості призводять до інших, вже більш очікуваних і логічних.

1. Вибори президента США 2016. Раптова перемога Дональда Трампа проти Хіллари Клінтон. На графіку видно, що спочатку гравці ринку почали різко розпродавати долар, однак потім стався ключовий розворот і долар почав різко зростати в ціні, що задало тренд на найближчі 2 місяці. Початковий розпродаж стався на фоні побоювань і паніки, адже гравці ринку очікували перемоги Клінтон, однак потім вхід пішли аргументи передвиборної компанії Трампа, який обіцяв

провести податкову реформу, що мало прискорити зростання економіки. І як говорилося вище, зростання економіки – це зростання процентних ставок. Необхідно звернути увагу на обсяг торгів в той день – він був значно вище.

2. Подальше ослаблення євро проти долара пов'язане з монетарною дивергенцією, яка полягала в тому, що Федеральна Резервна Система США збільшила процентну ставку на 25 базисних пунктів і пообіцяла зробити це 3 рази в 2017 році. Завдяки перемозі Трампа, фондовий ринок США оновлював історичні максимуми, а економічні показники були на висоті.
3. Перемога Еммануеля Макрона на виборах у Франції. Політичні ризики, пов'язані з розпадом ЄС, зійшли нанівець, що викликало різке зростання курсу євро проти світових валют (на графіку видно геп – розрив між ціною закриття одного дня і ціною відкриття наступного). Це багато в чому визначило злам тренду на зростання долара, новим трендом, де вже домінував євро.
4. Зниження політичних ризиків, дешевий євро (на момент зими 2016-2017 рр.) прискорив зростання економіки ЄС, її ділової та інвестиційної активності, а також покращив настрої споживачів. Все це призвело до того, що чиновники ЄЦБ (Європейського центрального банку) заговорили про зміну монетарного курсу та зменшенні програми стимулювання скупки активів QE. Це означає, що курс монетарної експансії в найближчому майбутньому змінить курс монетарної рестрикції. Чутки про це, а також численні економічні підтвердження зростання добробуту ЄС вилилися в зростання євро в травні-вересні 2017.

Розглянуті вище ситуації показують, як працює фундаментальний аналіз і на що він спирається, якщо мова йде про ринок FOREX. На даному етапі розвитку технологій програмування дуже складно описати ці ланцюжки «причина і наслідок», чутки, які зароджуються на ринках та інші геополітичні

події, пов'язані з діяльністю світової політичної і економічної еліти. Неможливо передбачити, що скаже президент США і тим більше передбачити реакцію ринку на його слова. Єдиний аспект цієї школи, який можна запрограмувати – це облік економічних показників: порівняння значень прогнозів експертів з результатами різноманітних звітів і економічної статистики, однак в даній роботі цей аспект не використовується.

Фундаментальний аналіз використовується великими гравцями ринку для довгострокових інвестицій і більше застосовується на фондових ринках в силу того, що аналізувати необхідно лише діяльність компанії, її конкурентів і галузі в цілому. Це набагато більш вузький спектр чинників, ніж при аналізі на FOREX. Багато в чому через різноманітність факторів довгострокове прогнозування на FOREX не користується попитом і тут переважають короткостроковий і середньостроковий тип торгівлі, де набагато краще проявляє себе школа технічного аналізу і різні його види.

2. 3. Технічний аналіз

2. 3. 1. Поняття технічного аналізу

При розробці алгоритму прийняття торгових рішень, єдине з чим матиме справу математична модель автоматизованої торгівельної системи – це числа. Графік ціни формує історія котировок, а за допомогою різних прийомів, які будуть описані нижче, створюються різні інтерпретації даного графіка.

Технічний аналіз – сукупність інструментів прогнозування ймовірної зміни цін на основі закономірностей змін цін в минулому в аналогічних обставинах. Базовою основою є аналіз графіків цін – «чартів» і/або біржової склянки. Теоретично, технічний аналіз можна застосувати на будь-якому ринку. Але найбільшого поширення він отримав на високоліквідних вільних ринках, яким і є FOREX.

В кінці XIX століття американський журналіст Чарльз Доу опублікував серію статей про ринки цінних паперів, які лягли в основу теорії Доу і послужили початком бурхливого розвитку методів технічного аналізу на

початку ХХ століття. Розвиток комп'ютерних технологій у другій половині ХХ століття сприяв удосконаленню інструментів і методів аналізу, а також появи нових методів, які використовують можливості обчислювальної техніки.

У технічному аналізі безліч інструментів та методів, але всі вони засновані на одному припущенні: динаміка руху ціни визначається психологією поведінки учасників ринку, які в подібних обставинах під впливом людських інстинктів – жадібності, стадності, страхів тощо – поведуться однаково, формуючи потоки попиту й пропозиції, а також ступінь дисбалансу між ними. Це і дозволяє прогнозувати рух цін.

Технічний аналіз не розглядає причини того, чому ціна змінює свій напрямок (наприклад, внаслідок низької прибутковості акцій, коливань цін на інші товари або зміни інших умов), але враховує лише факт, що ціна рухається в тому чи іншому напрямку, або певним чином, наприклад, перебуваючи в межах якогось інтервалу цін протягом деякого часу. Різні методики можуть також враховувати обсяги торгів, суму відкритих позицій, обсяги виставлених заявок на купівлю та продаж тощо.

З точки зору трейдера, дохід може бути отриманий при будь-якому стані руху ринку, якщо правильно розпізнати тренд і відкрити позицію в напрямку тренда, а потім вчасно закрити торгівельну позицію. Наприклад, якщо ціна починає зростати, треба відкривати позицію на покупку і чекати закінчення росту. Якщо ціна має тенденцію до зниження, треба відкривати позицію на продаж або закривати раніше зроблену покупку.

У технічному аналізі часто розглядають і аналізують так звані патерни або моделі – типові комбінації, що формуються на цінових графіках. Прикладами можуть служити розворотні моделі («Потрійна вершина», «Подвійна вершина», а також зворотні до них «Потрійне дно» «Подвійне дно»), моделі продовження тренда (різноманітні «Трикутники», «Прапор» тощо).

Важливе значення мають цінові рівні. Зазвичай в їх ролі виступають попередні максимуми/мінімуми цін. При підході ціни до попередніх пікових значень учасники ринку очікують аналогічних розворотів і прагнуть виставити

відповідні накази (торгові ордера). Це формує відповідні рівні «опору» або «підтримки», через які ціні важко пройти. Різні методики розглядають моделі відскоку ціни від рівня або прориву рівня.

Висновки, отримані на підставі технічного аналізу, можуть розходитися з висновками з фундаментального аналізу, який ґрунтується на припущенні, що поточна ринкова ціна товару (цінного паперу, валютної пари) відрізняється від якоїсь «реальної вартості» (вона переоцінена або недооцінена), але прагне до неї. Якщо розрахувати «правильну» ціну, то можна очікувати, що «ринок скоректується» до цього рівня.

Необхідно відзначити, що обидва методи не враховують можливого спотворення ринкової кон'юнктури під впливом різних зовнішніх факторів. Політичні, економічні події, як в світі, так і всередині країни здатні надати серйозний вплив на поведінку як окремих акцій, так і цілих секторів. Такі дії можуть бути пов'язані з природними або техногенними катастрофами, зміною законів, введенням або скасуванням мит, зміною ставок податків і інших подій.

2. 3. 2. Принципи технічного аналізу

Існує три принципи на яких ґрунтується технічний аналіз:

1. «Рухи цін на ринку враховують всю інформацію»

Згідно цій аксіомі вся інформація, що впливає на ціну товару, вже врахована в самій ціні і обсязі торгів, тому немає необхідності окремо вивчати залежність ціни від політичних, економічних та інших факторів. Досить зосередитися на вивченні динаміки ціни/обсягу і отримати направлення найбільш ймовірного розвитку ринку.

2. «Рух цін підпорядкований тенденціям»

Кожна окрема зміна ціни може здаватися випадковою. Але сукупність змін за певний проміжок часу може формувати деякі тренди (тенденції), тобто тимчасові ряди цін можна розбити на інтервали, в яких переважають зміни цін в певних напрямках.

Коли попит перевищує пропозицію – виникає висхідна тенденція і навпаки, якщо пропозиція перевищує попит – виникає спадна тенденція. У випадку, коли попит і пропозиція врівноважують один одного – виникає горизонтальна тенденція (бокова, флет, або говорять, що «немає тренда»).

Тенденції не бувають нескінченними. Вважається, що ймовірність продовження поточної тенденції вище, ніж вірогідність її зміни. Але на певному етапі тренд може змінитися протилежним рухом.

3. «Історія подібна»

Учасники ринку в схожих обставинах поведуться однаково, формуючи схожу динаміку зміни цін. Це дозволяє застосовувати графічні моделі зміни цін, виявлені на основі аналізу історичних даних.

2. 4. Методи та інструменти технічного аналізу

2. 4. 1. Класичний аналіз

Як було зазначено раніше, основа технічного аналізу – це робота з графіком зміни ціни. Підхід до аналізу починається з вибору типу графіка.

Існує 3 основних типи:

1. лінійний;
2. японські свічки;
3. бари;

Всі інші типи є похідними від цих, наприклад, графік «Ренко», похідний від японських свічок.

Лінійний графік (Line chart) – вид графіка, який використовується в класичному аналізі. Вихідними даними для його побудови є ціна відкриття (максимуму, мінімуму або закриття) періоду. Лінійний графік будується в прямокутній системі двох координат. По горизонтальній осі відкладається час, по вертикальній осі – ціна. Кожен елемент (найменша складова частина) лінійного графіка – це точка, яка відповідає значенню валютного курсу за

одиничний період (хвилину, годину, день, тиждень тощо). Найчастіше використовується ціна закриття, або середнє значення.

Після того, як ціна періоду зафіксована на графіку, відбувається переміщення на одиничний інтервал вправо для нанесення значення ціни наступного періоду в новій точці. Послідовність з'єднаних безперервної лінією точок утворює лінійний графік.

Приклад лінійного графіка наданий на рисунку 2.4.

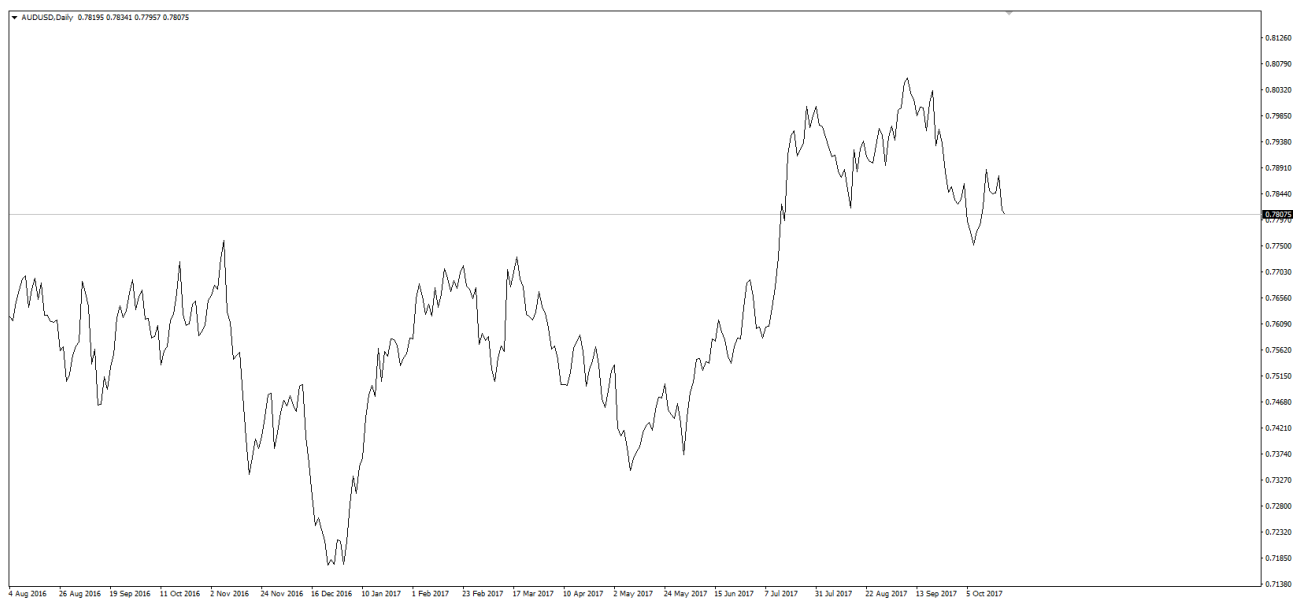


Рис. 2.4. Денний графік AUD/USD при лінійному відображенні

Лінійний графік зручний для загальної оцінки ринкової тенденції, однак на ньому не видно гепи (цінові розриви), а також не ясний характер зміни ціни всередині кожного торгового періоду. На цей тип графіку накладаються різні лінії (тренда, підтримки та опору) і графіки технічних індикаторів.

Японські свічки – це вид інтервального графіка, який застосовується головним чином для відображення в торговому терміналі змін цін на різні фінансові інструменти. Він найчастіше використовується при технічному аналізі.

Японська свічка складається з тіла (чорного або білого кольору) і тіні (верхньої або нижньої). Верхня та нижня межа тіні відображає максимум і

мінімум ціни в ході торгів відповідно, в той час як кордони тіла відображають ціну відкриття і закриття японської свічки.

Якщо котировки росли, то тіло японської свічки біле, нижня межа тіла відображає ціну відкриття, а верхня – ціну закриття (рисунок 2.5).

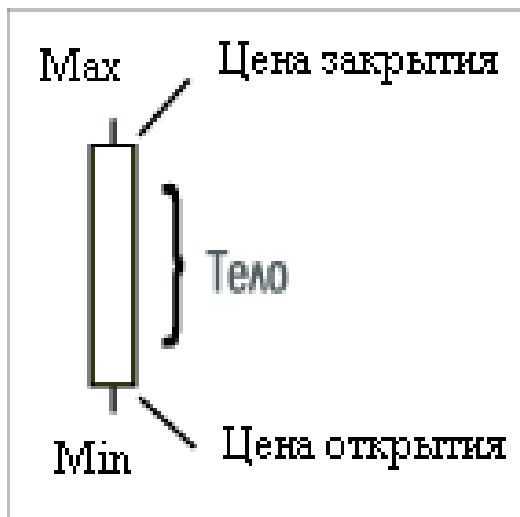


Рис. 2.5. Тіло японської свічки біле

Якщо котировки падали, то тіло японської свічки чорне, верхня межа тіла відображає ціну відкриття, нижня – ціну закриття, як це показано на рисунку 2.6.

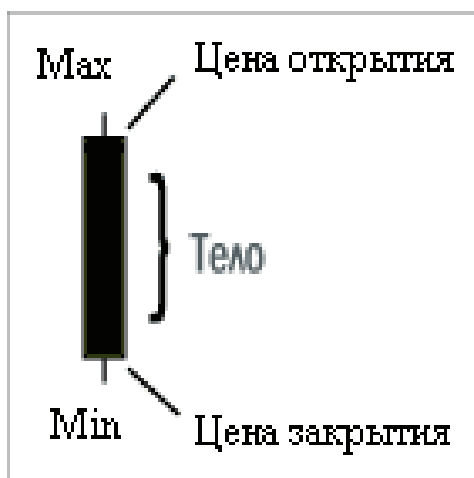


Рис. 2.6. Тіло японської свічки чорне

Нижче представлений той же торговий інструмент, що і на рисунку 2.4, тільки у вигляді японських свічок (рисунок 2.7).



Рис. 2.7. Денний графік японських свічок AUD/USD

Якщо порівняти лінійний графік і графік японських свічок, то видно, що останній куди більш точно відображає поведінку ціни. Це дозволяє набагато чіткіше бачити важливі цінові рівні і те, як ринок поводить себе на цих рівнях.

Графік бар (bar chart) – один з відомих способів відображення зміни ціни, що використовується замість японських свічок. Незважаючи на максимальну популярність японських свічок, графік бар продовжує використовуватися певною кількістю трейдерів. Це можна пояснити особливістю сприйняття, адже фактично бари і японські свічки дуже схожі в плані відображення інформації, але трохи по-іншому будуються на графіку.

Графік бар являє собою вертикальну лінію, яка показує інформацію про зміну ціни протягом певного періоду часу (залежить від таймфрейма). На графіку бари відображаються послідовно, незалежно від того, як змінюється ціна (рисунок 2.8).

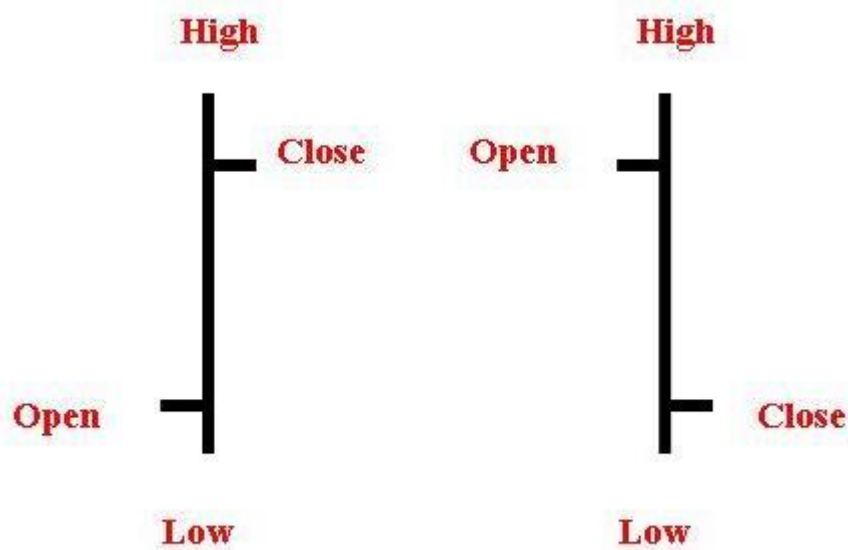


Рис. 2.8. Схеми побудови бару

У бар, як і у японських свічок, є 4 важливих показника цінового руху:

- ціна відкриття (Open) – значення ціни в самому початку тимчасового періоду. Відображається у вигляді невеликої горизонтальної риски зліва;
- ціна закриття (Close) – значення ціни в самому кінці тимчасового періоду. Відображається у вигляді невеликої горизонтальної риски праворуч;
- верхня межа (High) – максимальне значення ціни, яке було досягнуто протягом тимчасового періоду;
- нижня межа (Low) – мінімальне значення ціни, яке було досягнуто протягом тимчасового періоду.

Основною відмінністю є зорове сприйняття висхідних і спадних інтервалів. Японські свічки дозволяють при відносно малому масштабі визначити тенденцію інтервалу через те, що вони відрізняються за кольором. У графіку барів необхідно вдивлятися в відрізки, які показують ціни відкриття і закриття. Це не дуже зручно при перегляді графіка на малих масштабах, де

деталізація кожного окремого елемента мала. Нижче представлений графік барів, який наочно це демонструє (рисунок 2.9).

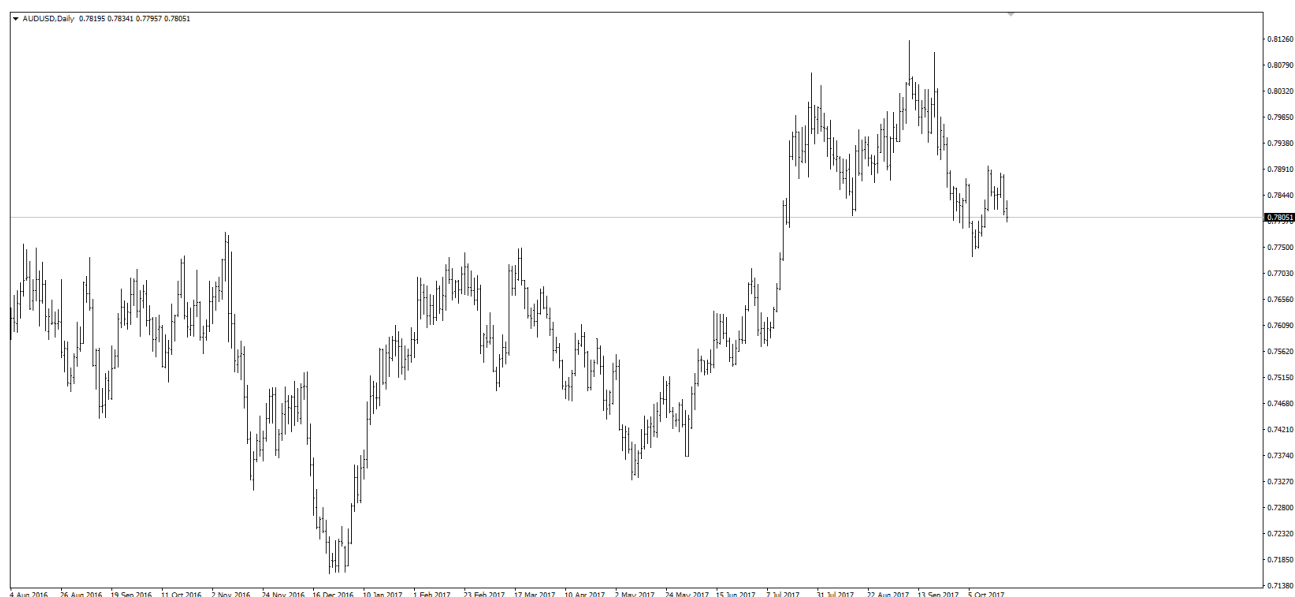


Рис. 2.9. Денний графік барів AUD/USD

З малюнка видно, що бари здаються куди більш розмитими. Саме тому в більшості випадків використовують японські свічки.

Далі, при описі алгоритму роботи автоматизованої торгової системи, всі приклади будуть представлені у вигляді графіка японських свічок.

2. 4. 2. Графічний аналіз

Найчастіше, для більш точного прогнозування руху ціни, на чистий графік наносять додаткові графічні об'єкти і використовують різні інструменти. Уміння вдало використовувати їх безпосередньо впливає на успішність торгової системи, яку використовує трейдер.

Оскільки ринок не підкоряється абсолютним закономірностям, то більшість сигналів, які з'являються завдяки використанню інструментів графічного аналізу, мають дві і більше інтерпретації подальшого руху ціни. Це ускладнює процес формалізації використання даних інструментів при автоматизації торгівельної діяльності, так як автоматизована система повинна

бути здатною з максимально можливою вірогідністю правильно інтерпретувати поточну ситуацію на ринку і грамотно на неї реагувати.

Далі будуть більш детально описані інструменти графічного аналізу, які використовуються в торгівельній системі, а також на докладних прикладах показана проблема подвійної інтерпретації торгівельних сигналів.

2. 4. 2. 1. Рівні підтримки та опору

Це основний інструмент графічного аналізу, який є найважливішим компонентом ринкової психології. І перш ніж перейти до розгляду правил побудови ліній підтримки та опору, цінових рівнів, які вони утворюють, та правил проторгування даних рівнів, необхідно уточнити фундаментальні причини їх появи.

Механізм їх утворення ґрунтується на силі попиту та пропозиції. Ринкова ціна рухається під впливом учасників ринку – і як свідчить теорія Доу, ринок містить у собі всю доступну інформацію.

Ціни падають, коли більшість учасників ринку вважає, що вони занадто високі. І зростають, коли вони, на думку більшості, надмірно низькі.

Рівні попиту і пропозиції – це той компонент аналізу, який лежить в основі психології ринку. Від уміння визначати їх силу і слабкість залежить успіх торгової діяльності. Рівні формуються завдяки визначенню важливих цінових значень. Оскільки це суто суб'єктивний процес, безліч гравців ринку визначають різноманітні цінові значення в межах одного рівня. Об'єднуючи їх, формуються важливі ринкові зони відкритого інтересу. Основна перевага зон зводиться до того, що вони можуть існувати як хвилини, дні, місяці, так і роки.

Побудова ліній підтримки та опору, які характеризують деякий ціновий рівень, і визначення (на основі цих ліній) важливих ринкових зон є базовою концепцією технічного аналізу. Проте вона має велику кількість торгових інтерпретацій.

2. 4. 2. 2. Механіка попиту та пропозиції

Пропозиція – це доступність певного активу в окремо взятій проміжок часу. У міру того, як зростає пропозиція певної валюти, вона знецінюється. Якщо на ринку занадто багато продавців, вони змушені скидати ціни, щоб хтось купив їх торговий інструмент.

По інший бік економічного рівняння знаходиться попит – це те, наскільки актив популярний. Попит на валюту прямо протилежний пропозиції за своїм ефектом. Чим більше попит на валюту, тим вона дорожча, чим менше – тим дешевше. Якщо на ринку з'являється попит на який-небудь інструмент через ряд різних зовнішніх факторів, то продавці цього інструменту можуть підвищувати ціну за своїм бажанням.

Як попит, так і пропозицію можна представити у вигляді двох діагональних ліній. При використанні котирувань (як показано нижче на малюнку 2.10) ідеальна рівновага між продавцями і покупцями по EUR/USD буде на позначці ціни 1.21.

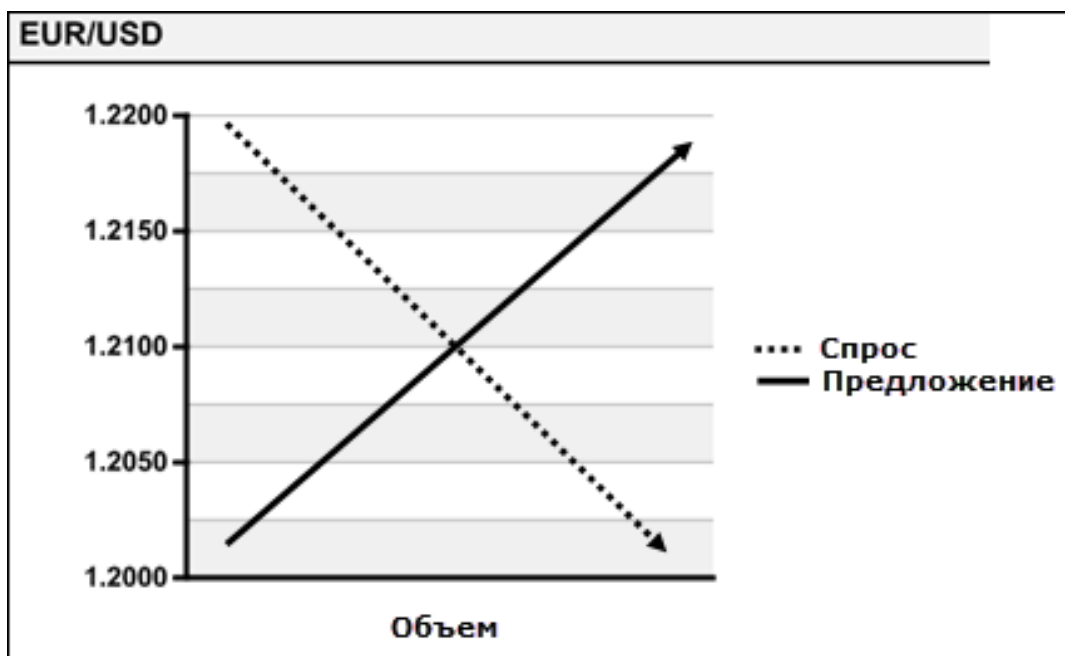


Рис. 2.10. Точка балансу попиту і пропозиції

Уявимо, що пропозиція зростає, значить "ідеальна" ціна – точка між покупцями і продавцями – зміщується нижче і правіше, переміщаючись прямо

по лінії попиту. Перетин двох ліній, відповідно, знижується за шкалою Y . В результаті, "ідеальний" рівноважний обмінний курс стає все менше (рисунок 2.11).

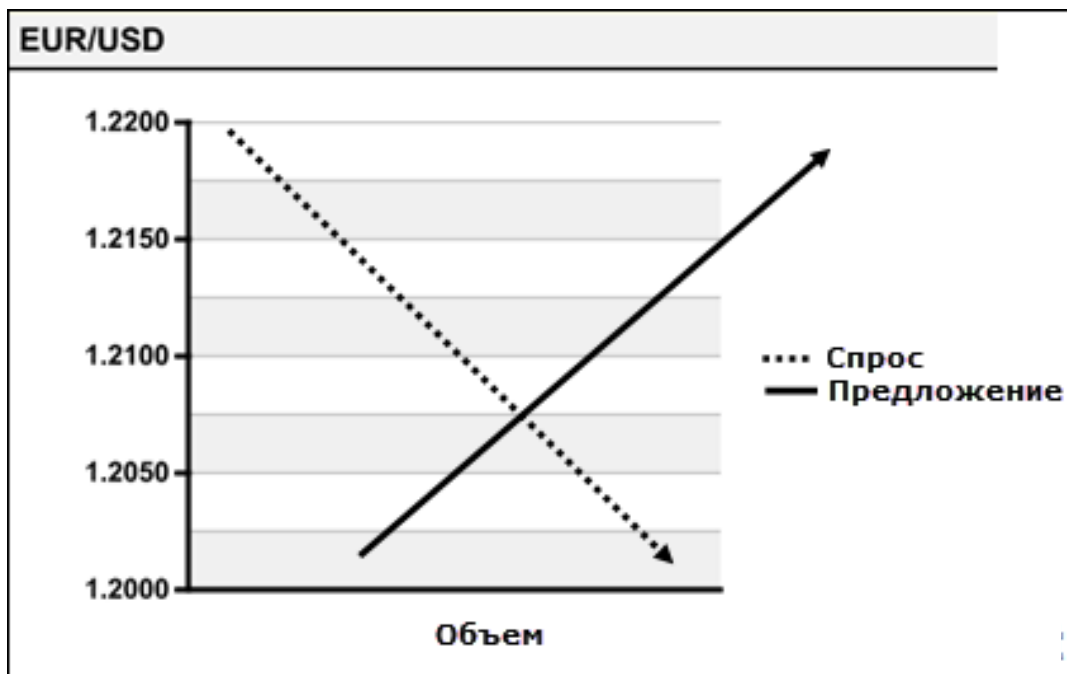


Рис. 2.11. Нова точка балансу при збільшенні пропозиції

І навпаки, зі зменшенням пропозиції ситуація змінюється кардинально, а лінія пропозиції зміщуються вліво і вгору. В результаті, "ідеальний" обмінний курс стає все вище (малюнок 2.12).

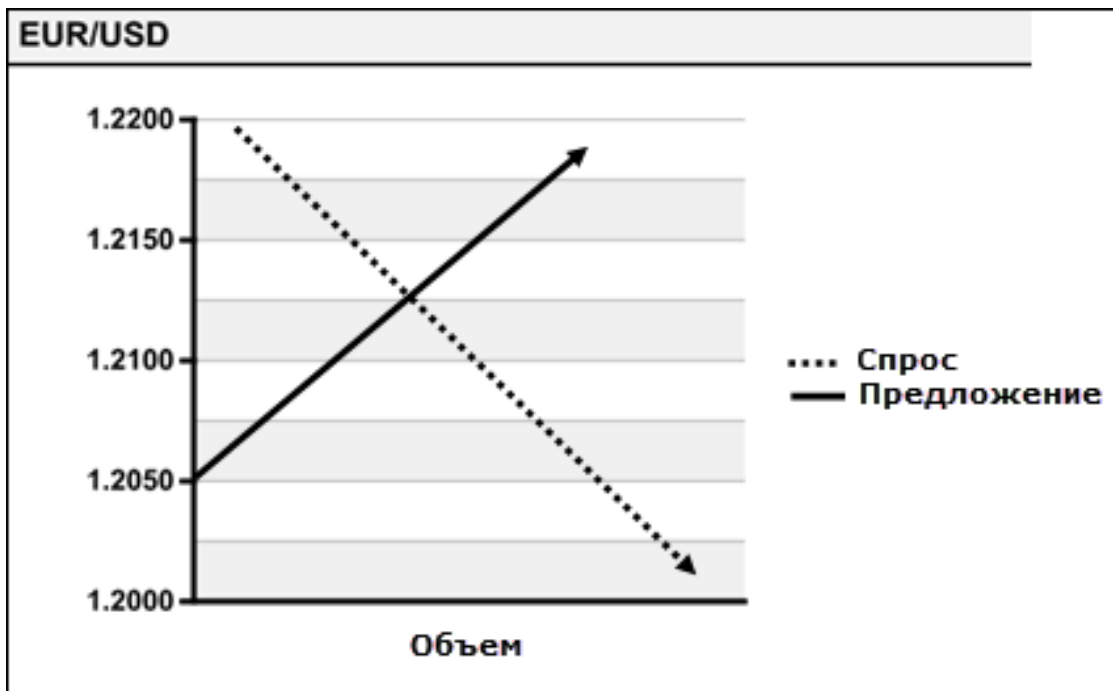


Рис. 2.12. Нова точка балансу при зниженні пропозиції

На прикладах вище були показані зміни під час маніпуляцій з лінією пропозиції. При аналогічних маніпуляціях з лінією попиту, результат буде відповідний. У разі збільшення потреби в певному активі (лінія попиту зсувається вгору та вбік уздовж лінії пропозиції), зростає і "ідеальна" ціна (малюнок 2.13). Справедливою буде й зворотна ситуація, коли попит зменшується. Тоді перетин зміститься вниз і вліво.

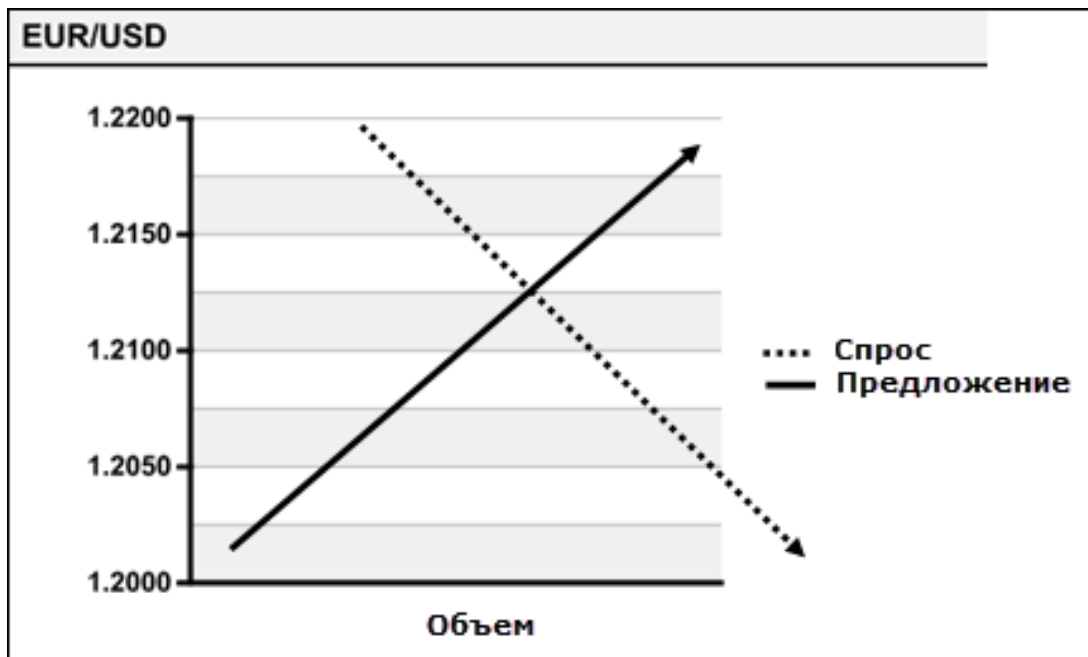


Рис. 2.13. Точка балансу при збільшенні попиту

Обмеження на рух капіталу були зняті в більшості розвинених країн світу. Тому тепер саме глобальний ринок вирішує, якою буде курс валют тієї чи іншої країни в залежності від чистого балансу потреби та пропозиції. Звідси три важливих наслідки:

1. Рух ціни на вільному ринку обумовлено попитом і пропозицією.
2. В ціну вже закладені всі фактори, що на неї впливають.
3. Ціна – це рівняння, де вплив двох сил (покупців і продавців) прагне до нуля.

2. 4. 2. 3. Рівень підтримки

Рівень підтримки (він же рівень попиту) завжди знаходиться внизу (рисунок 2.14). Коли попит сильніше пропозиції, то ціна рухається вгору і не в змозі пробити певний ціновий рівень. Саме тут працюють покупці.



Рис. 2.14. Приклад утворення лінії підтримки

На рівні підтримки учасники ринку очікують, що попит перевищить пропозицію, тому падіння ціни сповільнюється у міру того як ціна приближується до цього рівня. Потім очікується і відскік ціни вгору, оскільки підтримка – це ринковий рівень, на якому попит досить сильний, щоб утримувати ціну в зоні підтримки.

Психологічний аспект досить простий. Він полягає в тому, що ціна на торговий інструмент надто низька та ринок вважає, що варто його купувати. Ціна доходить до рівня і починаються хвилі покупок. Цілком логічно, що в міру того, як ціна падає до лінії підтримки і стає менше, покупцям все більш цікаво купити цей знецінений актив. Починає зростати попит, за ним зростає і ціна.

2. 4. 2. 4. Рівень опору

Рівень опору являє собою зону, де пропозиція надмірно перевищила попит, тому зростання ціни зупиняється (рисунок 2.15). Саме тут починають працювати продавці, які володіють переоціненим, на їх думку, активом.



Рис. 2.15. Утворення лінії опору

Якщо вони вважають ціну достатньо високою, то починаються масові продажі. Іншими словами, на рівні опору тиск продажу вище, ніж попит. Іноді цей тиск настільки сильний, що ціна змінює свій курс.

2. 4. 2. 5. Торгові інтерпретації

На наведеному нижче прикладі візуалізуємо підтримку і опір на свічковому графіку EUR/USD (рисунок 2.16). У міру того, як ціна відходить від лінії опору 0.9080, попит вичерпується і починається сильний спадний тренд. На рівні 0.9080, де розташована лінія опору, знаходиться найвища концентрація продавців, а найменша, відповідно, на рівні 0.8750.



Рис. 2.16. Відпрацювання лінії опору

Якщо розглянути цей графік з точки зору підтримки/попиту (рисунок 2.17), то видно, що у деякий проміжок часу ціна піднялася від рівня 0.8750, оскільки попит був максимальним.



Рис. 2.17. Відпрацювання лінії підтримки

Коли потреба на певну валюту досить висока, швидкість зниження ціни сповільниться, що і сталося на рівні 0.8750. Однак, як видно з малюнка, є друга торгова інтерпретація – пробій лінії підтримки або опору (в даному випадку підтримки). Саме про це йшла мова раніше, коли була озвучена проблема подвійної інтерпретації. Чому це відбувається?

Попит і пропозиція існують на ринку постійно. Якщо розглянути попередній приклад, то ближче до середини графіка волатильність ціни зменшується. На тому рівні, де попит і пропозиція приблизно рівні, починається боковий рух – консолідація (рисунок 2.18).

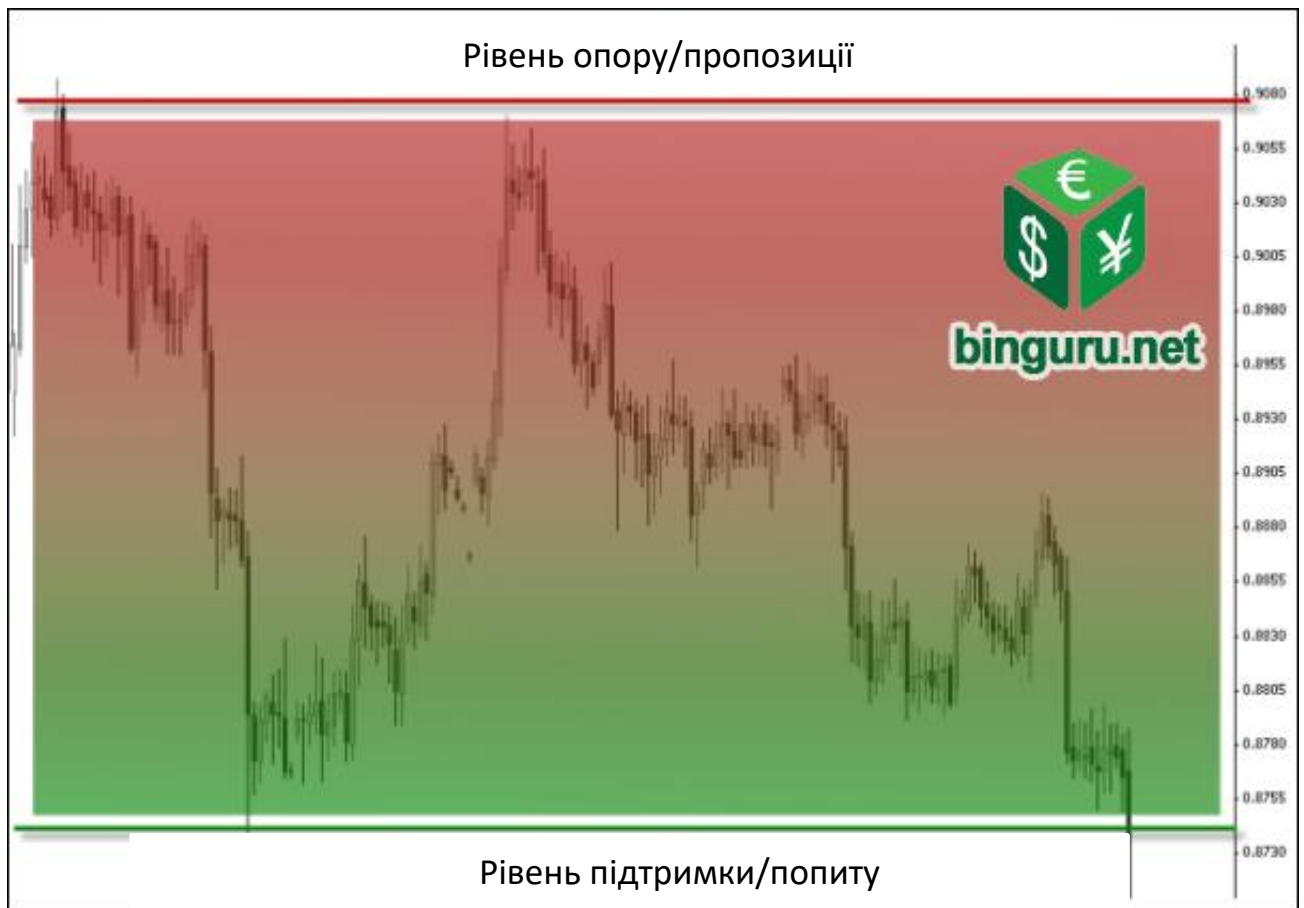


Рис. 2.18. Приклад утворення зони консолідації

На рівні підтримки або опору критична маса трейдерів (або капіталу) має намір продавати або купувати торговий інструмент. Як визначалося раніше, рівні не є однозначними лініями на графіку. Вони формують ринкові зони, які будуються навколо цих ліній з умовним діапазоном. Це відбувається під впливом суб'єктивної оцінки риночної ситуації кожним окремим трейдером або фінансовим інститутом, тому в рамках однієї ринкової зони кожен може визначити свої цінові лінії. Ці зони постфактум ґрунтуються на поведінці учасників ринку в минулому.

В результаті, ціна "відскакує" від зони декілька раз, поки ситуація не зміниться. Інакше, ринок вічно знаходився б у стані консолідації. На динаміку ціни впливає безліч факторів, які постійно змінюються. Тому те, що вважалося хорошою ціною для покупки (рівень підтримки) неминуче зміниться після виходу важливих економічних даних і ціна проб'є цей рівень.

Графік, який наведено нижче, це наочно ілюструє (рисунок 2.19). На ньому зображено, як рівень підтримки не витримав, а його пробій підтверджує той факт, що зазначений ціновий рівень (зелена лінія) більше не є цікавим для покупців.



Рис. 2.19. Приклад пробою лінії підтримки

Цей приклад показує, що будь-яка лінія підтримки та опору може проторговуватися двома варіантами: сигнали на відбій та на пробій рівня. Якщо трейдер прогнозує відбій, то він купить актив при наближенні ціни до даного рівня. Якщо трейдер прогнозує пробій, то на цьому рівні він буде продавати актив.

Але проблема полягає в тому, що відсутня можливість дізнатися, як відпрацює даний сигнал в момент його утворення. Існує третя торгова інтерпретація – випадки, коли ціна пробиває рівень, а потім різко повертається назад за цей рівень. Такі пробої називаються помилковими і сигналізують про силу рівня.

Ці зони часто стають трампліном для різкого розвороту ціни. Психологія помилкових пробоїв зрозуміла – зайвий оптимізм покупців або продавців, які прогнозували подальший рух ціни в напрямку пробитого рівня в момент утворення сигналу (рисунок 2.20).

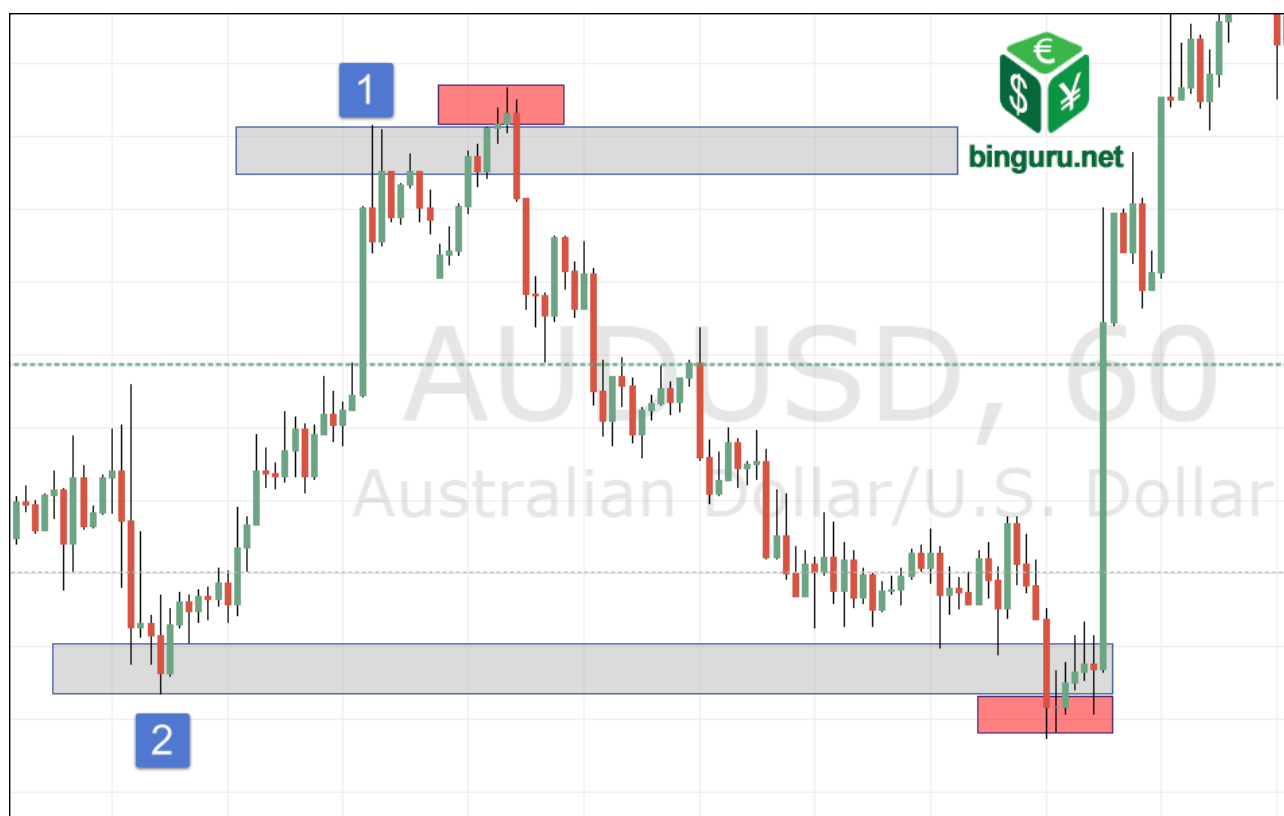


Рис. 2.20. Приклад фальшивого пробою

У висновку необхідно підвести підсумок, чому було приділено стільки уваги лініям підтримки і опору. Даний інструмент лежить в основі технічного аналізу, на підставі якого працює велика кількість різних торгових механік і інших, більш складних видів аналізу. Тому виникає потреба, щоб торговий робот, використовуючи історичні дані за певний період і на певному тайм-

фреймі, мав змогу грамотно визначати сильні рівні і проторговувати їх в описаних вище ситуаціях (відскік, пробій і помилковий винос).

2. 4. 2. 6. Динамічні рівні підтримки та опору

Більшу частину часу ринок перебуває в стані консолідації. У такій ситуації ціна рухається від одного горизонтального рівня до іншого. Але різні події в світі провокують односпрямований рух ціни – тренд. Як наслідок, з'являється новий інструмент графічного аналізу, а саме лінії тренда – один із видів динамічних рівнів підтримки та опору.

На відміну від горизонтальних ліній, які будуються навколо визначеного трейдером цінового значення, динамічні характеризують спрямований покроковий рух котирувань під впливом певного настрою учасників торгів.

Динамічна лінія формується, як горизонтальна, на основі двох точок, де початкова точка повинна являти собою явний рівень підтримки або опору. Перший відкат у рамках тренду формує другу точку і генерується лінія. Таким чином, тренд – це оновлення локальних максимумів (висхідний рух) або мінімумів (спадний рух).

На тижневому графіку (рисунок 2.21) дві динамічних лінії тренду виступають в якості опору.

Варто відзначити, що в торговельному алгоритмі системи робота, пошук і побудова ліній тренда не передбачена, проте залишається необхідність визначати трендові рухи.

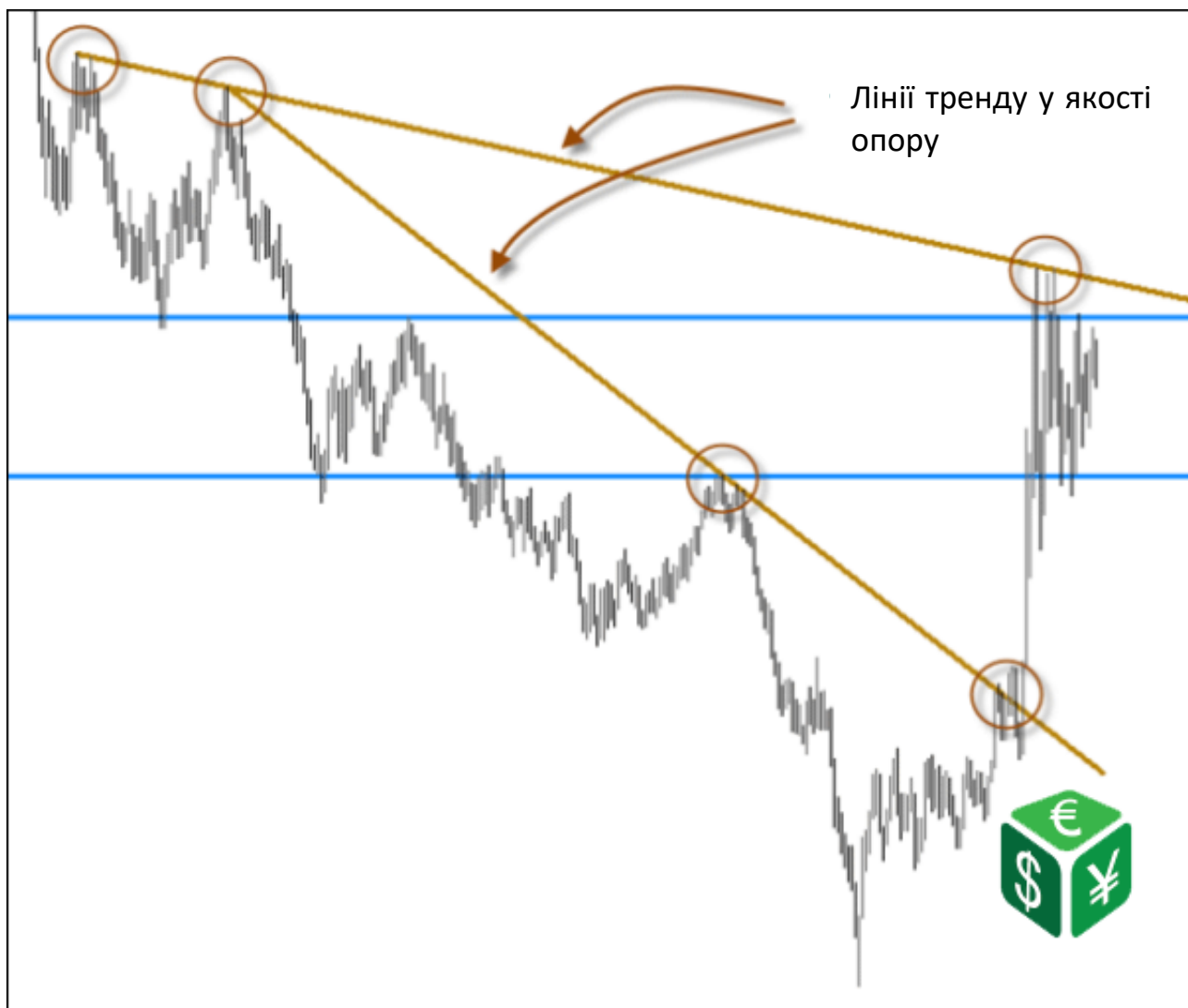


Рис. 2.21. Відпрацювання і пробій ліній тренда

Іншим видом динамічних рівнів є ковзні середні. Використання ковзних середніх ліній є частиною індикаторного аналізу, однак, через контекст їх основного застосування, цей інструмент буде описаний далі.

Ковзні середні (далі – МА) використовуються для визначення поточного тренда і пошуку точки входу в його напрямку. Як і у випадку з горизонтальними лініями, при взаємодії ціни з ковзними лініями існує три можливі торгові інтерпретації – відскік, пробій та помилковий пробій. Крім того, нанесення на графік декількох МА дозволяє визначити моментум, флет та зміну тренда.

Моментум – це темп зміни руху ціни. Найпростіший спосіб визначити моментум – це нанести на графік три ковзні середні лінії різного періоду і

спостерігати, як вони співвідносяться одна з одною. При сильному моментумі лінії розправляються одна за одною, за зростанням (рисунок 2.22).



Рис. 2.22. Варіанти використання різних МА

Поєднання трьох МА різних періодів допомагає визначити моментум ціни, динамічні рівні підтримки і опору (на малюнку добре видно, як МА 100 відпрацьовує в якості підтримки) і зміну тренда (праворуч всі МА перетинаються, що вказує на початок спадної тенденції).

При консолідації – лінії будуть перетинатися впродовж деякого часу, а ціна постійно входить з ними в контакт. Зазначені фактори вказують на відсутність сильної тенденції (рисунок 2.23).



Рис. 2.23. МА підтверджують, що ринок знаходиться в стані флету

Існує безліч різновидів ковзних середніх, але всі вони зводяться до однієї концепції. Це простий і наочний спосіб отримати усереднений рух ціни за певний проміжок часу.

Середні ковзаючі лінії бувають трьох видів:

- прості (англ. Simple moving average, SMA);
- експоненціальні (англ. Exponential moving average, EMA);
- зважені (англ. Weighted moving average, WMA).

Просте середнє ковзне періоду n на момент k – це середня арифметична величина n значень від $k-n + 1$ до k . Іншими словами, 5-ти денне середнє ковзне на поточний момент часу вираховується шляхом додавання п'яти попередніх цін (поточна плюс чотири минулих) і поділом їх на 5.

При розрахунку експоненціального середнього вважається, що пізні дані більш важливі. Отже, цей вид середнього ковзного швидше реагує на поточні зміни ціни.

Розрахунок значення експоненціального середнього ковзного складніший: обчислення значення 5-ти денного експоненціального середнього ковзного на кожний момент часу проводиться за формулою:

$$EMA[k, n] = EMA[k - 1, n] + \frac{2}{n+1} (P - EMA[k - 1, n]), \quad (2.1)$$

де $EMA[k, n]$ – експоненціальне ковзне середнє періоду n на момент k ;
 P – поточна ціна.

Зважене середнє ковзне розраховується, як і проста МА, з тією різницею, що кожену ціну, яка входить в прорахунок зваженого ковзного середнього, необхідно помножити на її порядковий номер, а потім розділити всю цю суму на суму порядкових номерів.

$$WMA = \frac{\sum_{i=1}^n P_i * W_i}{\sum_{i=1}^n W_i} \quad (2.2)$$

де WMA – зважене ковзне середнє;
 P_i – поточна ціна;
 W_i – значення ваги для ціни i -того періоду.

Різниця між трьома видами змінних середніх для аналітика, за великим рахунком, незначна. Проте, вона досить помітна (рисунок 2.24).



Рис. 2.24. Порівняння різних видів МА. SMA – червона, ЕМА – експоненціальна, WMA – зважена

В алгоритмі торгового робота використовується дві SMA з періодами 100 та 200. У разі відкриття позиції, МА використовуватися для:

1. визначення поточного стану ринку (зростаючий, спадний тренд або консолідація);
2. проторговки відбою від МА в напрямку тренда за умови, що поточна ціна і ціна закриття попереднього дня знаходиться в рамках тренда, який відповідає періоду, вказаному в налаштуваннях торгової системи;
3. проторговки невдалого пробою МА в напрямку тренда за умови, що поточна ціна знаходиться в рамках тренда, а попередня свічка є пін-баром.

2. 4. 3. Індикаторний аналіз

Одним з важливих методів технічного аналізу на ринку FOREX є індикаторний аналіз. У торговому роботі закладено низку стандартних індикаторів різних типів, які є інтерпретацією руху ціни з різних сторін.

Індикатор – це невелика програма, в яку закладено певний алгоритм поведінки ціни. Використання індикаторів звільняє трейдера від необхідності проводити складні розрахунки і графічні побудови. Все це за нього зробить програма. У торговому терміналі за замовчуванням є набір найбільш популярних індикаторів. Крім цього, є можливість використовувати як сторонні індикатори, так і власні.

Індикаторний аналіз – це найбільш швидкий спосіб оцінки ситуації на ринку, однак він вимагає певного практичного досвіду фільтрації помилкових сигналів з обов'язковим попереднім тестуванням роботи того чи іншого індикатора на демо рахунку.

Індикатори, що найбільш часто використовуються в технічному аналізі, можна розділити на декілька груп: трендові, каналні індикатори і осцилятори.

Трендові індикатори виконують функцію стеження за трендом. Такий індикатор дозволяє візуально визначити напрямок руху ціни фінансового інструменту, а також своєчасно відстежити розворот тренда по перетинанню ціною лінії індикатора. Основний індикатор з даного сімейства – МА, був описаний раніше в розділі динамічних ліній підтримки і опору.

2. 4. 3. 1. Канальні індикатори

Канальними називаються індикатори, що відображають канал, в межах якого рухається ціна. У каналних індикаторах одна з ліній служить підтримкою, інша опором. Яскравим прикладом є канал Дончіана (рисунок 2.25). Сигналами для покупки або продажу може служити, як відбій від ліній каналу, так і їх пробій.

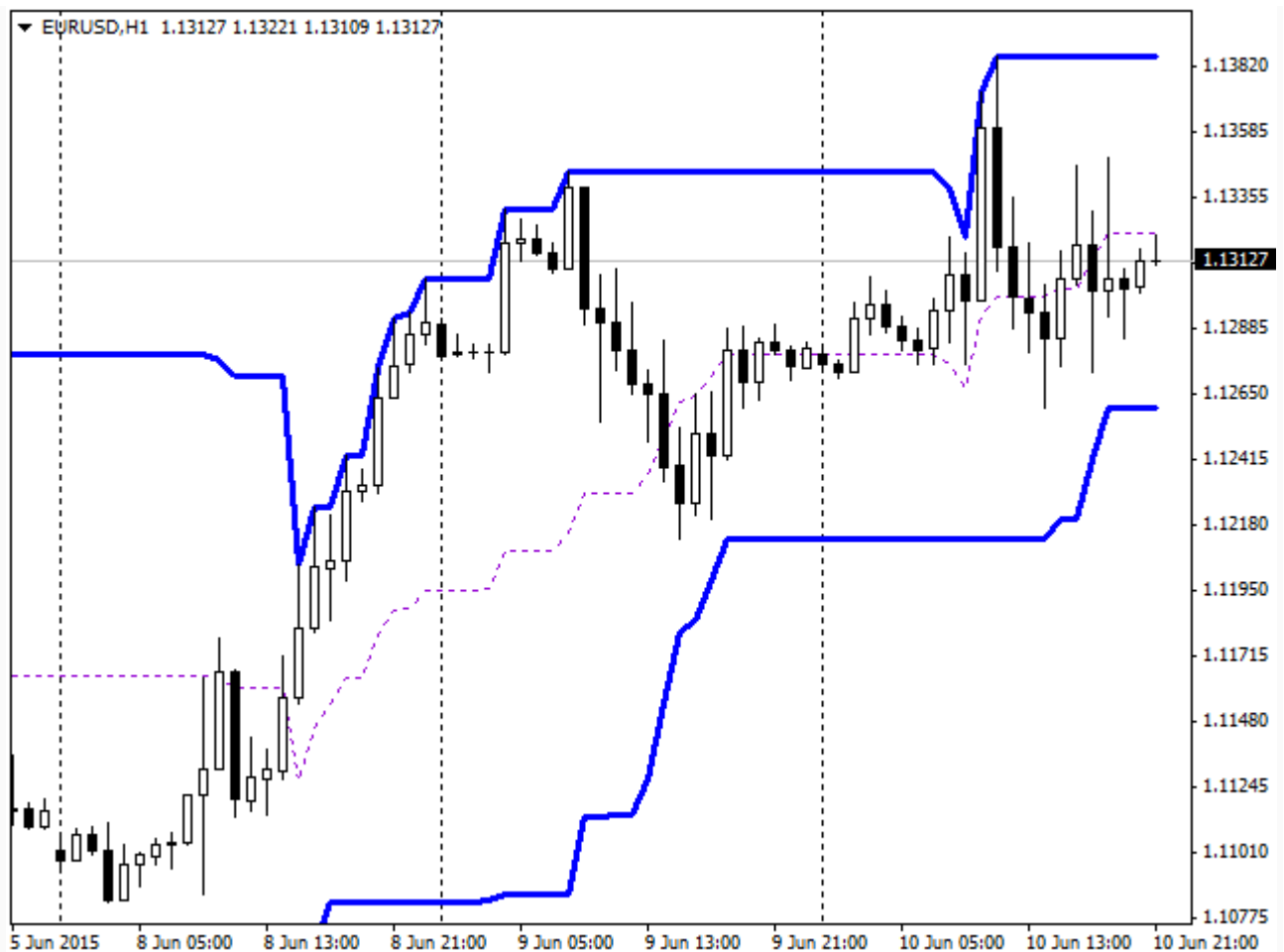


Рис. 2.25. Приклад каналного індикатора

В автоматизованій торговій системі, яка розроблена в цій дипломній роботі, каналні індикатори в чистому вигляді не використовуються, за винятком «Смуг Боллинджера», який відображає канал, однак каналним не є.

2. 4. 3. 2. Осцилятори

Осцилятори – це вид індикаторів, які за своїм призначенням визначають перекупність або перепроданість активу. Осцилятори є показником відхилення ціни від середнього її значення. За допомогою осциляторів можна передбачити наближення корекції або напрямку фази коливання ціни. Найкраще осцилятори підходять для прийняття рішень у відсутності виражених трендів.

Існує велика кількість осциляторів:

- Average True Range;

- Bears Power;
- Bulls Power;
- Chaikin Oscillator;
- Commodity Channel Index;
- DeMarker;
- Force Index;
- MACD;
- Momentum;
- Moving Average of Oscillator;
- Relative Strength Index;
- Relative Vigor Index;
- Stochastic Oscillator;
- Triple Exponential Average;
- Williams' Percent Range.

Далі будуть розглянуті деякі з них, а саме:

- MACD;
- Relative Strength Index;
- Stochastic Oscillator.

2.4.3.3. MACD

Moving average convergence/divergence – популярний осцилятор, розроблений Джеральдом Аппелем в 1979 році. Будується індикатор на основі таких компонентів (рисунок 2.26):

1. лінія MACD;
2. сигнальна лінія;
3. гістограма.

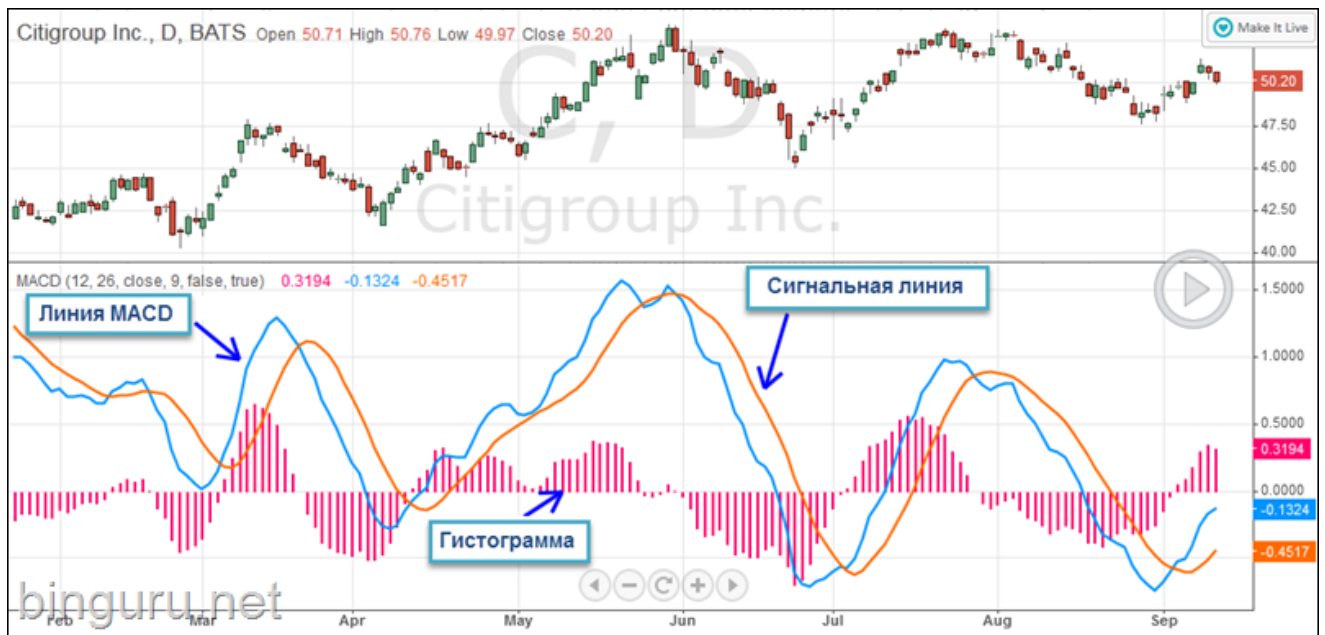


Рис. 2.26. Опис компонентів MACD

Лінія MACD являє собою довгу середню ковзну.

Сигнальна лінія – це ЕМА з деяким відхиленням від лінії MACD, значення якої за замовчуванням дорівнює 9.

Гістограма показує відстань між лінією MACD і сигнальною лінією. Її завдання визначити напрям та розходження ліній по відношенню одна до одної.

Використання MACD:

1. Перетин сигнальної лінії

Це найпопулярніший, актуальний і затребуваний сигнал індикатора MACD. Якщо лінія MACD перетинає сигнальну лінію – можна очікувати різкої зміни поточної тенденції ціноутворення. У прикладах лінія MACD виділена синім кольором, сигнальна лінія – червоним.

Отже при перетині ліній може виникнути дві ситуації:

- лінія MACD перетинає сигнальну лінію знизу вгору: прогнозуєма цінова динаміка може стати позитивною;
- лінія MACD перетинає сигнальну лінію зверху вниз: очікується рух ціни вниз.

Приклад на рисунку 2.27 наочно показує описані вище сигнали.

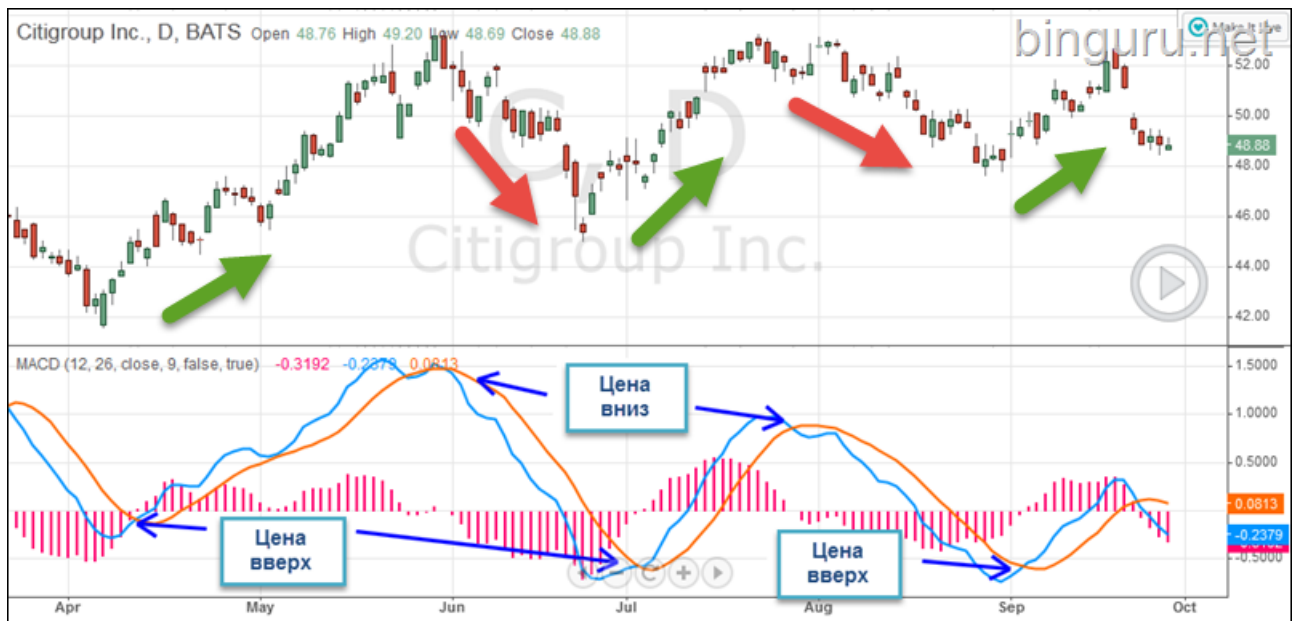


Рис. 2.27. Відпрацювання сигналів перетинання ліній індикатора MACD

2. Перетин нульової лінії

Другий можливий тип сигналів – це сигнали, коли лінія MACD перетинає нульову лінію і деякий час рухається вгору або вниз (рисунок 2.28). Як і в минулому випадку, може виникнути дві ситуації:

- якщо MACD перетинає нульову лінію знизу вгору – курс піде вгору;
- перетин нульової лінії зверху вниз – курс піде вниз.

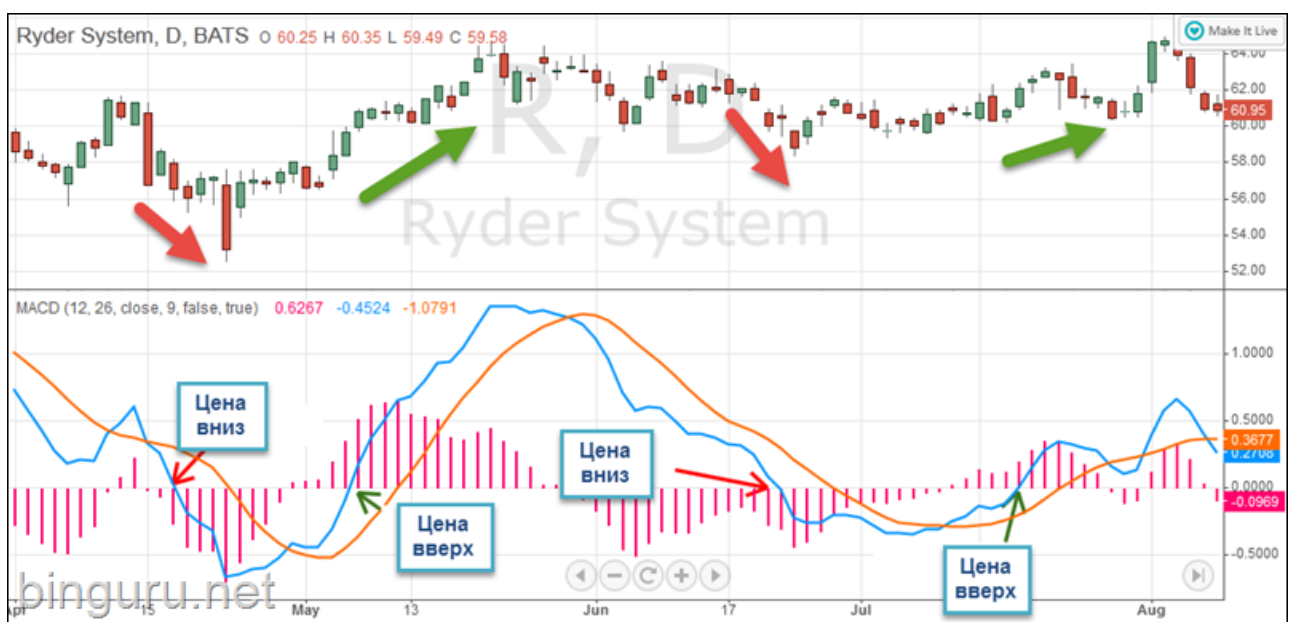


Рис. 2.28. Відпрацювання сигналів на перетин нульової лінії

3. Дивергенція

Дивергенція – це розбіжність показників індикатора і реального руху ціни, при виникненні якої перевага віддається показанням індикатора. В даному випадку: якщо лінії MACD йдуть вгору, а ціна рухається вниз, то трейдер може очікувати змінення напрямку руху ціни вгору. Нижче наведено два приклади позитивної і негативної дивергенції (рисунки 2.29-2.30).



Рис. 2.29. Позитивна дивергенція

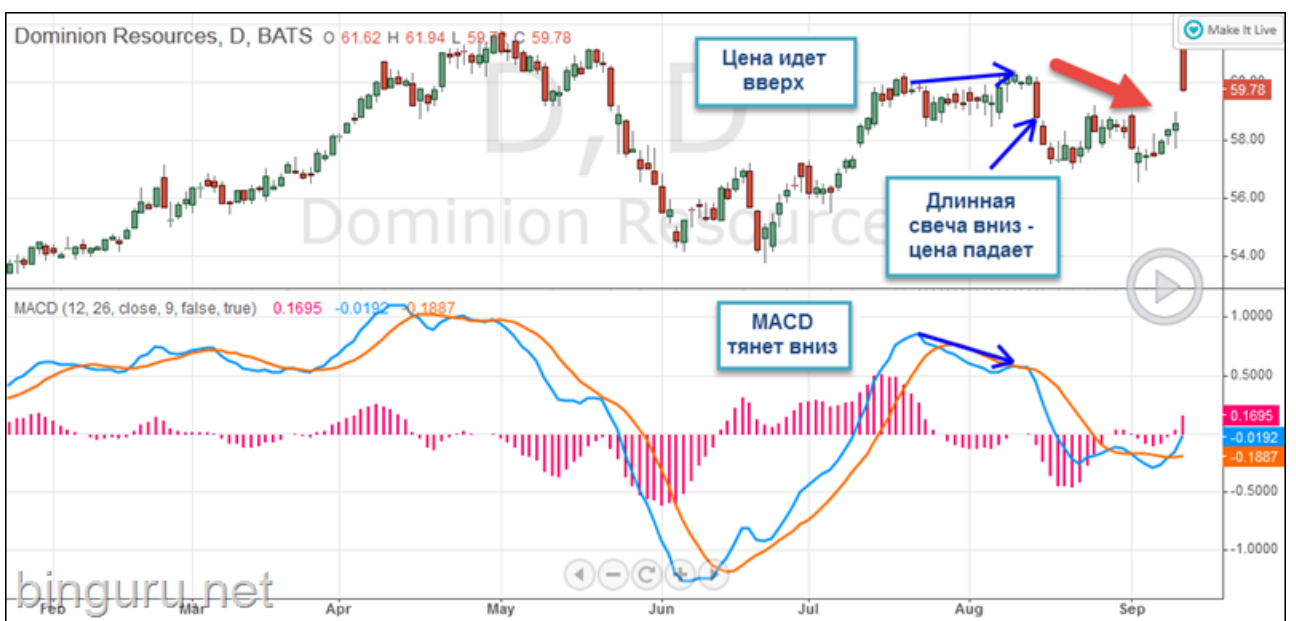


Рис. 2.30. Негативна дивергенція

2. 4. 3. 4. Stochastic

Індикатор Stochastic, розроблений економістом Джорджем Лейном, дозволяє нам виявити кілька ключових сигналів:

- надмірно низький або високий рівень ціни;
- дивергенції;
- сигнали на підйом або зниження ціни.

Нижче представлені скріншоти використання даного індикатора (рисунки 2.31-2.32).



Рис. 2.31. Використання Стохастика на живому графіку

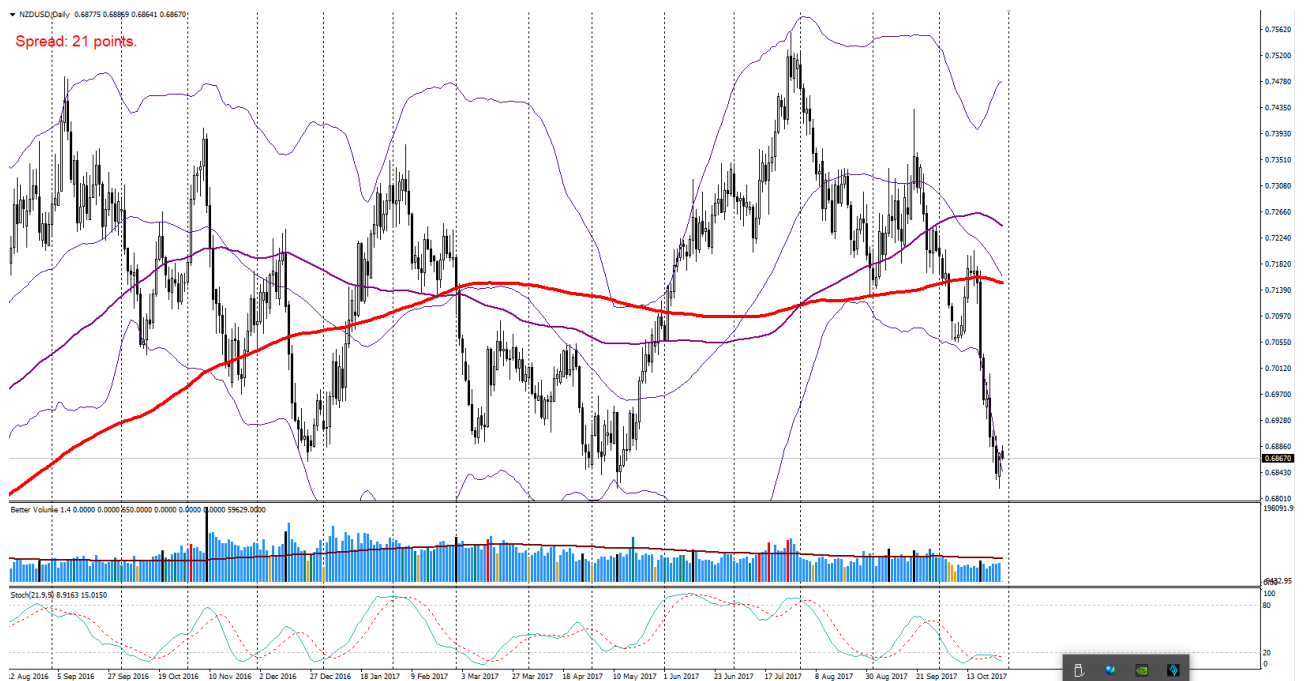


Рис. 2.32. Використання Стохастика всередині терміналу МТ4

Стохастичний осцилятор складається з двох ліній, які відображаються в шкалі від 0 до 100.

Перша лінія К відображає поточну ціну закриття по відношенню до вибраного трейдером діапазону ціни.

Друга лінія D – це звичайна проста ковзна середня, що обчислюється на основі лінії К.

Сигнали, які видає цей індикатор:

- Попадання ліній індикатора в зону перекупності, перетин і вихід з неї – сигнал на продаж.
- Попадання ліній індикатора в зону перепроданості, перетин і вихід з неї – сигнал на покупку.
- Дивергенція.

Всі три типи сигналів будуть розглянуті нижче на прикладі індикатора RSI.

2.4.3.5. RSI

Індекс відносної сили RSI (Relative Strength Index) – один з ключових осциляторів, який використовується у всьому світі.

Цей індикатор, як і МА, відображає моментум – тобто швидкість і амплітуду, з якою змінюється рух ціни (рисунок 2.33).

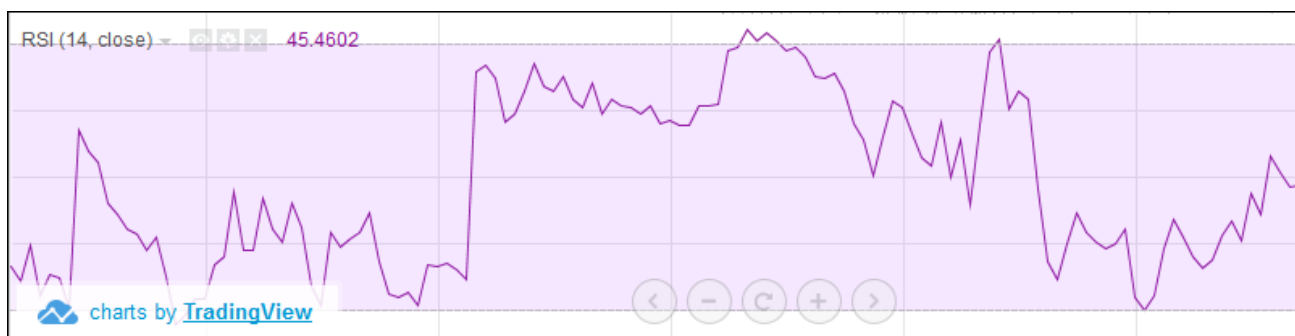


Рис. 2.33. RSI с періодом 14

Іншими словами, за допомогою RSI можна аналізувати силу або ж слабкість поточного ринкового стану ціни. Індикатор заснований на використанні цін закриття за певний проміжок часу.

Формула RSI проста:

$$RSI = 100 - 100 / (1 + RS) \quad (2.2)$$

де RS – середній приріст поділений на середнє падіння.

Як і більшість осциляторів, RSI відображається на шкалі в діапазоні від 0 до 100 (рисунок 2.34).

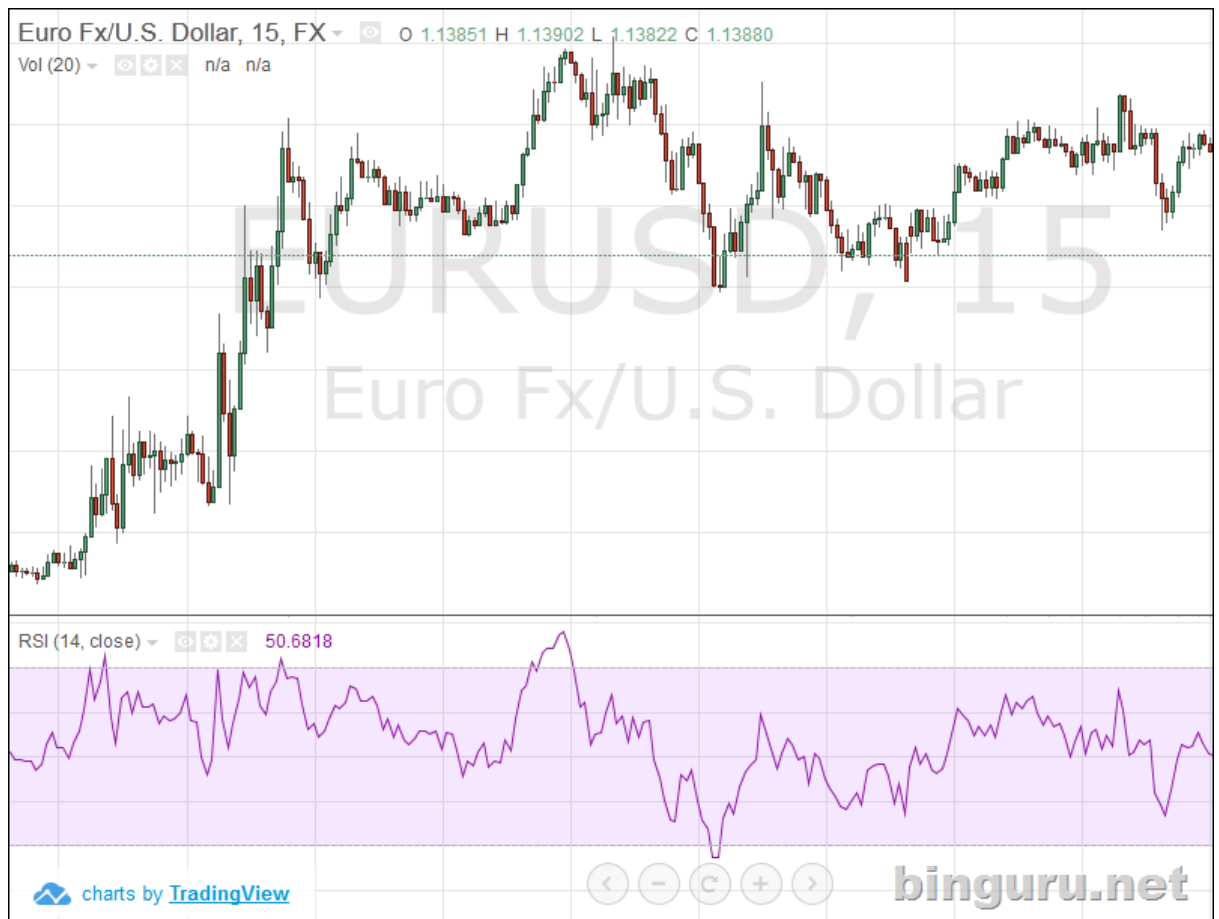


Рисунок 2.34. Залежність величини моментуму та амплітуди ціни від діапазону показань RSI

Чим ближче RSI до 0 – тим більше уповільнюється рух ціни вниз.

Чим ближче RSI до 100 – тим активніше сповільнюється рух ціни вгору.

За замовчуванням, індикатор враховує ціни за 14 днів. Це значення можна змінити в налаштуваннях автоматизованої торгової системи.

Використовувати RSI можна кількома способами, які є загальними для багатьох осциляторів:

- перекупленість;
- перепроданість;
- дивергенція.

Перекупність та перепроданість визначається при перетині лінією індикатора рівнів, які визначені в його налаштуваннях. Наприклад:

- RSI більше 70 – перекупленість;

- RSI менше 30 – перепроданість.

На скріншоті наведена ситуація, де RSI більше 70, тобто він знаходився в зоні перекупленості, що є сигналом на розворот тенденції (рисунок 2.35).

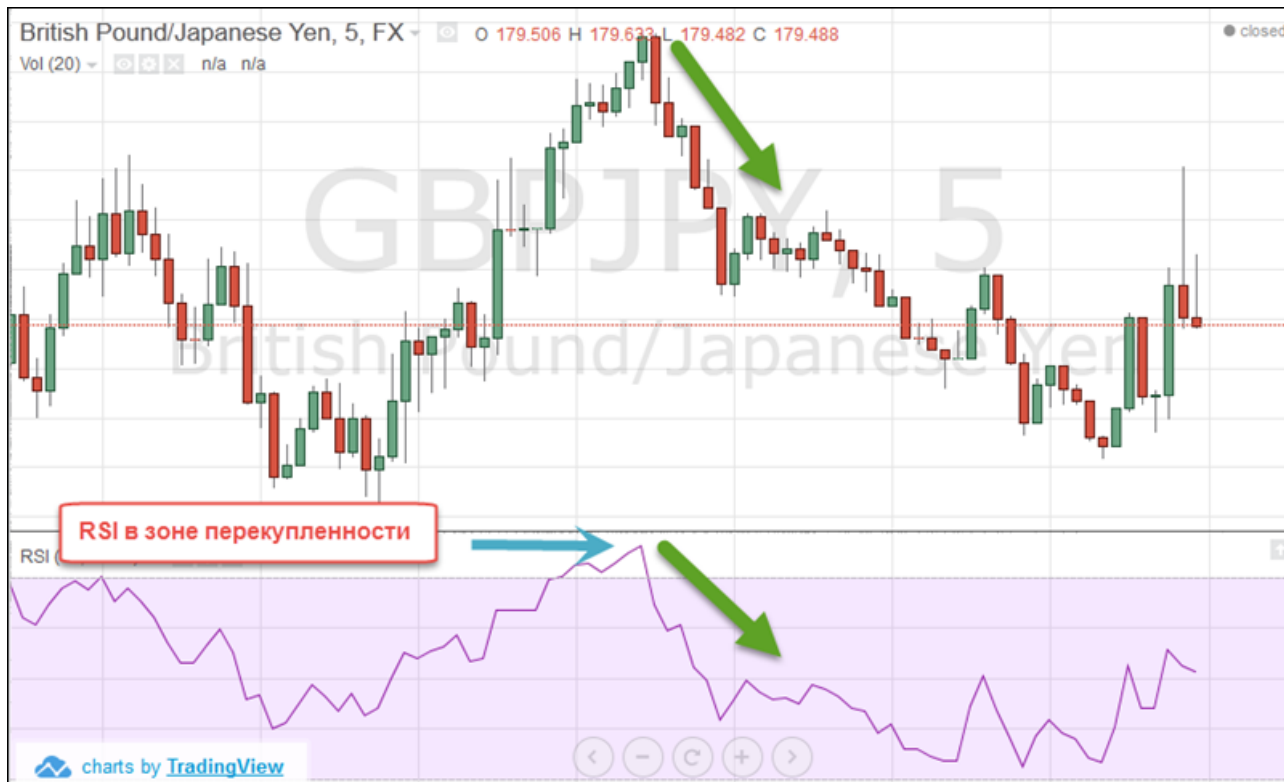


Рис. 2.35. Відпрацювання сигналу перекупленості

Зворотна ситуація. RSI в зоні перепроданості (рисунок 2.36).

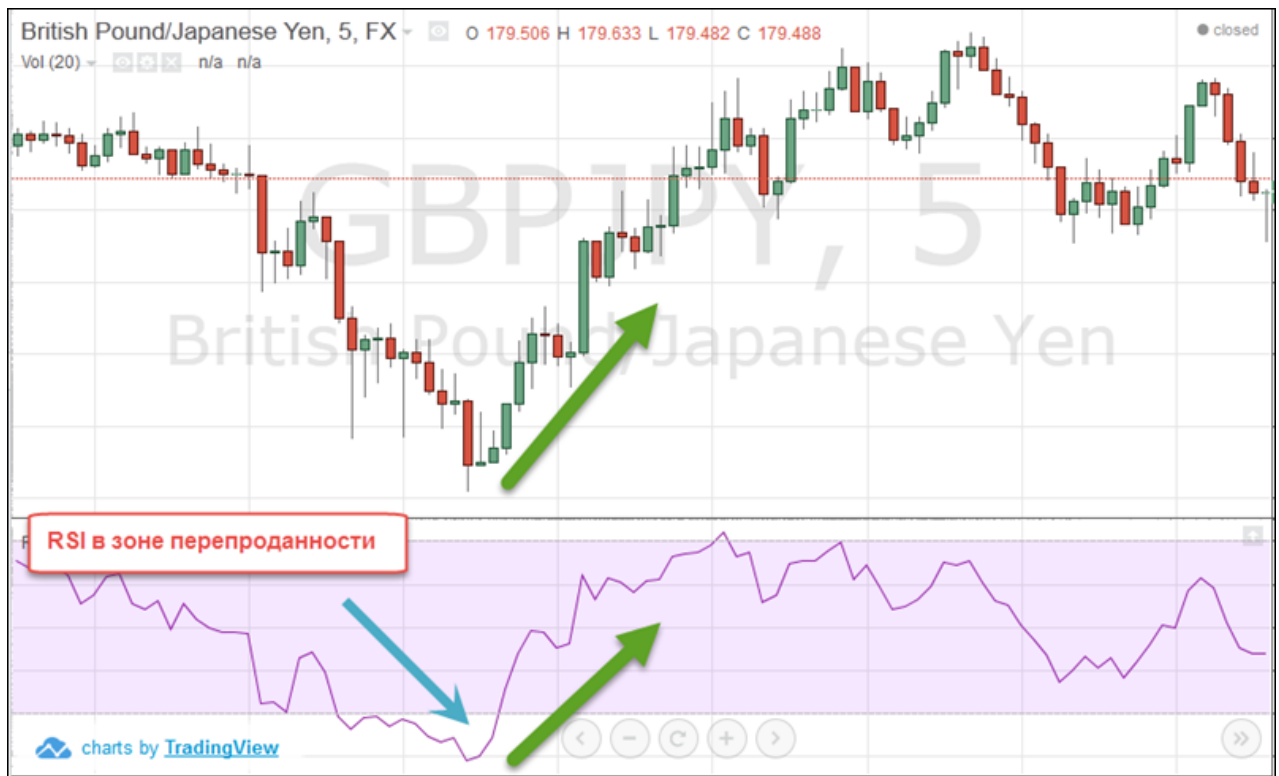


Рис. 2.36. Відпрацювання сигналу перепроданості

У разі знаходження RSI між рівнями 70 і 30 – це свідчить про нейтральну ситуацію, а біля значення 50 – відсутність тренда (рисунок 2.37).



Рис. 2.37. RSI вказує на консолідацію

Як і у будь-якого іншого осцилятора, однозначно працюючий сигнал – це дивергенція (рисунок 2.38).

Позитивна дивергенція – ціна вниз, RSI вгору.

Негативна дивергенція – ціна вгору, RSI вниз.



Рис. 2.38. Негативна дивергенція

Особливим сигналом RSI є внутрішній розворот, який буває позитивним або негативним, кожний з яких складається з 4 кроків.

Позитивний розворот (на зростання ціни):

- RSI падає нижче 30 (перепроданність);
- підскакує вище 30;
- відскакує назад, але все ще вище 30;
- прорив вгору – зміна тренда відбулася.

Негативний розворот (на падіння ціни):

- RSI піднімається вище 70 (перекупність);
- падає нижче 70;
- піднімається знову, але все ще нижче 70;
- прорив вниз – зміна тренда.

Приклад негативного розвороту приведено на рисунку 2.39.



Рис. 2.39. Негативный разворот

Окремо варто виділити такі випадки дивергенції, а саме:

1. Дивергенція, що підтверджує тренд

Якщо тренд дійсно сильний і стійкий, то дивергенція може його лише підтвердити. Про це писав Ендрю Кардвелл (Andrew Cardwell).

За його спостереженнями, в сильному тренді дивергенція RSI лише підтверджує цей тренд, а зовсім не спростовує його. На графіку нижче представлений цей випадок (рисунок 2.40).



Рисунок 2.40. Дивергенція, що підтверджує тренд

Така ситуація показує, чому потрібно використовувати всі можливості технічного аналізу для визначення трендів. У сильному тренді більшість індикаторів будуть лише його підтверджувати, а не вказувати на розворот.

2. Зворотня дивергенція

Раніше була представлена класична дивергенція. Вона полягає в тому, що при розбіжності показань індикатора і основного графіка ціни, правильним вважаються показання індикатора, які вказують на швидку зміну тенденції. Проте, Кардвелл описував і, так звані, зворотні дивергенції. У них все навпаки:

- ціна вгору, RSI вниз – очікується сильний рух вгору;
- ціна вниз, RSI вгору – очікується сильний рух вниз.

Нижче наведено приклад випадку зворотної дивергенції (рисунок 2.45).



Рисунок 2.41. Зворотній дивергенція

Підводячи підсумок, необхідно зазначити, що з усіх видів дивергенції, реалізований був лише пошук класичних, який, на жаль, поки що значно погіршує результати роботи торгової системи. Багато в чому це пов'язано зі складністю точно визначати максимуми і мінімуми на графіку ціни і осцилятора, отже неможливо точно визначити дивергенцію. Тому при використанні осциляторів, сигнали дивергенції відключені.

2. 4. 3. 6. Хвилі Боллинджера

Окремо варто розглянути індикатор «Хвилі Боллинджера», розроблений Джоном Боллинджером. За своєю суттю, смуги Боллинджера – це осцилятор, який показує волатильність, тобто розкид ціни.

Він складається з трьох ліній, а саме:

- проста ковзна середня (SMA на 20 днів), по центру;
- верхня смуга: SMA 20 + (стандартне відхилення $\times 2$);
- нижня смуга: SMA 20 - (стандартне відхилення $\times 2$).

В основі обчислень, за якими формуються ці смуги, лежить стандартне відхилення, яке можна визначити за формулою:

$$S = \sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2} \quad (2.3)$$

де S – стандартне відхилення, незміщена оцінка середньоквадратичного відхилення випадкової величини x відносно її математичного очікування;

x_i – i -й елемент вибірки;

\bar{x} – середнє арифметичне вибірки;

n – обсяг (розмір) вибірки.

Основа Боллинджера – це коридор, який утворюється в результаті відтворення SMA і кордонів стандартного відхилення (рисунок 2.42).



Рисунок 2.42. Канал Боллинджера

Чим більше волатильність – тим смуги Боллинджера ширше. І навпаки, коли ціна має невеликий розкид (низька волатильність) – звужуються і смуги.

Нижче будуть наведені приклади використання індикатора на практиці:

1. Звуження і розширення

Смуги Боллинджера – ідеальний індикатор волатильності, що циклічно звужується або розширюється, демонструючи хвилеподібну структуру ціни (рисунок 2.43). Після кожного звуження каналу відбувається розширення – і ціна різко рухається вгору або вниз. Отже, після кожного «звуження» цілком логічно чекати розширення каналу.



Рис. 2.43. Смуги Боллинджера наочно вказують на наявність тенденцій та консолідацій.

Поки ціна знаходиться в каналі, це вказує на відсутність торгових можливостей і ринкових аномалій.

2. Пробивання лінії або високі/низькі ціни

Як правило, свічки на графіку знаходяться всередині каналу. Але як тільки ціна наближається до верхньої або нижньої лінії – це породжує кілька можливих торгових сигналів.

Наприклад, якщо свічка перетинає нижню лінію – ціна може почати рух вгору (рисунок 2.44).

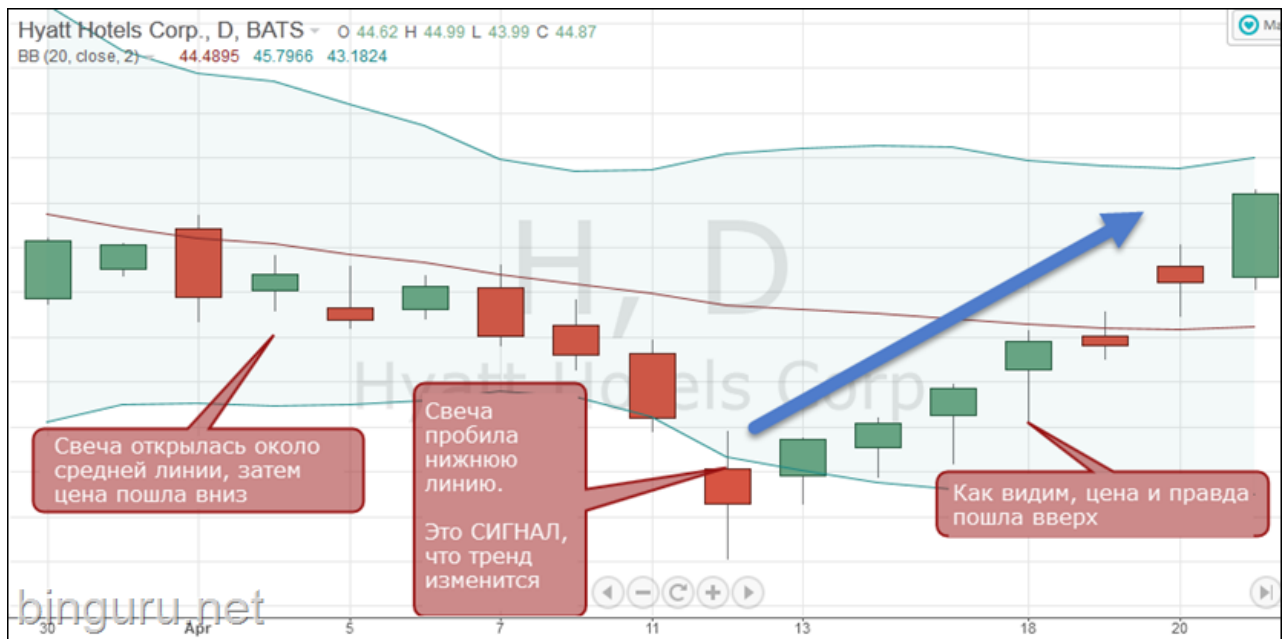


Рис. 2.44. Відпрацювання відбою від кордону каналу

Точно також працює й інший приклад – перетин верхньої лінії може означати швидке падіння.

Однак цей тип сигналу спрацює лише в тому випадку, коли немає яскраво вираженого тренду. Якщо він є – то перетин свічкою лінії Боллинджера буде свідчити лише про те, що тренд буде тривати.

Це називається «прогулянкою по смузі» (рисунок 2.45).

3. Прогулянка по смузі

Коли свічка перетинає лінію – це торговий сигнал, який несе в собі подвійну інтерпретацію про те, що:

- рух ціни зміниться;
- або тренд продовжиться.



Рис. 2.45. «Прогулянка по смуззі» вказує на продовження висхідного тренда

Виникає питання, як визначити, який саме сигнал полягає в тому, що стосується кордонів Боллинджера. Раніше в розділі про динамічні рівні підтримки та опору, було описано, що торговий алгоритм використовує дані двох SMA з періодами 100 і 200 для визначення поточного стану ринку: консолідації або тренду. В основі каналу Боллинджера так само лежить SMA, в даному випадку за замовчуванням встановлений період 50.

Алгоритм передбачає аналіз поточних значень і частоти перетину всіх трьох SMA між собою за заданих в налаштуваннях автоматизованої торгової системи, період. На основі цих даних визначається стан ринку.

Тепер, знаючи умови трактування сигналів на перетин кордонів каналу і поточний стан ринку, можна виділити наступні торгові прикази:

- якщо всі 3 SMA віддалені одна від одної (беручи до уваги, що відстань між старшими SMA за певний період збільшувалася не менше 50% часу, частота перетинів SMA 50 і 100 не перевищувала 2, а SMA з періодами 100 і 200 не перевищувала 1), то вважається, що на ринку тренд. Це означає, що перетин кордону каналу трактується як продовження тренда і дана ситуація та сигнал проторгується на пробій;

- в іншому випадку вважається, що на ринку немає яскраво вираженого тренду і перетин каналу трактується як сигнал на відбій.

В підсумку необхідно відзначити, що «смуги Боллинджера» це останній технічний індикатор, дані якого закладено в алгоритмічну модель торгової системи.

2. 5. Опціонний аналіз

2. 5. 1. Опис ринку ф'ючерсів

При описі ринку FOREX був відзначений той факт, що цей вид ринку не має єдиного дилінгового центру, через який можна було б відстежити операції купівлі-продажу на ринку. Проте, валютний ринок тісно пов'язаний з ринком ф'ючерсів і опціонів, торгівля якими проходить на великих біржах США і Європи. Перш ніж уточнити, в чому полягає опціонний аналіз на ф'ючерсних ринках і як він пов'язаний з FOREX, необхідно уточнити поняття ф'ючерсного контракту.

Ф'ючерс або ф'ючерсний контракт – це договір про фіксацію умов покупки або продажу стандартної кількості певного активу в обумовлений термін в майбутньому, за ціною, встановленою під час укладання договору. Загальноприйнято, що між фіксацією умов угоди і виконанням самої угоди проходить більше двох робочих днів, інакше ці угоди відносять до *Today*, *tom* і *Spot*.

В ф'ючерській угоді беруть участь дві сторони – продавець і покупець. Покупець ф'ючерсного контракту бере зобов'язання купити актив в обумовлений термін. Продавець ф'ючерсного контракту бере зобов'язання продати актив в обумовлений термін.

Обидва зобов'язання відносяться до стандартної кількості певного товару, в конкретний термін в майбутньому, за ціною, встановленою на момент укладання ф'ючерсу.

Основна корисна особливість ринку ф'ючерсів на відміну від FOREX – це той факт, що біржа, де відбувається торгівля ними, збирає, зберігає та

опубліковує всю інформацію за ф'ючерсними контрактами і опціонами, які були укладені всередині неї. І ця інформація доступна для зовнішнього аналізу.

2. 5. 2. Сутність опціонного аналізу

Метод опціонного аналізу заснований на оцінці біржових звітів найбільшої Північноамериканської Чиказької біржі (CME). Це одна з найбільших бірж, в неї входить Чиказька торгова палата (CBOT) і Нью-Йоркська товарна біржа (COMEX і NYMEX). Чиказька біржа на вимогу американського законодавства зобов'язана щодня викладати у вільний доступ звіти по всім торговим інструментам, які беруть участь у торгах. В опціонному аналізі беруть участь шість валютних пар, пов'язаних з індексом долара: EURUSD, GBPUSD, AUDUSD, USDCAD, USDCHEF і USDJPY. Ідея опціонного аналізу полягає в оцінці розстановки обсягів, їх переміщення та доглядів з ринку. Причому можна відстежити як гроші біржового натовпу, так і великих гравців, наприклад, хедж-фондів, центробанків і маркетмейкерів. Всі необхідні для аналізу дані можна щодня викачувати у вільному доступі з сайту Чиказької біржі CME Group. Ці звіти мають такий вигляд (рисунок 2.46).



27 Side 01		BRITISH POUND CALL OPTIONS							27 Side 01	
		2012 DAILY INFORMATION BULLETIN - http://www.cmegroup.com/dailybulletin								
		CME Group, Inc.								
		20 South Wacker Drive, Chicago, Illinois 60606-7499. Customer Service: (800) 331-3332 Fax: (312) 930-8203 E-Mail: info@cmegroup.com								
		PRELIMINARY								
PG27 BULLETIN #126@		BRITISH POUND CALL OPTIONS							Fri, Jun 29, 2012 PG27	
INDIVIDUAL CONTRACT PRICE DATA FOR OPEN, HIGH, LOW, CLOSE AND SETTLEMENT PLUS CLEARED VOLUME FIGURES REFLECT CME REGULAR TRADING HOURS SESSION ONLY. RTH VOLUME REFLECTS PIT TRADING AND CASH-FOR-FUTURES ONLY. VOLUME OR OPEN INTEREST (BOTH BEFORE AND AFTER THE LAST DAY OF TRADING) MAY BE AFFECTED BY: CASH FOR FUTURES, SPREADS, PRIOR DAYS' CLEARED TRADES (OUT-TRADES), POSITION ADJUSTMENTS, OPTIONS EXERCISES, POSITIONS IN DELIVERY, OR POSITIONS IN A CASH SETTLEMENT CYCLE. B=BIID A=ASK N=NOMINAL P=POST SETTLEMENT SESSION #=NEW CONTRACT HIGH PRICE *=-NEW CONTRACT LOW PRICE R=RECORD VOL OR OPN INT. SETTLEMENT PRICE DETERMINED BY CME RULE 813. B=BIID A=ASK N=NOMINAL P=POST SETTLEMENT. CAB= (CABINET TRANSACTION) - 1/2 TICK VALUE = \$6.25. PRODUCT LISTINGS REPRESENT CONTRACTS WITH PRICE/VOLUME ACTIVITY AND/OR HAVE ESTABLISHED OPEN INTEREST. PRODUCTS ELIGIBLE TO TRADE BUT ARE INACTIVE DO NOT APPEAR IN THIS REPORT.										
BRITISH POUND CONTRACTS LAST TRADE DATES										
EXPIRATION:	JUN12	JUL12	AUG12	SEP12	DEC12	MAR13	JUN13			
BRIT PND OPT	06/08	07/06	08/03	09/07	12/07	03/08	06/07			
WKBP-1B-C	06/01									
WKBP-2B-C		07/13	08/10							
WKBP-3B-C	06/15	07/20								
WKBP-4B-C	06/22	07/27								
WKBP-5B-C	06/29									
BRIT PND OPT (EUR)	06/08	07/06	08/03	09/07	12/07	03/08	06/07			
B PND WK 1 (EUR)	06/01									
B PND WK 2 (EUR)		07/13	08/10							
B PND WK 3 (EUR)	06/15	07/20								
B PND WK 4 (EUR)	06/22	07/27								
B PND WK 5 (EUR)	06/29									
BRITISH POUND OPTIONS ON FUTURES										
ELIGIBLE OPTIONS SERIES (CALLS & PUTS) FOR TRADING: FRI, JUN 29										
		WEEK 1	WEEK 2	WEEK 3	WEEK 4	WEEK 5				
1B	JUN12	129.0-189.0		129.0-187.0	129.0-187.0	129.0-187.0	129.0-187.0			
	JUL12		129.0-187.0	129.0-187.0	129.0-187.0					
	AUG12		129.0-187.0							
1E	JUN12	737.5-942.5		737.5-942.5	737.5-942.5	737.5-942.5				
	JUL12		737.5-942.5	737.5-942.5	737.5-942.5					
	AUG12		737.5-942.5							
1P	JUN12	129.0-189.0		129.0-187.0	129.0-187.0	129.0-187.0				
	JUL12		129.0-187.0	129.0-187.0	129.0-187.0					
	AUG12		129.0-187.0							

Рис. 2.46. Звіт CME

Звіти CME – це звід таблиць. Проаналізувавши ці звіти, можна отримати всю необхідну інформацію про розстановку обсягів великих гравців на поточний день та торговий опціонний місяць. За допомогою цих звітів можна дізнатися реальні обсяги капіталу. На ринку FOREX отримати обсяги неможливо, тому що він не має єдиного торговельного майданчика, а сама торгівля ведеться між крупними банками. Торгові стратегії типу VSA не мають під собою сили, тому що в них використовується тиковий обсяг, а це інструмент дуже приблизний і неточний. Звіти CME надають дані щодо обсягів по ф'ючерсах, зокрема в звітах публікується інформація про обсяги опціонів на валютні ф'ючерси: EUR, GBP, AUD, CAD, CHF і JPY. З них при додаванні індексу USD формується шість основних валютних пар, що використовуються в торгівлі на FOREX.

З цього випливає необхідність пов'язати між собою ринок FOREX, ринок ф'ючерсів і ринок опціонів. Якщо порівняти графік ф'ючерсу на євро і графік валютної пари EURUSD на FOREX (рисунк 2.47), то можна помітити, що графіки практично ідентичні, вони повторюють один за одним всі рухи, але мають відмінності за волатильністю і поточною ринковою ціною.

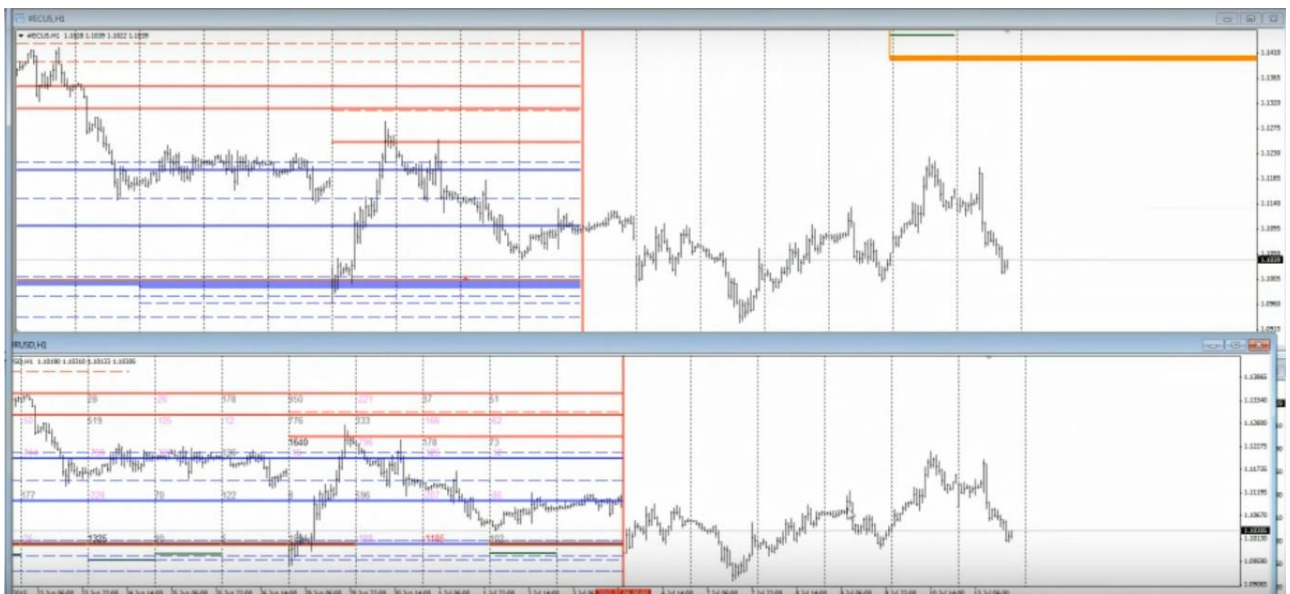


Рис. 2.47. Порівняння графіків котирувань ф'ючерсу на євро і валютної пари EURUSD на FOREX

Оскільки на FOREX не можливо отримати реальних обсягів, то виникає необхідність проведення аналізу даних по опціонах на валютні ф'ючерси в звітах CME за минулу торгову сесію. Завдяки цій інформації стає можливою побудова на графіку опціонних рівнів, від яких відштовхується ціна з приголомшливою точністю.

Раніше було описано, що таке ф'ючерсний контракт, проте для розуміння механізму опціонного аналізу, потрібно знати, що з себе представляє опціон.

Опціон, як і ф'ючерс, надає право здійснити операцію купівлі-продажу конкретного товару за конкретною ціною, зафіксованою в момент часу і яку було затверджено учасниками угоди. Це основна відмінність від ф'ючерсу. Якщо ф'ючерс – це зобов'язання, то опціон – це право.

Опціони використовуються різними великими фінансовими інститутами (маркетмейкерами, банками, хедж-фондами) для хеджування ризиків. Торгуючи опціонами, а саме, продаючи їх, можна заздалегідь прорахувати свій прибуток, а також з високою точністю спрогнозувати, до яких цінових рівнів дійде ціна в певний проміжок часу, і в якому ціновому коридорі протягом цього часу ціна буде торгуватися. Це дуже вигідно маркетмейкерам.

Опціонний аналіз – це симбіоз фундаментального і технічного аналізу, тому важливо розуміти економічні та політичні чинники, які рухають ціну в певний бік.

Припустимо, сьогодні 05.07.2016 р, а поточна ціна по одній з валютних пар - 1.1145 на FOREX, проте на ринку ф'ючерсів можна купити даний торговий інструмент за ціною 1.0990 на дату 30.07.2016 р, тобто нижче ринкової ціни, або продати за ціною 1.1400, тобто вище ринкової ціни (рисунок 2.48). На відміну від опціонного ринку, на FOREX неможливо купувати і продавати вище або нижче поточної ціни, є можливість лише виставляти відкладені ордери.



Рис. 2.48. Пример опционного анализа

Опціони бувають двох типів – CALL і PUT. Опціон CALL виступає в ролі опору, а опціон PUT – в ролі підтримки. Опціони можна купити або продати. При продажу опціонів продавець буде отримувати певну комісію, яку оплачує покупець. Ця комісія називається премією опціону. При продажу опціону CALL за ціною 1.1400, наприклад, продавець відразу отримує премію 100 пунктів за ризик, а збиток він отримає тоді, коли ціна піде вище премії за ризик, тобто $1.1400 + 100 = 1.1500$. При продажу опціону PUT трейдер отримає збиток, якщо ціна піде нижче цих 100 пунктів, тобто $1.0990 - 100 = 1.0890$. Чим ближче до ринкової ціни продається опціон, тим вище премія, тому ще є ризик, що ціна дійде до рівня відкриття опціону. При покупці і продажу опціонів виникають зовсім різні ризики. Коли трейдер купує опціон, він отримує обмежений збиток і необмежений прибуток. І навпаки, при продажу опціону він отримує обмежений прибуток і необмежений збиток. Здавалося б,

вигідніше купувати опціони, але учасники біржі в основному їх продають. У чому переваги саме продажу опціонів:

- По-перше, маркетмейкери заздалегідь знають розмір свого майбутнього прибутку. Ця інформація їм потрібна, щоб захеджировати певну суму грошей;
- По-друге, якщо на момент експірації (закриття) опціону ціна знаходиться в зоні збитку, то отримані збитки не фіксуються, натомість угода виводиться на ринок FOREX. В цьому випадку маркетмейкери починають маніпулювати ринком і рухати значення ціни в зону без збитку;
- По-третє, простіше спрогнозувати ринок. Аналітичні відділи маркетмейкерів враховують всі фундаментальні фактори і визначають ціновий коридор, в якому буде торгуватися ціна до моменту експірації опціону.

Володіючи такою інформацією зі звітів CME, можна визначити, на яких цінових рівнях накопичується найбільша кількість контрактів. Чим більше обсяг, тим сильніше рівень. Фактично обсяги на опціонних рівнях показують, скільки було продано опціонів CALL (верхній рівень цінового коридору) і PUT (нижній рівень). Відповідно, від цих рівнів можна незабаром очікувати відскоку ціни.

Таким чином, завдяки аналізу опціонних рівнів, які можна отримати з щоденних біржових звітів, стає наглядною модель ринку. За допомогою опціонних рівнів стає можливим побачити розстановку реальних грошей біржових гравців, і це дозволяє з упевненістю знати, куди піде ціна. Таким чином, з'являється можливість достовірно визначити, з яких цінових рівнів варто очікувати зміни тренда або корекції ринку, слідкуючи за намірами великих гравців.

Торговий робот аналізує дані зі звітів CME і вибудовує ці рівні на графіку. Якщо вони збігаються з іншими орієнтирами, отриманими вході графічного і

індикаторного аналізу, то цінова зона вважається сильною і, ґрунтуючись на поточній ринковій ситуації, може проторговуватися певним чином.

2. 6. Патерни Price Action

2. 6. 1. Школа Price Action. Поняття патернів та моделей

Школа Price Action (перекладається з англійської, як "рух ціни") – це вид свічкового аналізу, який полягає в роботі з чистим графіком японських свічок. Під цим терміном мається на увазі аналіз цінового руху за певний проміжок часу. Робота полягає в пошуку постійно повторюваних цінових моделей ("патерни" або "сетапи"), які вказують на зміну настрою учасників ринку. Price Action відмінно комбінується з іншими видами технічного аналізу, де найчастіше сигнали підтверджуються кількома індикаторами і лініями підтримки та опору. Це робить дану школу частиною технічного аналізу.

Як відомо з теорії Доу, ціна включає в себе всі цінові фактори, що на неї впливають, будь-які новини і події. Економічні показники фільтруються трейдерами, що працюють вручну або через алгоритмічні системи, а рух, викликаний їх роботою, можна побачити на самому графіку. Отже, замість того, щоб намагатися аналізувати мільйони економічних факторів (що майже неможливо), послідовники цієї школи займаються вивченням самої суті цінового руху. Це дає можливість торгувати, фактично, за тими ж економічними факторами, що вже містяться в динаміці ціни.

Price Action може використовуватися на будь-якому фінансовому ринку: бінарні опціони, FOREX, ф'ючерси, біржові опціони, акції. Традиційно Price Action знайшов своє застосування в роздрібному FOREX, куди простіше отримати доступ, ніж на фондові ринки. Освоюють його і трейдери бінарними опціонами (далі – БО) , враховуючи, що багато брокерів FX давно додали БО в свої платформи.

Існує велика кількість патернів і сетапів, однак немає необхідності використовувати їх всі, досить виділити найбільш ефективні і визначити ситуації, при яких дані патерни краще відпрацьовують. Валютний ринок

регулярно демонструє одні і ті ж поведінкові моделі, тому ця школа здобула велику популярність на заході, а зараз отримує популярність в СНД.

Як було сказано раніше, при побудові графіка японських свічок, трейдер отримує можливість бачити характер руху ціни за певний момент часу. Свічки, які формуються в залежності від обраного таймфрейма, мають різний напрямок, розмір щодо інших свічок, співвідношення тіла і тіней. Одна або кілька свічок здатні сформувати патерн.

Патерн або сетап – це комбінація кількох свічок, які вказують на певну поведінку ціни в найближчому майбутньому. Нижче на графіку представлений торговий сигнал, який використовується на трендовому ринку, відомий як *inside bar* – внутрішній бар (рисунок 2.49).



Рис. 2.49. Пример одного из самых популярных трендовых паттернов

Крім цього, Price Action використовується для визначення ринкового тренду. Щоб визначити рух вниз, досить знайти нижні максимуми (LN - lower

high) і нижні мінімуми (LL - lower low). Для тренда вгору необхідно знаходити, відповідно, верхні максимуми (HH - higher high) і верхні мінімуми (HL - higher low).

В наведеному прикладі видно, як спадний тренд переходить в висхідний і назад в спадний (рисунок 2.50). Саме таким чином, торговий робот визначає ринковий тренд за певний період, якщо використання ковзних середніх відключено. Якщо ж ні – то використовуються обидва методи, що тільки підсилює якість аналізу.

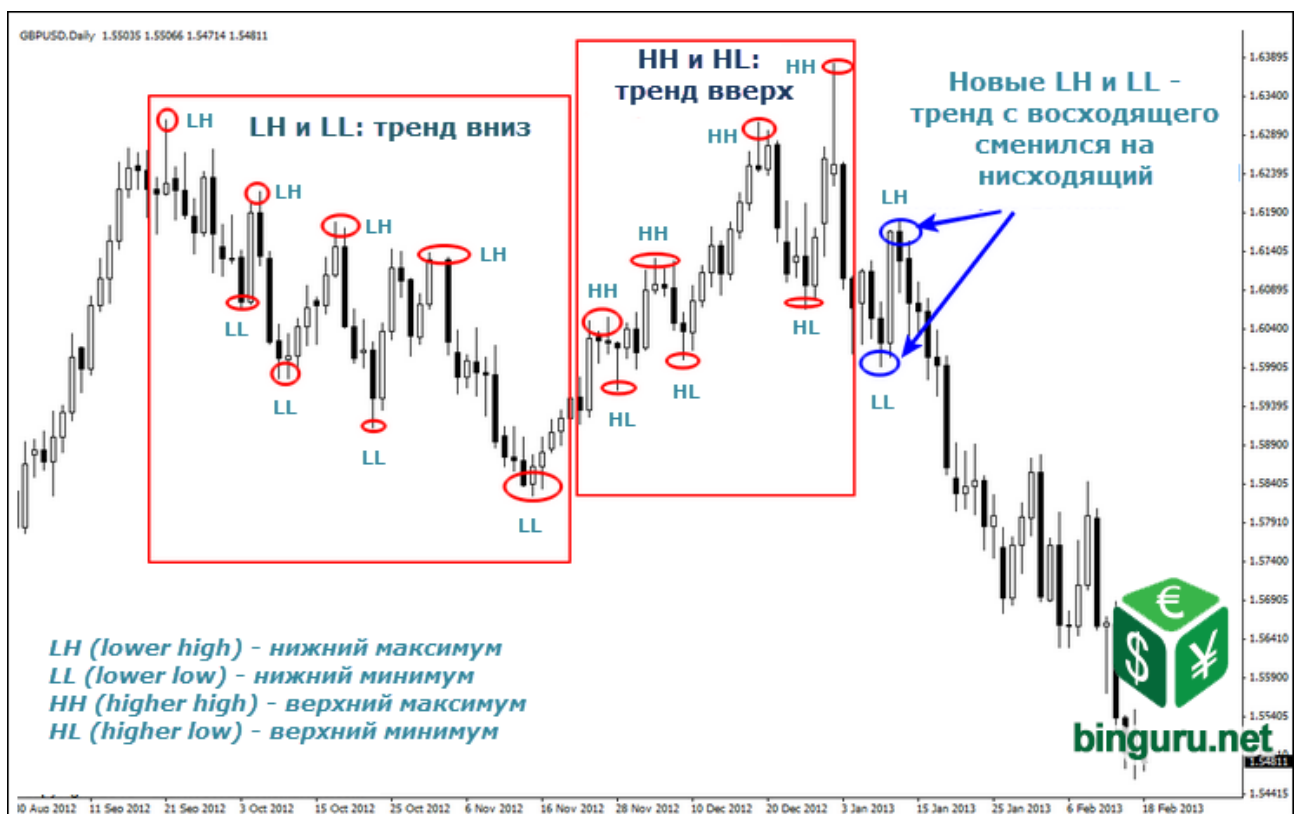


Рис. 2.50. Визначення тренда без використання ковзних середніх

Незважаючи на те, що Price Action здатний видавати дуже сильні і надійні сигнали, технічний аналіз в цілому має на увазі структурний підхід до торгівлі з використанням декількох сигналів, де під "структурністю" слід розуміти зони або цінові рівні, що мають очевидно важливе значення. У цих структурних зонах необхідно знайти поєднання кількох факторів, таких як, наприклад, явний сигнал Price Action і ключовий ціновий рівень.

Існує чимало структурних факторів (деякі з них показані на графіку нижче). Як видно, вони формуються на ключових рівнях підтримки та опору, що вже є важливим структурним елементом. Утворюється відразу кілька сигналів, включаючи пінбар, внутрішній бар і сетап "фейки" (рисунок 2.51).

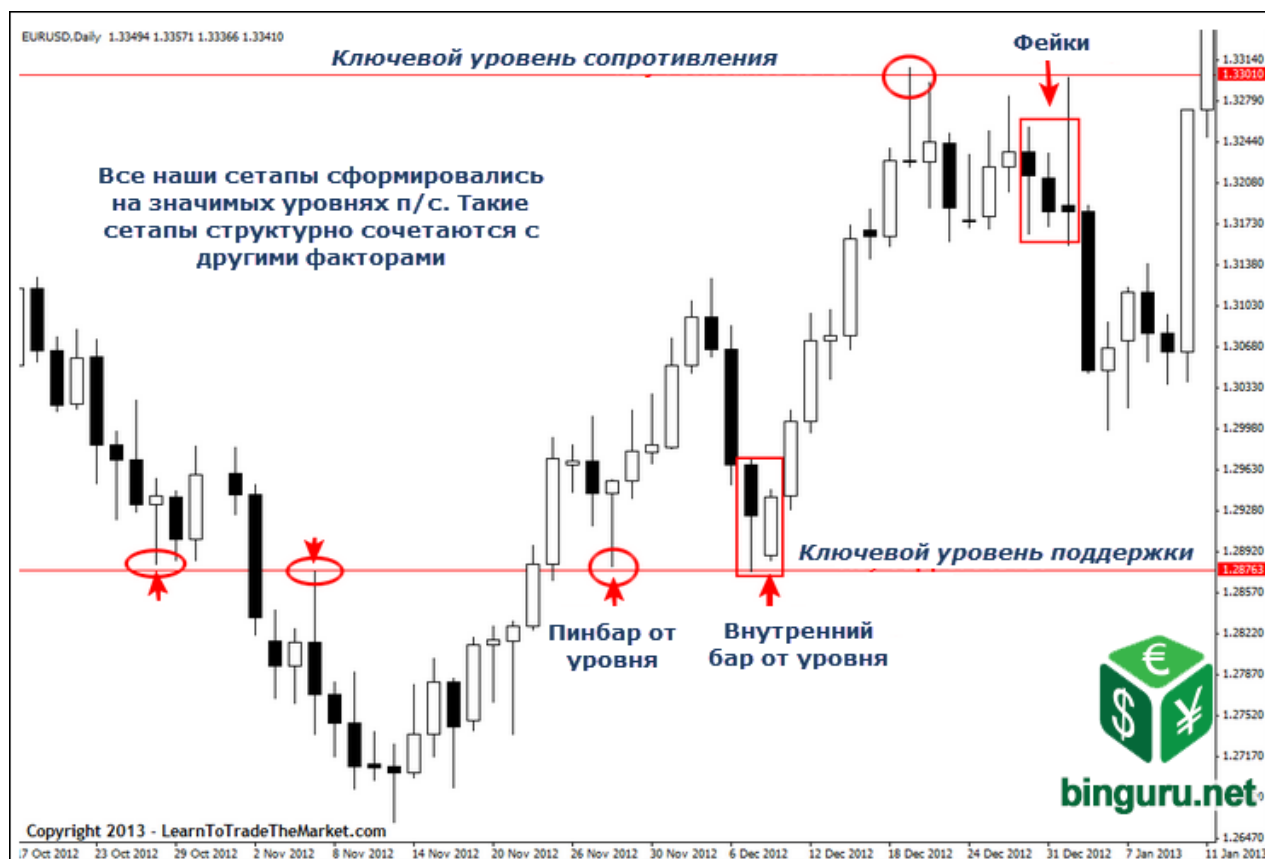


Рис. 2.51. Поєднання ліній підтримки і опору з сетапом Price Action, дає можливість краще визначити поведінку ціни на важливих, структурних рівнях

2. 6. 2. Аналіз ефективності патернів Price Action

Мета аналізу – визначити ефективність свічкових патернів на історії котирувань, а також виявити кореляцію між їх прибутковістю та днем тижня або місяця, в якому ці патерни виникли. Це можливо реалізувати шляхом збору попередньої статистики по кожному паттерну по кожній валютній парі на таймфрейме H1-D1 згідно з такими параметрами:

- оптимальне співвідношення стоп-лосс до тейк профіту;

- розподіл угод по днях тижня;
- розподіл угод по днях місяця;
- розподіл угод по місяцях року.

Результати, отримані в ході аналізу, використовуються для складання вагових коефіцієнтів в торговому алгоритмі.

Під час аналізу досліджуються такі патерни:

- доджі;
- пін-бар;
- поглинання;
- внутрішній бар.

Для визначення ефективності патернів, розглянутих в попередньому пункті, була використана спрощена версія автоматизованої торгової системи (без опціонного і індикаторного аналізів, вагових коефіцієнтів під час визначення рівнів та прив'язки до динамічних ліній підтримки й опору MA).

Проте свічкові патерни у будь-якому випадку розглядаються в контексті поточної ситуації, а саме: патерн повинен спиратися на рівень і вхід повинен бути в напрямку поточного тренда. Визначення тренда відбувається за допомогою методу з ковзними середніми, описаними раніше.

Для визначення рівнів використовується пошук пари фракталів приблизно на однаковому рівні з погрішністю, яка дорівнює 10 пунктам на історії в 100 свічок. При цьому не важливо, був це фрактал вгору чи вниз – головне, що ціна розгорталася.

Всього було проаналізовано близько 90 тисяч угод по 23 валютним парам на трьох періодах по чотирьом свічним паттернам з 2000 року. У цьому пункті надані результати аналізу тайм-фрейму D1. Результати аналізу за періодами H1 та H4 наведені у додатку Г.

Графіки прибутковості (рисунки 2.52 – 2.55):

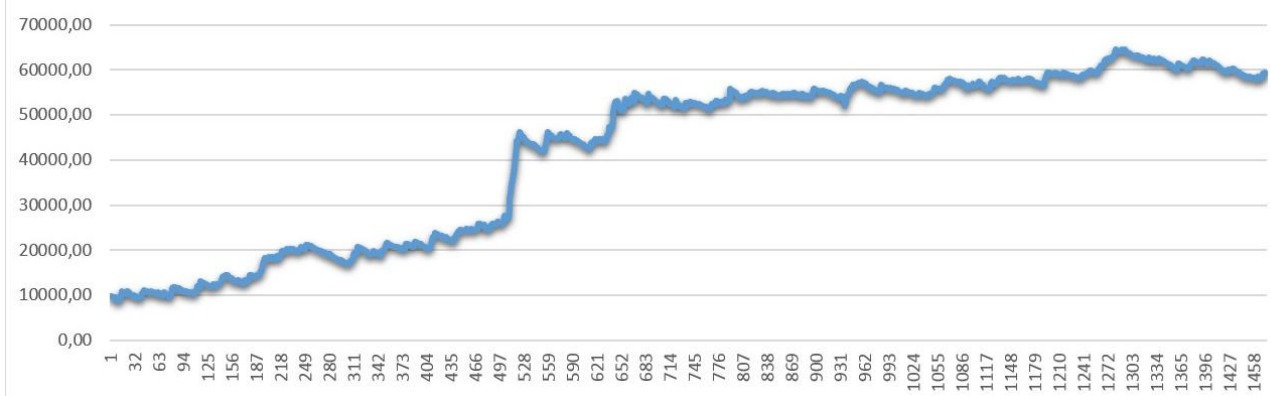


Рис. 2.52. Графік прибутковості патерну «внутрішній бар»

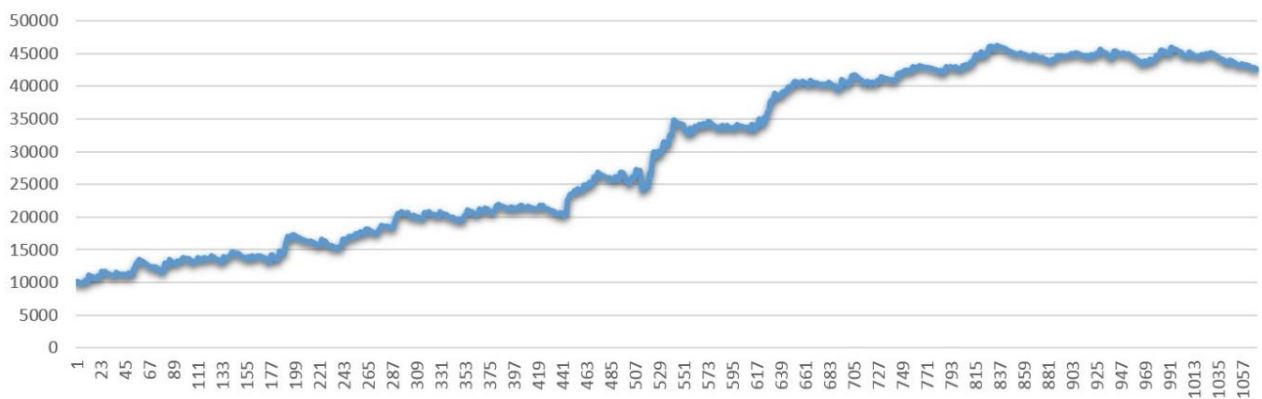


Рис. 2.53. Графік прибутковості патерну «доджи»

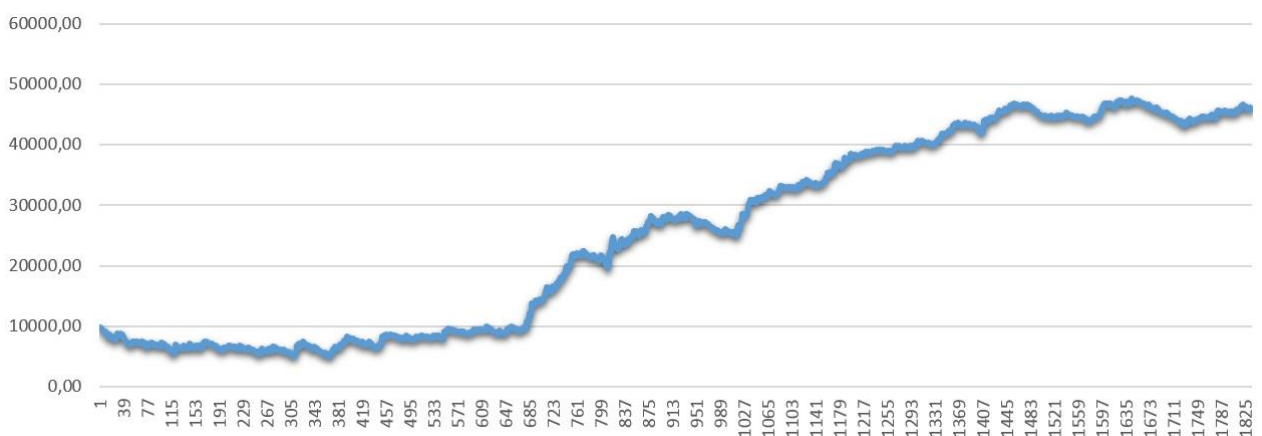


Рис. 2.54. Графік прибутковості патерну «пінбар»

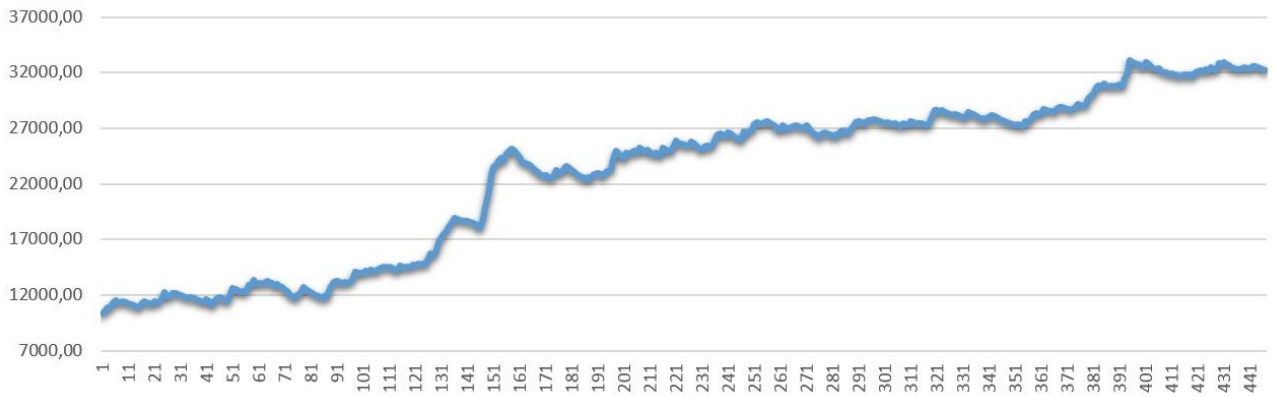


Рис. 2.55. Графік прибутковості патерну «поглинання»

Торгівля будь-яким з патернів навіть без додаткової фільтрації, без урахування рівнів і тренда, на періоді D1 здатна приносити стабільний прибуток.

Оптимальні відносини стоп-приказу до тейк профіту (рисунки 2.56 – 2.59):

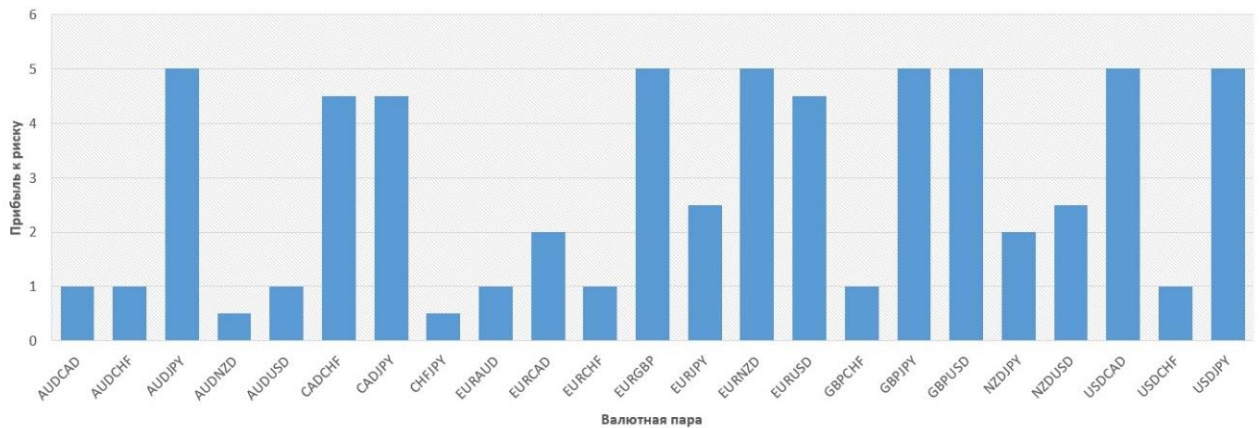


Рис. 2.56. Оптимальне відношення SL к TP для патерна «внутрішній бар»

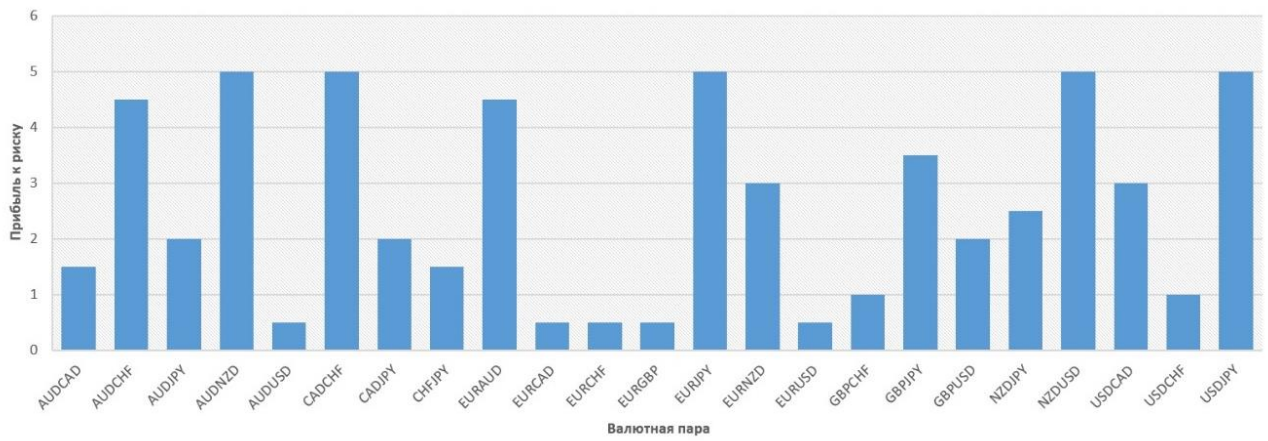


Рис. 2.57. Оптимальне відношення SL к TP для патерна «доджи»

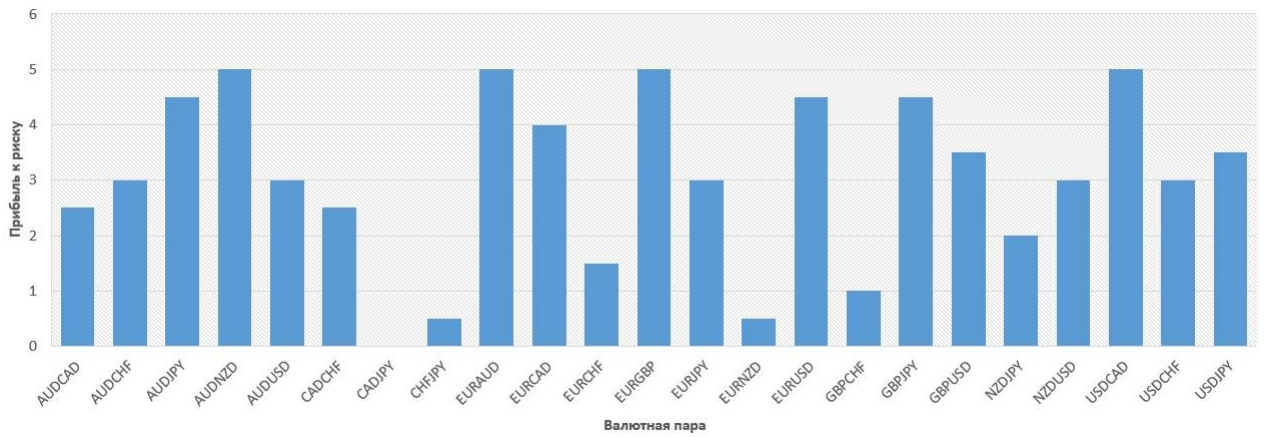


Рис. 2.58. Оптимальне відношення SL к TP для патерна «пінбар»

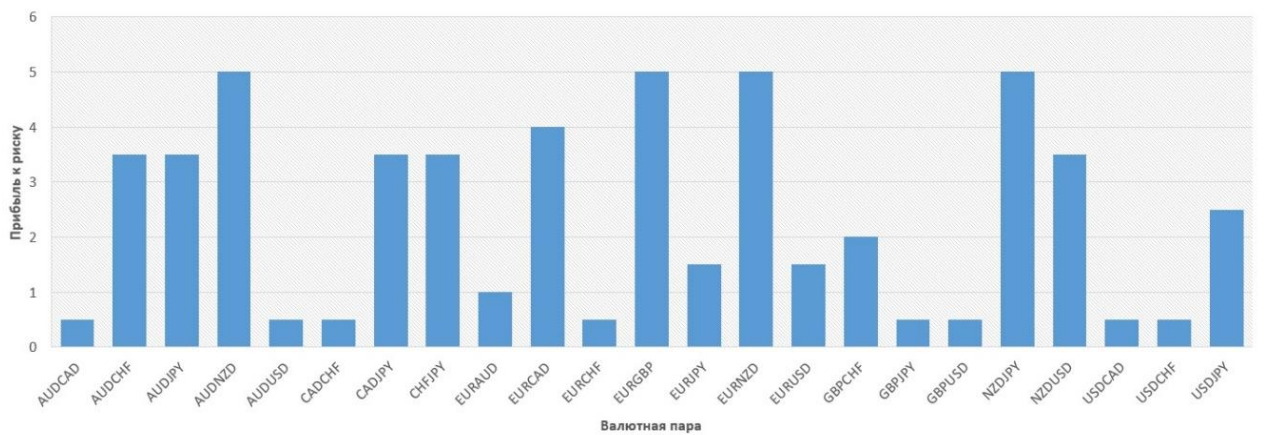


Рис. 2.59. Оптимальне відношення SL к TP для патерна «поглинання»

Оптимальне співвідношення прибутку до збитку становить в середньому від 3:1 до 4:1.

Прибутковість за днями тижня (рисунки 2.60 – 2.63):

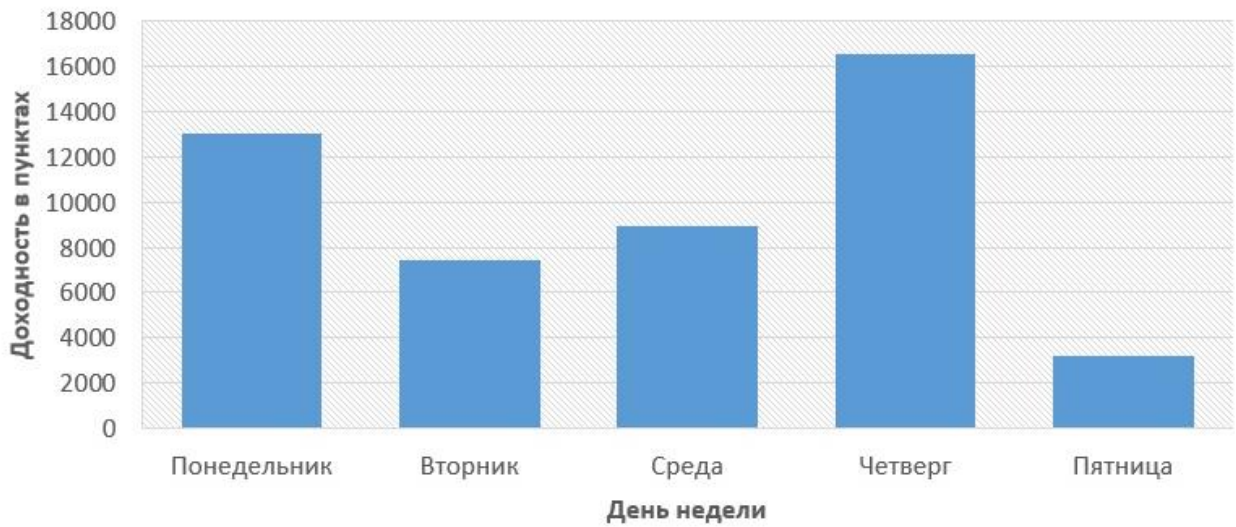


Рис. 2.60. Статистика прибутку для патерна «внутрішній бар» за днями тижня

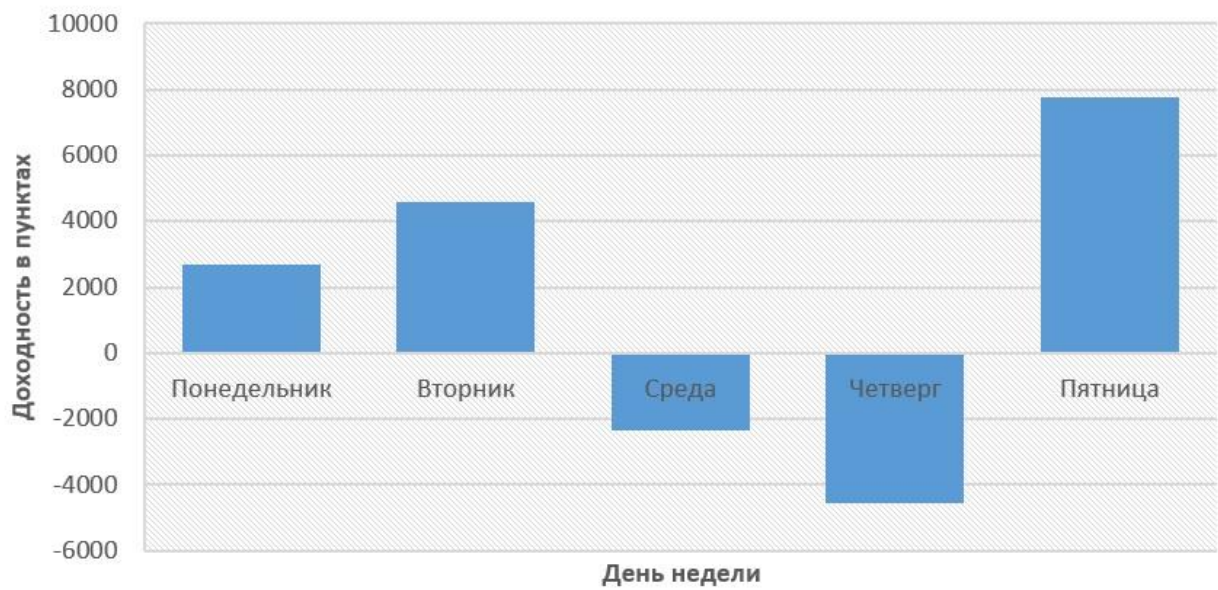


Рис. 2.61. Статистика прибутку для патерна «доджи» за днями тижня

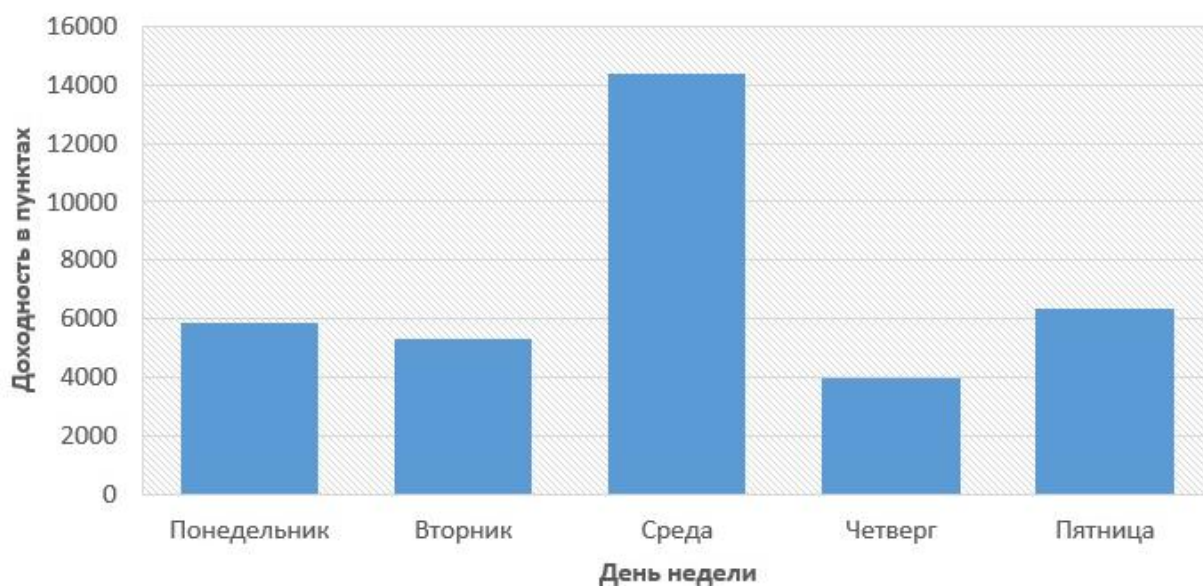


Рис. 2.62. Статистика прибутку для патерна «пінбар» за днями тижня

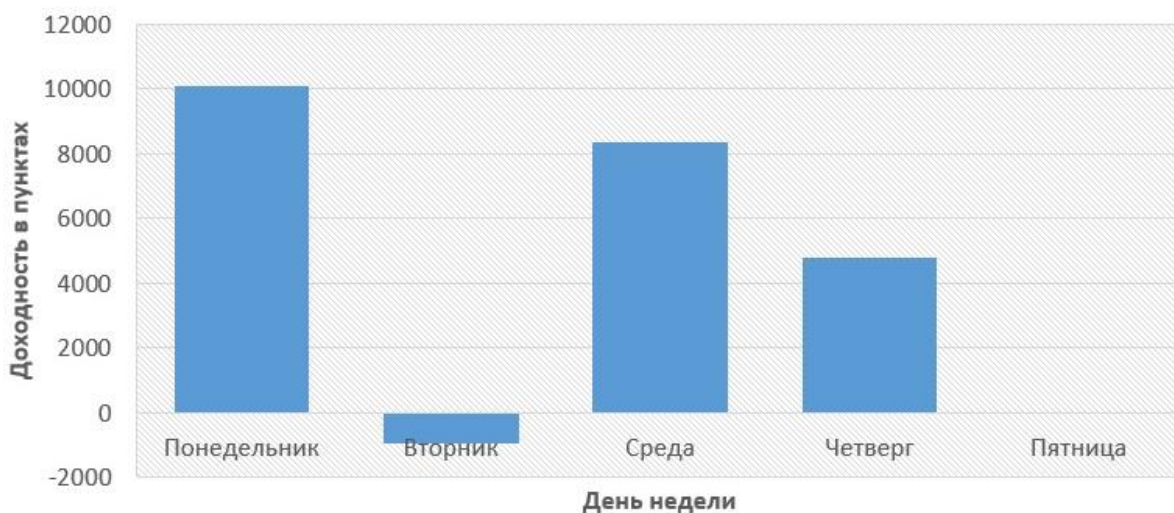


Рис. 2.63. Статистика прибутку для патерна «поглинання» за днями тижня

Вивчення прибутковості за днями тижня дає можливість зробити висновок, що внутрішній бар на періоді D1 незадовільно працює при сигналі в п'ятницю, а доджи – в середу і четвер. Пінбар найкраще відпрацьовує в середу. «Поглинання» відпрацьовує як слід в понеділок, середу та четвер.

Прибутковість за числами місяця (рисунки 2.64 – 2.67):

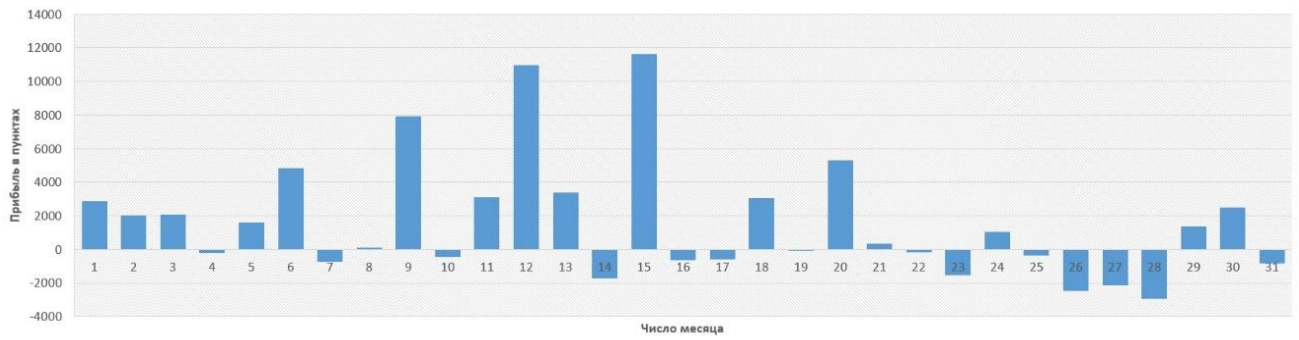


Рис. 2.64. Статистика прибутку патерну «внутрішній бар» за числами місяця

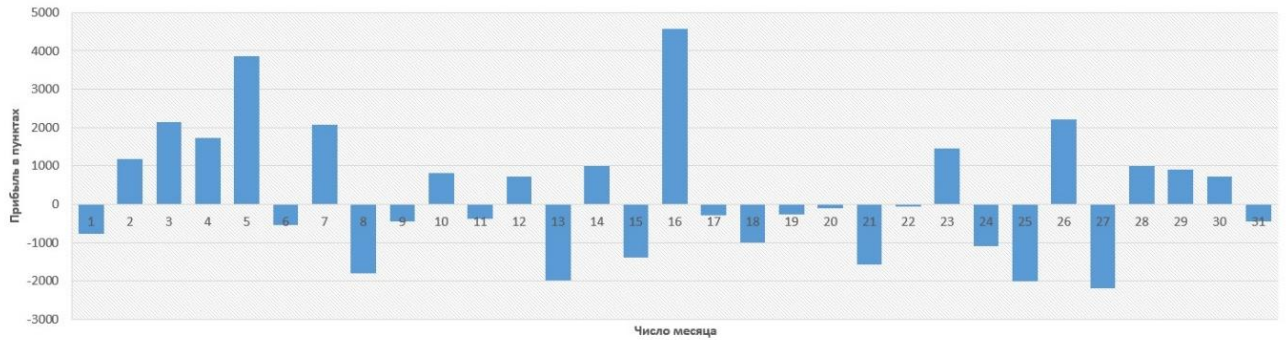


Рис. 2.65. Статистика прибутку патерну «доджи» за числами місяця

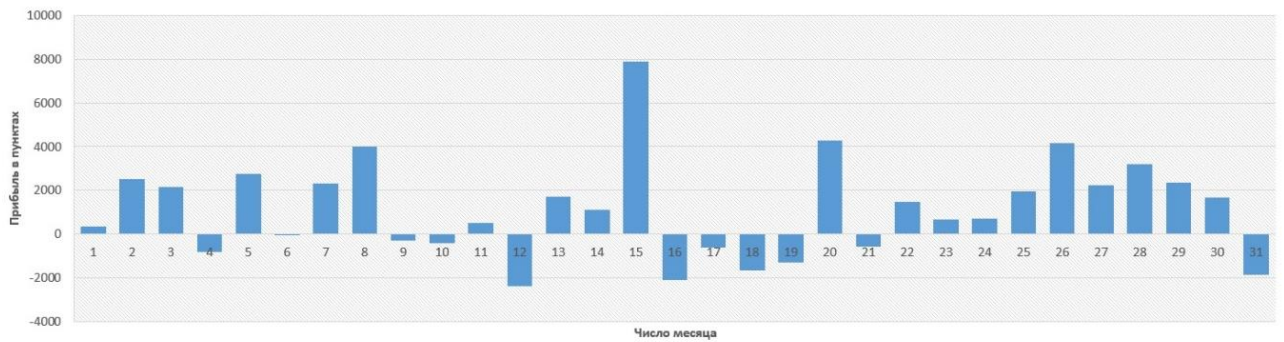


Рис. 2.66. Статистика прибутку патерну «пінбар» за числами місяця

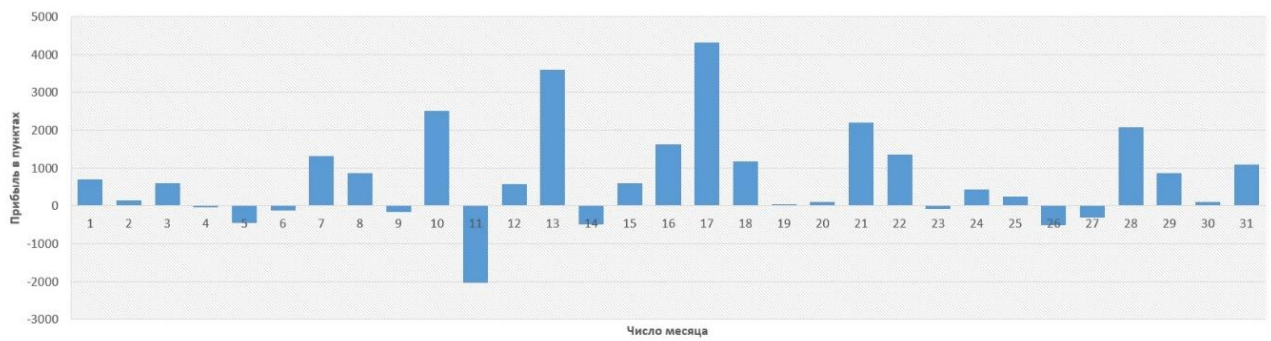


Рис. 2.67. Статистика прибутку патерну «поглинання» за числами місяця

Прибутковість за місяцями року (рисунки 2.68 – 2.71):

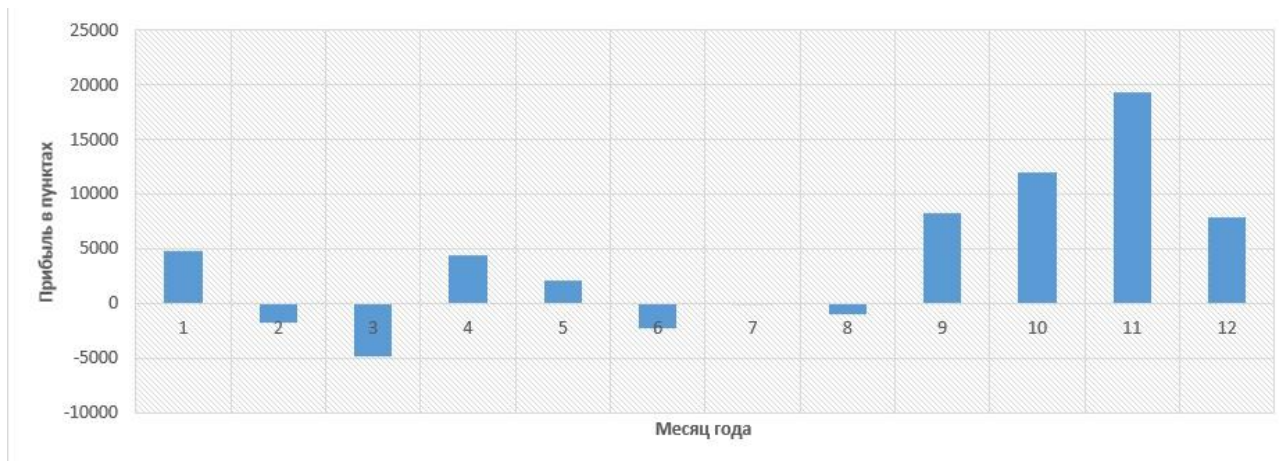


Рис. 2.68. Статистика прибутку патерну «внутрішній бар» за кожен місяць

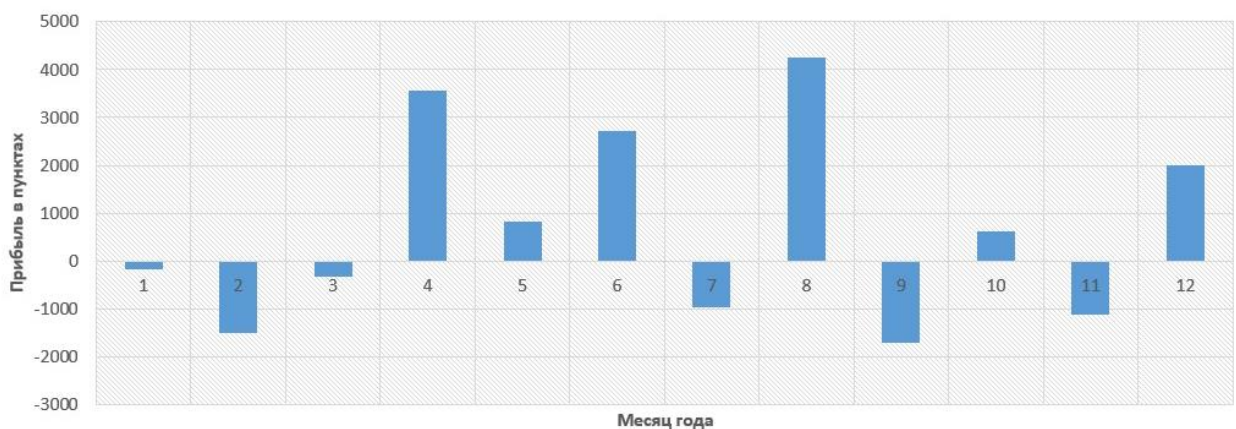


Рис. 2.69. Статистика прибутку патерну «доджи» за кожен місяць

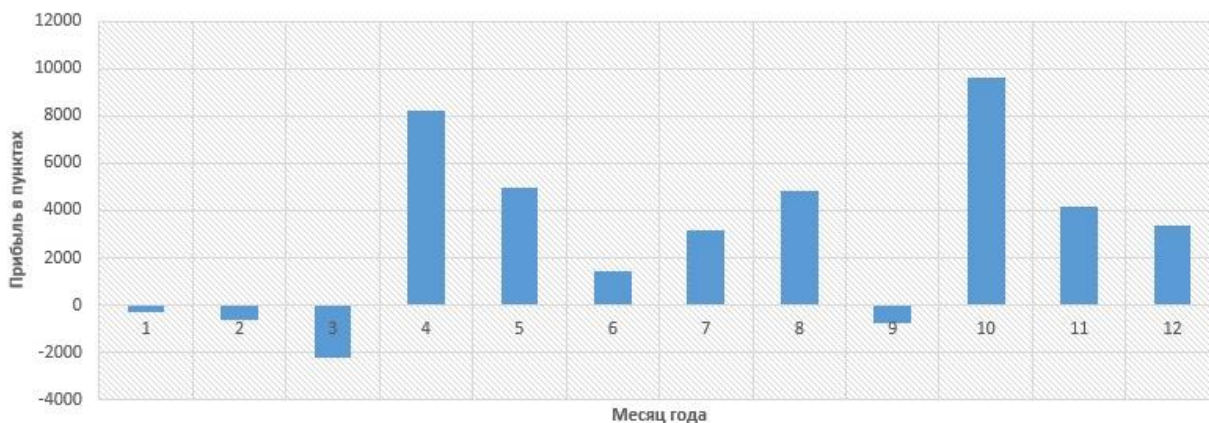


Рис. 2.70. Статистика прибутку патерну «пінбар» за кожен місяць

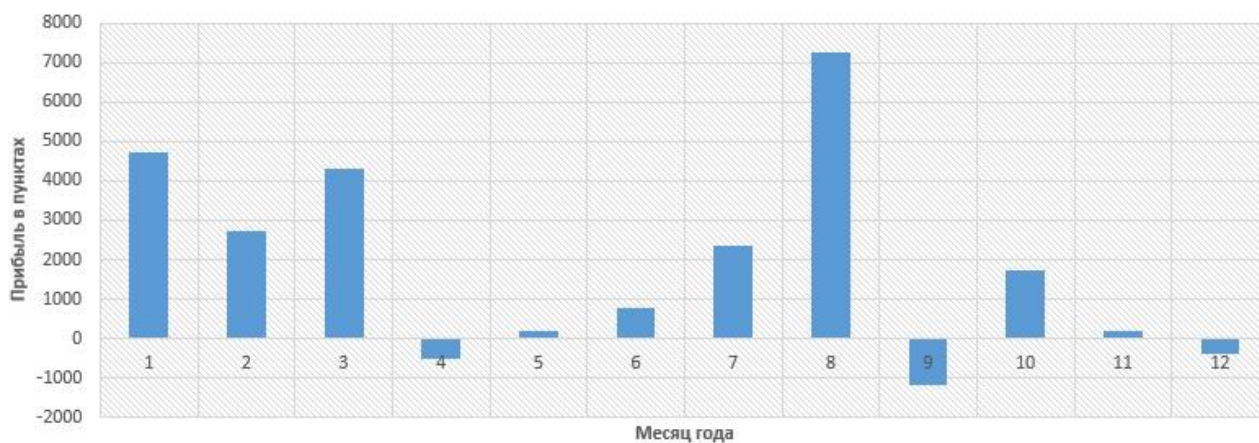


Рис. 2.71. Статистика прибутку патерну «поглинання» за кожен місяць

За результатами аналізу ефективності патернів Price Action, єдину закономірність, яку можна відстежити при роботі на D1 щодо місяців року – це той факт, що статистично більшість патернів мають високу ефективність в зимову пору року.

На періоді D1 всі патерни Price Action навіть без додаткового підтвердження цілком можуть приносити прибуток. Але, зазвичай, кращих результатів можна досягти, враховуючи напрям тренда і основні рівні на графіку.

На основі роботи спрощеного алгоритму торгового робота і зібраних результатів з'явилася можливість виявити, що кореляція в залежності від дня

тижня дійсно існує. І найголовніше – була доведена ефективність використання свічкового аналізу Price Action на старших таймфреймах.

Це доводить обґрунтованість вибору основного джерела аналізу ринкової динаміки під час побудови автоматизованої торгової системи.

Висновки

1. Існує два основних типи аналізу ринкового ціноутворення – технічний та фундаментальний.
2. Фундаментальний аналіз на поточному етапі розвитку технологій дуже важко використовувати під час розробки автоматизованих торгових систем. Це пояснюється суб'єктивним методом оцінки подій, що трапляються на світових ринках, багато з яких неможливо передбачити.
3. Більшість автоматизованих торгових систем будуються на основі різноманітних засобів технічного аналізу ринку.
4. Рівні підтримки та опору – це базові структурні компоненти аналізу, без алгоритмізації пошуку яких неможлива ефективна торгівельна діяльність автоматизованої системи.
5. У якості підтвердження торгових сигналів, використовуються засоби індикаторного та опціонного аналізу, які дозволяють детальніше і краще оцінити поточну ринкову ситуацію.
6. Патерни Price Action – це окрема школа технічного аналізу, яка спеціалізується на свічковому аналізі та виявленні закономірностей при утворенні японських свічок.
7. Проведена оцінка ефективності патернів Price Action наглядно демонструє ефективність цієї школи під час прогнозування ринкової динаміки, навіть без використання додаткових інструментів аналізу.

РОЗДІЛ 3.
РЕАЛІЗАЦІЯ АЛГОРИТМУ І ЗАСОБІВ УПРАВЛІННЯ РАХУНКОМ
АВТОМАТИЗОВАНОЇ ТОРГОВОЇ СИСТЕМИ

3. 1. Засоби розробки

3. 1. 1. Опис компонентів торгової платформи MetaTrader 4

MetaTrader – інформаційно-торгова платформа, розроблена MetaQuotes Software Corp., призначена для організації дилінгового обслуговування на ринках Forex, CFD и Futures. Це комплекс повного циклу, тобто для організації дилінгового обслуговування при наявності MetaTrader 4 не потрібно додаткового програмного забезпечення. Серверна частина працює тільки на платформі сімейства Windows. Клієнтська частина є в версіях для Windows, Android і iOS.

Станом на червень 2016 року активно використовується четверта версія платформи і випущена п'ята. Перша, друга і третя версії платформи не використовуються і не підтримуються.

Офіційний випуск інформаційно-торгової платформи MetaTrader 4 відбувся 1 липня 2005 року. Платформа включає в себе наступні компоненти:

MetaTrader 4 Server – ядро системи, серверна частина. Призначена для обробки запитів користувачів на здійснення торгівельних операцій, виставлення і виконання ордерів. Також транслює котирування і новини, веде протоколи і архіви. Працює у вигляді служби. Не має самостійного інтерфейсу. Виробник радить організувати виділений сервер з платформою Windows Server, на якому крім MetaTrader 4 Server більше нічого не запускається. Практика показує, що паралельний запуск на цьому ж обладнанні будь-який інший програми різко знижує стійкість роботи MetaTrader 4 Server. Використання кожної копії ліцензується окремо.

MetaTrader 4 Administrator – призначений для віддаленого управління серверної частиною у вигляді вказівки параметрів настройки, настройками фінансових інструментів, баз даних і так далі. Має незначні можливості операцій з клієнтськими рахунками. Платформа Windows Vista / XP / 2000.

Ліцензія на використання необмеженого числа копій включена до ліцензії сервера.

MetaTrader 4 Manager – призначений для обробки торгових запитів і управління рахунками клієнтів. Платформа Windows Vista / XP / 2000. Ліцензія на використання необмеженого числа копій включена до ліцензії сервера.

MetaTrader 4 Data Center – являє собою спеціалізований проксі-сервер, призначений для підвищення масштабованості і безпеки платформи, і може бути проміжною ланкою між серверною частиною і клієнтськими терміналами. Використання не обов'язково, але стає доцільним для локальних мереж з декількома клієнтами, дозволяючи мінімізувати трафік котирувань і їх історії. Може встановлюватися на одному з робочих місць. Так само може використовуватися для зменшення навантаження на сервер. В цьому випадку рекомендується використання на виділеному комп'ютері і для клієнтських терміналів він стає альтернативним сервером. Платформа Windows Vista / XP / 2000. Ліцензування не потрібно.

MetaTrader 4 Client Terminal – клієнтська частина, встановлюється на комп'ютері трейдера. Призначена для проведення торгових операцій і технічного аналізу в режимі реального часу. Кілька типів ордерів дозволяє віддавати розпорядження на проведення операцій негайно або з додатковими умовами (при досягненні передбачену ціну). Забезпечується перегляд поточних новин, трансляцію яких здійснює серверна частина комплексу. Внутрішній Сі-подібна мова програмування MQL4 дозволяє запрограмувати торгові стратегії, індикатори, сигнали. Є можливість забезпечити повністю автоматичну торгівлю, коли програма-радник не лише виводить зображення і сигнали, але і посилає команди на відкриття/закриття угод. Використання можливості програмування не обов'язково – є 50 базових індикаторів, кожен з яких можна додатково підлаштовувати. Клієнтський термінал забезпечує роботу з будь-яким MetaTrader 4 Server без прив'язки до попередньо встановлених адресами. Платформа Windows 10/8/7/Vista / XP / 2000/98. Ліцензування не потрібно.

MetaTrader 4 для iPhone – управління торговим рахунком зі смартфонів на базі Apple iOS версії 7.0 і вище, підтримка iPhone, iPad і iPod Touch. Безкоштовна установка з AppStore. Підключення можливо до торгових серверів тільки тих брокерів, які оплатили розробникам відповідну послугу. Програма включає 30 технічних індикаторів, всі види ордерів, звукові сигнали (Alerts).

MetaTrader 4 для Android – управління торговим рахунком зі смартфонів на базі Android версії 4.0 і вище. Безкоштовна установка з Google Play. Підключення можливо до торгових серверів тільки тих брокерів, які оплатили розробникам відповідну послугу. Програма включає всі види ордерів, підтримку графіків 3 типів, 9 таймфреймів, підтримку планшетів.

MetaTrader 4 Mobile – управління торговим рахунком за допомогою мобільних пристроїв на платформі Windows Mobile (Pocket PC 2002 / Mobile 2003). Без ліцензії можлива робота тільки з деякими серверами. Для кожного сервера його власники можуть замовити у розробника окрему версію програми, яка не має можливості змінити сервер. Клієнт самостійно може купити ліцензію, яка дозволяє працювати з будь-яким MetaTrader 4 Server без прив'язки до попередньо встановлених адресами. Компанія MetaQuotes припинила підтримку MetaTrader 4 Mobile для ОС Windows Mobile і Windows Mobile SE з 26 червня 2014 року Платформа не вдається з'єднатися з торговим серверів останніх версій.

Веб-платформа MetaTrader 4 – торгівля через браузер. Підтримує всі стандартні функції торгівлі та 30 вбудованих індикаторів. Використання програм, написаних на MQL4, і пов'язаних з ними функцій, не підтримується.

Платформа орієнтована на маржинальний торгівлю. Можлива торгівля CFD. Але платформа не призначена для повномасштабної роботи на фондовому ринку або для проведення операцій без маржинальних умов:

- немає можливості виставити власну заявку в ринок, щоб її побачили інші торговці;
- немає можливості подивитися список існуючих на ринку заявок;
- немає механізмів роботи з опціонами;

- немає можливості підключити додаткове джерело котирувань і новин;
- немає механізмів роботи в національній валюті (звіт на клієнтському терміналі завжди формується на англійській мові із зазначенням USD в якості валюти).

3. 1. 2. Опис інтерфейсу клієнтського терміналу MT4

MetaTrader 4 Client Terminal – стандартний додаток Win32 призначений для клієнтів брокерських компаній.

У клієнтському терміналі є вбудовані засоби, що дозволяють проводити технічний аналіз ринку і здійснювати ручне управління торгівлею. Для аналізу ринку можуть застосовуватися технічні індикатори та різні лінійні інструменти – лінії підтримки/опору, трендові канали, рівні Фібоначчі тощо.

Для ручного управління торгівлею використовується панель управління ордерами. За допомогою цієї панелі трейдер може відкривати, закривати і модифікувати ордери. Крім того, в клієнтському терміналі є можливість автоматично управляти положенням стоп-наказів ринкових ордерів. В результаті впливу трейдера на вбудовані засоби управління торгівлею в клієнтському терміналі формуються торгові накази (ордери), які відправляються на сервер.

Інтерфейс терміналу містить всі необхідні функції для повноцінної торгівлі CFD (рисунок 3.1).

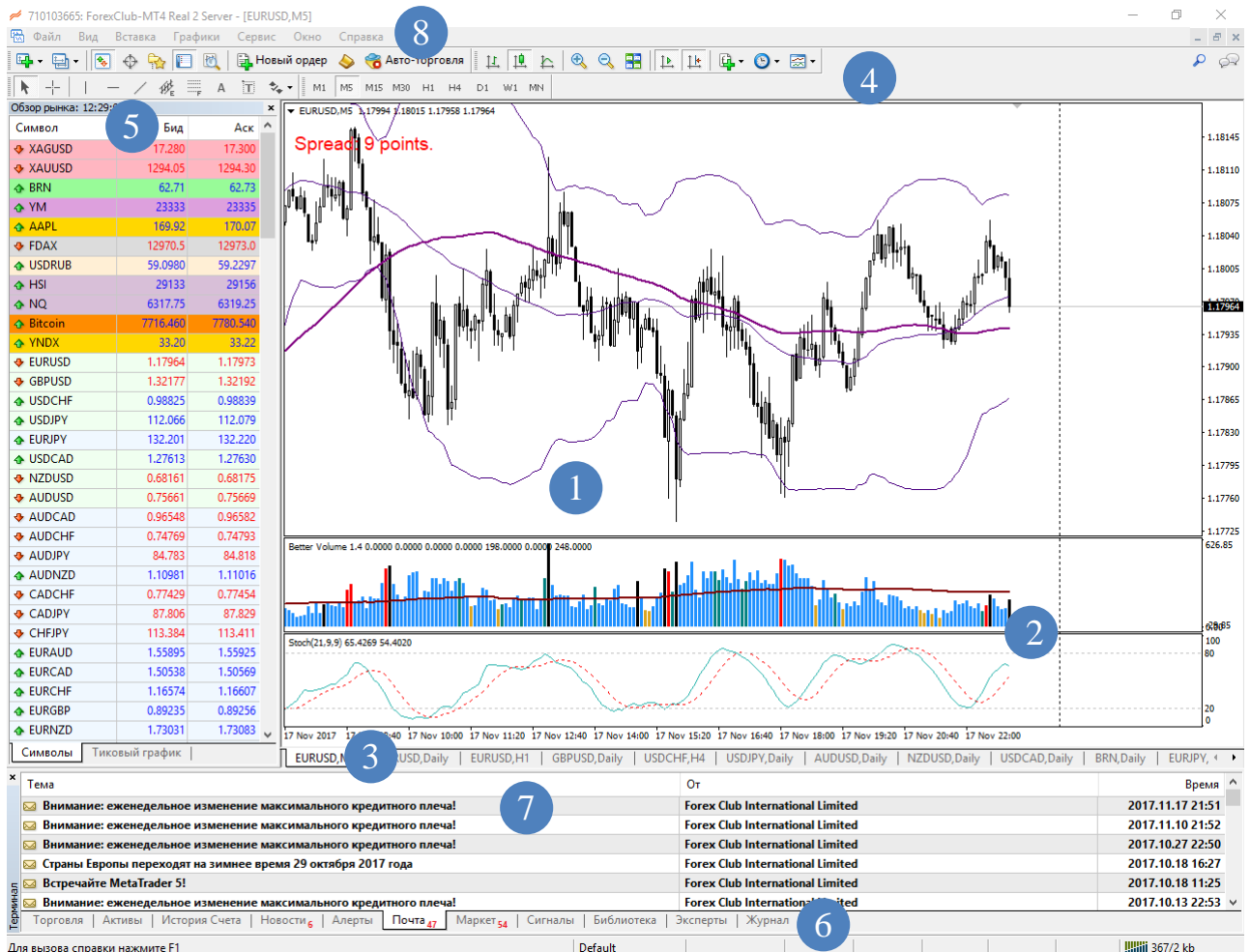


Рис. 3.1. Интерфейс клієнтського терміналу MT4

Під кожним номером відзначені наступні елементи:

1. Вікно графіка відкритого торгового інструменту з підключеними графічними індикаторами.
2. Додаткові вікна технічних індикаторів, пов'язаних з основним графіком.
3. Перелік винесених торгових інструментів.
4. Панелі налаштувань основного графіка (таймфрейм, відображення), інструментарію технічного аналізу (горизонтальні і довольні лінії, канали, підписи) і торгів.
5. Огляд котирувань всіх або певної кількості торгових інструментів.
6. Панель огляду відкритих торгових позицій і виставлених ордерів, пошти, стану рахунку, доступних зовнішніх торгових сигналів і т. Д.

7. Вікно огляду відкритої вкладки панелі 6.

Більшість елементів також мають додаткові поп вікна, що містять додаткові специфічні настройки. Наприклад, при натисканні ПКМ на вікно індикатора, користувач зможе видалити даний індикатор і відкрити його налаштування.

3. 1. 3. Опис MT4 Editor

MetaEditor – це багатофункціональний спеціалізований редактор, призначений для створення, редагування та компіляції прикладних програм, написаних на мові MQL4. Редактор має дружній інтерфейс, що дозволяє користувачеві легко орієнтуватися при написанні і налагодженні програм.

Клієнтський термінал розрізняє види програм за місцем їх розташування в підлеглих директоріях.

Всі прикладні програми зосереджені в директорії *Каталог_термінала\experts*. Експерти, скрипти і призначені для користувача індикатори, які трейдер передбачає використовувати в практичній роботі, повинні перебувати в своїх директоріях (рисунок 3.2). Експерти знаходяться безпосередньо в каталозі *Каталог_термінала experts*, а скрипти і індикатори - в піддиректоріях *Каталог_термінала\experts\scripts* і *Каталог_термінала\experts\indicators*.

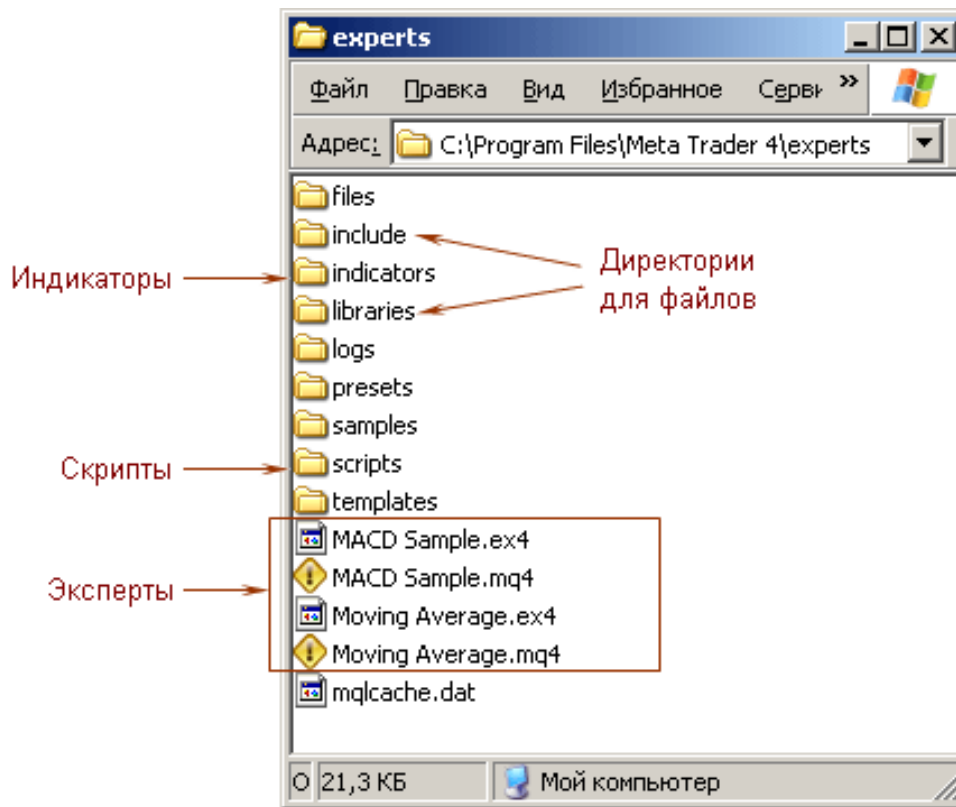


Рис. 3.2. Директорії для зберігання файлів, створених користувачем

Користувач може на свій розсуд створювати і інші директорії і зберігати в них деякі свої файли, однак використання готових програм, які перебувають в таких директоріях, в клієнтському терміналі не передбачено.

У мові MQL4 розрізняють 3 типи файлів, які несуть програмний код: mq4, ex4 і mqh.

Файли типу mq4 – це вихідний код програми. У файлах цього типу містяться вихідні тексти програм усіх видів (експертів, скриптів і індикаторів). Для створення кодів програм використовується редактор MetaEditor. Після того як код програми складено повністю або частково, його можна зберегти, а згодом – знову відкрити в редакторі для доопрацювання. Файли типу mq4 не можуть бути використані для виконання в клієнтському терміналі. Для того щоб запустити програму на виконання, її необхідно попередньо скомпілювати. В результаті компіляції вихідного коду програми створюється однойменний файл з розширенням ex4.

Файли типу `ex4` – це скомпільована програма, готова для практичного використання в клієнтському терміналі. Для файлів цього типу можливість редагування не передбачена. Якщо потрібно внести зміни в програму, то необхідно звернутися до її вихідного коду (файлу типу `mq4`), відредагувати його, а потім знову скомпільувати. За назвою файлу `ex4` не можна судити про те, до якого виду програм він відноситься – чи є він скриптом, експертом або індикатором. Файли `ex4` можуть використовуватися як бібліотечні.

Файли типу `mqh` – файли, що включаються. Це вихідний текст часто використовуваних блоків призначених для користувача програм. Такі файли можуть включатися в вихідні тексти експертів, скриптів і призначених для користувача індикаторів на етапі компіляції. Зазвичай файли, що включаються містять описи імпортованих функцій (дивіться в якості прикладу файли `stdlib.Mqh` або `WinUser32.mqh`), або опису загальних констант і змінних (`stderror.mqh` і `WinUser32.mqh`). Файли типу `mqh`, як правило, зберігаються в директорії *Каталог_термінала\experts\include*.

Зазвичай їх "підключають" на етапі компіляції до основного вихідного файлу за допомогою директиви `#include`. Незважаючи на те що файли типу `mqh` можуть містити вихідний код програми і MetaEditor їх компілює, вони не є самостійними і самодостатніми, тобто не вимагають компіляції з метою отримання виконуваних файлів типу `ex4`. У якості підключаємих файлів можуть також використовуватися файли `mq4`, які повинні зберігатися в каталозі *Каталог_термінала\experts\include*.

У навігаторі клієнтського терміналу в розділах "Радники", "Користувальницькі індикатори" і "Скрипти" висвічуються тільки ті імена файлів, які мають розширення `ex4` і при цьому перебувають у відповідному каталозі. Файли, скомпільовані в старішій версії MetaEditor, не запускаються і працювати не будуть.

Існують і інші види файлів, що не утворюють закінчену програму, але використовувані при створенні прикладних програм. Наприклад, програму можна скласти з декількох окремих файлів або використовувати для цієї мети

раніше створену бібліотеку. Користувач може створювати бібліотеки для використання функцій, призначених для зберігання часто використовуваних блоків призначених для користувача програм. Бібліотеки рекомендується зберігати в директорії *Каталог_терміналу\experts\libraries*. Як бібліотечні, можуть бути використані файли mq4 і ex4. Бібліотеки не можуть самостійно запускатися на виконання. Використання підключених файлів більш переважно, ніж використання бібліотек, через додаткові витрати комп'ютерних ресурсів при виклику бібліотечних функцій.

3. 1. 4. Опис програмних можливостей мови MQL 4

Мова програмування MQL 4 – це процедурна Сі-подібна мова, що містить спеціалізовані можливості, що спрощують написання прикладних програм, які виконуються з-під торгового терміналу, для автоматизації торговельної діяльності. Вона містить стандартні для Сі типи даних і має можливість написання призначених для користувача функцій. Всі програми мають вбудований джерело даних у вигляді чисельного уявлення графіка зміни ціни того активу, для якого дана програма була запущена. Для того, щоб зрозуміти можливості торгової системи, розглянемо схему взаємодії програми на MQL 4 з ринком (рисунок 3.3).

У клієнтському терміналі є інформаційне середовище – набір параметрів, які інформують про стан ринку і відносини між трейдером і дилінговим центром. Вона містить відомості про поточні ціни, обмеження щодо мінімального і максимального розміру ордера, мінімальної дистанції стоп-наказів, дозвіл / заборону здійснювати автоматичну торгівлю і безліч інших корисних параметрів, що характеризують поточну ситуацію. Інформаційне середовище оновлюється з надходженням в термінал нових тиків (зелена стрілка на рисунку 3.3).

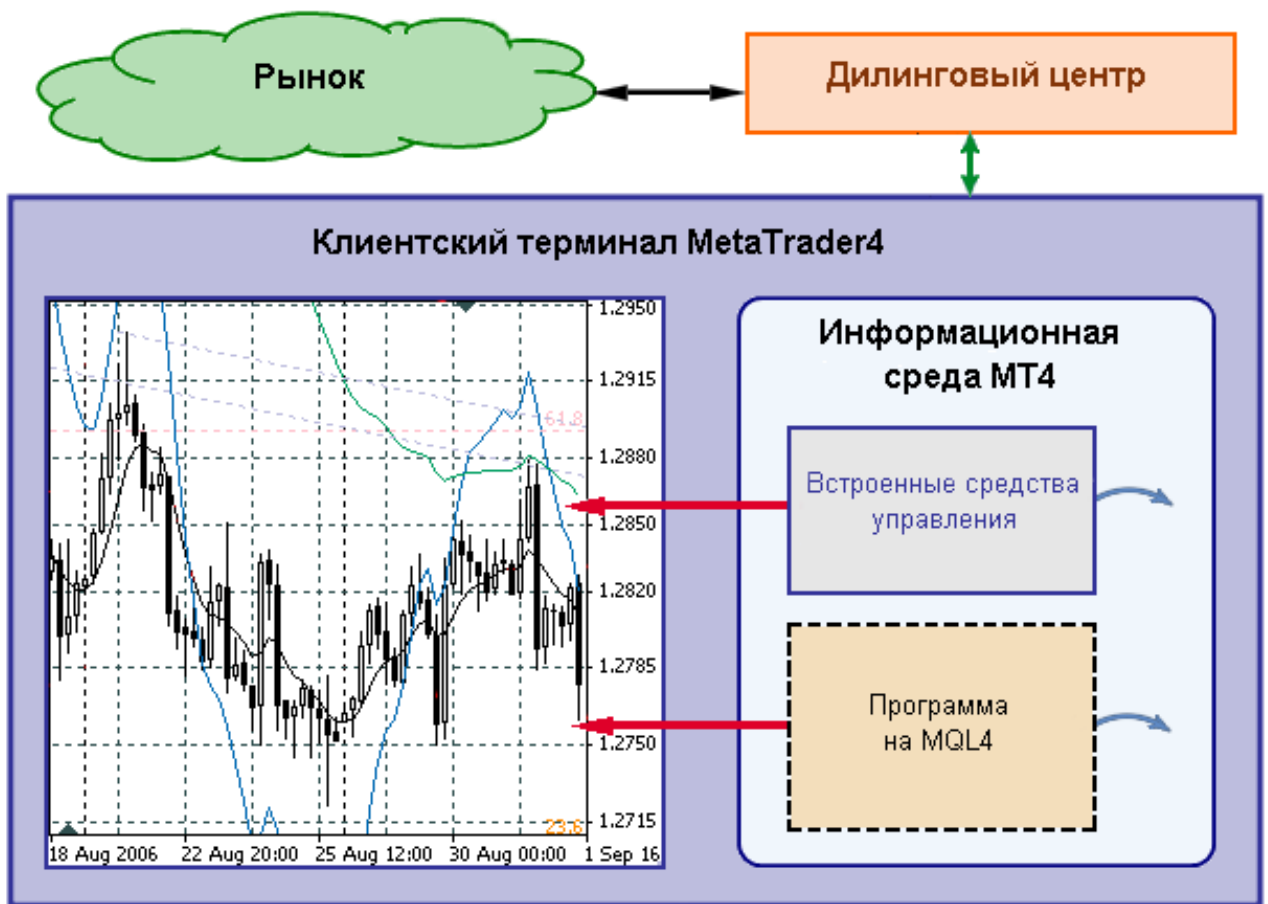


Рис. 3.3. Програма на MQL4 в складі клієнтського терміналу MetaTrader 4

MT4 розрізняє три види прикладних програм, що створюються на мові MQL4 і призначених для роботи в клієнтському терміналі:

- призначений для користувача індикатор – програма для графічного відображення будь-яких закономірностей на ринку, написана відповідно до авторським алгоритмом;
- експерт (радник, Expert Advisor) – програма, що дозволяє механізувати значну частину торгових операцій або повністю автоматизувати торгівлю;
- скрипт – програма для виконання одноразових дій, в тому числі – здійснення торгових операцій.

На рисунку 3.3. видно, що прикладна програма має ті ж засоби доступу до інформаційного середовища клієнтського терміналу, що і вбудовані засоби для ручної торгівлі (сині стрілки), а також може формувати керуючі накази (червоні

стрілки), що відправляються в клієнтський термінал. Програми різних видів можуть використовуватися одночасно і при цьому обмінюватися даними. За допомогою цих додатків програміст може механізувати значну частину торгових операцій або створити робот, здатний здійснювати торгівлю без втручання трейдера.

Прикладні програми і ручні засоби управління можуть використовуватися в клієнтському терміналі одночасно, взаємно доповнюючи один одного.

Основна технічна особливість торгівлі за допомогою інформаційно-торговельної системи MetaTrader полягає в тому, що всі дії, що керують рахунком, формуються в клієнтському терміналі і відправляються на сервер. Прикладні програми (експерт, скрипт, індикатор) можуть працювати тільки в складі клієнтського терміналу за умови його підключення до сервера (дилінгового центру). На сервері ніякі прикладні програми не встановлюються.

Сервер дозволяє лише обробляти накази, що надходять з боку клієнтського терміналу. Якщо клієнтський термінал відключений від Інтернету або запущена в ньому прикладна програма (експерт або скрипт) з якихось причин не генерує управляючі дії, то на сервері нічого відбуватися не буде.

Програми для автоматичного трейдингу мають набагато більше потенційних можливостей, ніж ручні засоби управління торгівлею.

У переважній більшості випадків програма дозволяє істотно полегшити працю трейдера, позбавивши його від необхідності безперервно, протягом тривалого терміну, перебувати біля монітора для спостереження за розвитком ринку. Вона також може допомогти знизити нервову навантаження і зменшити кількість помилок, що виникають в період сильного емоційного напруження. Але головне – застосування програмного способу управління торгівлею дозволяє розробити власні ідеї та перевірити їх на історичних даних, підібрати оптимальні параметри для застосування цих ідей, а в кінцевому підсумку – реалізувати на практиці продуману торгову стратегію.

3. 2. Загальні відомості про управління торговим рахунком

Торгівельна діяльність має на увазі не тільки аналіз фінансових інструментів, але і пошук точок входу для укладення потенційно вигідної для трейдера угоди. Кожен трейдер має справу з такими поняттями як капітал, депозит, ризик, час, обсяг. Від того, наскільки грамотно він оцінює свої можливості, розпоряджається капіталом, знає специфіку торгівлі в різний час доби і правильно обчислює обсяг однієї позиції на ринку, залежить його успіх на тривалому періоді часу. Всі ці завдання укладаються в три парадигми торгової системи (в даному контексті, торгова система – це не програма для автоматизації, а звід правил і принципів, за якими трейдер здійснює торгівлю):

- управління капіталом;
- управління ризиком;
- керування часом.

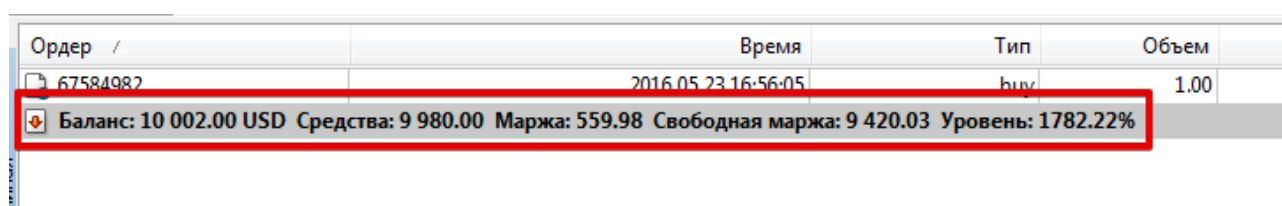
У світі, в тому числі і в СНД, використовуються поняття «money management», «risk management» і «time management». Торговий робот, який повинен полегшити вирішення всіх цих завдань, зобов'язаний містити в собі функції і настройки з управління коштами торгового рахунку, визначення обсягу однієї позиції і вибору оптимального часу для торгівлі. Тому, перш ніж перейти до розгляду структури автоматизованої торгової системи, необхідно розглянути дані парадигми і технічні терміни, пов'язані з ними:

- баланс;
- засоби;
- лот;
- обсяг;
- маржа;
- Margin Call і Stop Out
- ордер та їх типи;
- позиція;
- валютний SWAP.

3. 2. 1. Поняття балансу та вільних коштів

Баланс – це показник, який відображає суму грошових коштів на торговому рахунку трейдера. Особливістю показника «Баланс» є те, що він не враховує прибуток або збиток за відкритими операціями, а фіксує лише наявні кошти, а також плюс і мінус за закритими позиціями (рисунок 3.4).

Наприклад, на рахунку 10 000 доларів. Було закрито дві угоди. Одна з прибутком 500 доларів, інша зі збитком 498 доларів. Таким чином, баланс становить $10\,000 + 500 - 498 = 10\,002$ долара.



Ордер /	Время	Тип	Объем
67584082	2016.05.23 16:56:05	buy	1.00
Баланс: 10 002.00 USD Средства: 9 980.00 Маржа: 559.98 Свободная маржа: 9 420.03 Уровень: 1782.22%			

Рис. 3.4. Показники стану торговельного рахунку

Показник «Кошти», на відміну від показника «Баланс», враховує поточні результати за відкритими операціями. Іншими словами, «Кошти» – це гроші, які будуть на балансі, якщо закрити всі відкриті угоди.

Наприклад. Баланс становить 10 002 долара. Відкрита операція на покупку в парі EUR/USD. Поточний збиток за угодою становить 22 долари. Відповідно, показник «Засоби» дорівнює $10\,002 - 22 = 9\,980$ долара. Тобто, якщо трейдер закриє угоду прямо зараз, зі збитками в 22 долара, сума 9 980 доларів з показника «Кошти» перейде вже в «Баланс».

3. 2. 2. Визначення термінів «Лот» і «обсяг»

Кожна торгова операція на ринку FOREX вимірюється в лотах – це одне з основних понять і важливо вміти правильно розрахувати цей показник для торгівлі.

Лот – стандартна одиниця вимірювання обсягу валютної операції. Один лот дорівнює 100 000 базовим одиницям валюти. Наприклад, у валютній парі USD/JPY базовою валютою є долар США, отже, один лот валютного контракту

по цій парі буде дорівнює 100 тис. доларів. Обсяг угоди може вимірюватися цілим і дробовим числом лотів.

З визначення зрозуміло, що цілими лотами торгувати можуть або великі спекулянти, або фінансові інститути. Для виходу на ринок FOREX приватним особам, брокери і популярні дилінгові центри надають можливість використовувати різні варіанти стандартного лота:

- Стандартний лот (цілий лот) містить 100 000 одиниць базової валюти, його обсяг дорівнює 1;
- Міні лот містить 10 000 одиниць базової валюти і його обсяг дорівнює 0,1 від стандартного (цілого) лота;
- Мікро лот містить 1000 одиниць базової валюти і його обсяг дорівнює 0,01 від стандартного лота.

Обсяг показує, скільки коштує один пункт руху ціни брокеру або трейдеру, в залежності від того, куди рухається ціна (відповідно до прогнозу трейдера чи проти нього). Розглянемо приклад відкриття позиції об'ємом в 0.01. У цьому випадку один пункт буде дорівнювати десяти центам, тобто, купивши валютний інструмент EURUSD по 1,3245 і продавши його за 1,3255, дохід трейдера складе 10 пунктів. При зазначеному обсязі лота прибуток складе 1 долар. При лоті, об'ємом 0,1, один пункт уже буде дорівнювати одному долару, а 10 пунктів – 10 долларам.

Підведемо підсумок:

- Стандартний лот (обсяг 1) – зміна ціни валюти на 1 пункт дорівнюватиме 10 долларам.
- Міні лот (обсяг 0.1) – зміна ціни валюти на 1 пункт дорівнюватиме 1 долару.
- Мікро лот (обсяг 0.01) – зміна ціни валюти на 1 пункт дорівнюватиме 10 центам.

3. 2. 3. Поняття маржи, вільної маржи та «рівень»

Як відомо, легку доступність торгівлі на валютному ринку FOREX забезпечує кредитне плече, що дозволяє сплачувати трейдеру лише частину суми, необхідної для відкриття ордера. Показник «Маржа» саме відображає цю суму. Розмір застави залежить від величини використовуваного кредитного плеча. Чим більше плече, тим менше сума застави. При кредитному плечі 1:10 розмір застави для відкриття операції складе 10% від необхідної суми, при плечі 1:100 застава складе вже 1%.

Наприклад, трейдер відкриває угоду на покупку в парі євро/долар обсягом 1 стандартний лот. Як відомо, стандартний лот – це 100 000 одиниць валюти. Тобто, трейдер купує 100 000 євро за долари. Поточна ціна - 1,11995. Таким чином, для відкриття операції необхідно 111 995 доларів. Кредитне плече на торговому рахунку дорівнює 1:200, тобто, для відкриття угоди трейдеру необхідно $111\,995/200 = 559,98$ доларів, що і відображається в показнику «Маржа» (результат заокруглений з 559,975).

Показник «Вільна маржа» відображає кількість коштів на рахунку, доступних для відкриття угод. Розраховується він просто: Вільно = Гроші - Застава (Маржа).

Відповідно, в наведеному прикладі, показник «Вільно» становить 9 980 (Засоби) - 559,98 (Застава) = 9 420,03 (результат заокруглений, оскільки насправді заставу складає 559,975 доларів).

Показник «Рівень» найважливіший з усіх вищенаведених. Він відображає співвідношення показника «Засоби» до показника «Маржа», вираженого у відсотках.

Повернемося до прикладу. Показник «Рівень» буде дорівнює 9 980 (Засоби): $559,98$ (Застава) $\times 100\% = 1782,22\%$.

Показник «Рівень» безпосередньо пов'язаний зі стійкістю відкритих угод. Чому він такий важливий? Звичайна математика. Якщо збиток по одній або кільком угодам зростає, відповідно, в чисельнику значення показника «Кошти» зменшується, а знаменник «Маржа» залишається без змін.

FOREX брокери і ДЦ в своїх регламентах торгівлі, вказують граничне значення показника «Рівень», при досягненні якого відкриті угоди будуть закриті примусово. Це носить назву «Stop Out».

3. 2. 4. Margin Call та Stop Out

Термін Margin Call прийшов на валютний ринок FOREX з фондового ринку. Дослівно він переводиться як «дзвінок через проблеми із заставою (маржею)». Коли рівень маржи нижче 100%, брокери, як правило, інформують клієнта про те, що вільних коштів на підтримку відкритих позицій вже не вистачає. Якщо рівень маржи дійде до зазначеного брокером граничного значення (наприклад 25% або 50%), відкриті позиції будуть закриті примусово, а збитки будуть відображені на балансі трейдера. Це і є Stop Out.

Поняття Margin Call нерозривно пов'язане з поняттям кредитного плеча.

Наприклад:

- депозит 5000 \$;
- кредитне плече 1:100, яке надав брокер;
- загальний розмір депозиту склав 500 000 \$, з яких 5000 все також належать трейдеру і 495 000 – брокеру;
- відкрита угода із заставою тисячі \$;
- в результаті кошти (equity) на рахунку складають $5000 - 1000 = 4000$ \$;
- стандартний лот на FOREX становить 100 000 одиниць валюти. У цьому випадку, з них 1000 доларів належить трейдеру, 99000 – гроші брокера, надані в рамках кредитного плеча.

Припустимо, по відкритій угоді ціна йде не в той бік і збиток починає зростати. Трейдер не використовував Stop Loss, тому мінус продовжує збільшуватися. Margin Call настане, коли сума на рахунку стане дорівнювати або буде менша за маржу, зокрема – 1000 \$. Якщо у цифрах:

5000 \$ (початкова сума) + $495\ 000$ \$ (кредитне плече) - 4000 \$ (збиток) = $496\ 000$ \$ (при закритті рахунку).

З них 495 тисяч повертається брокеру, а 1 тис. залишається на рахунку. Брокер нічого не втрачає при збитковій діяльності свого трейдера, так само як, і нічого не заробляє, крім комісії. Всі ризики лягають виключно на трейдера. Треба зауважити, що трапляється ситуація, коли сума на рахунку залишається менше розрахованої. Це відбувається через різкі коливання ринку. Коли Stop Out спрацював, збиток зафіксований, але за ці мілісекунди ціна все ж встигла піти ще далі.

3.2.5. Ордер, типи виконання ордерів

Існує три типи виконання ордерів на ринку: «негайне виконання» (instant execution), «ринкове виконання» (market execution) та «відкладене виконання»:

- При відправці запиту на негайне виконання, термінал автоматично підставляє в ордер поточні ціни. У разі, якщо брокер приймає ціни, ордер буде виконаний (рисунок 3.5). Якщо брокер не приймає запитану ціну, то відбувається так зване "Перекотування" (Requote) – брокер повертає ціни, за якими може бути виконаний такий ордер (рисунок 3.6).

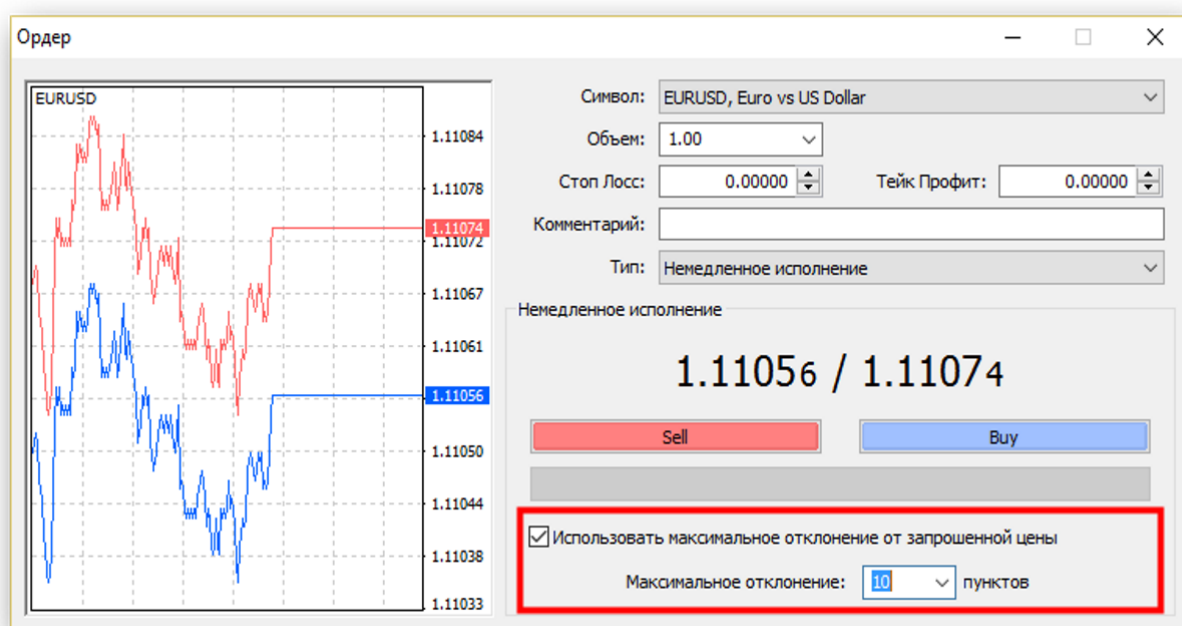


Рис. 3.5. Налаштування ордеру при негайному виконанні

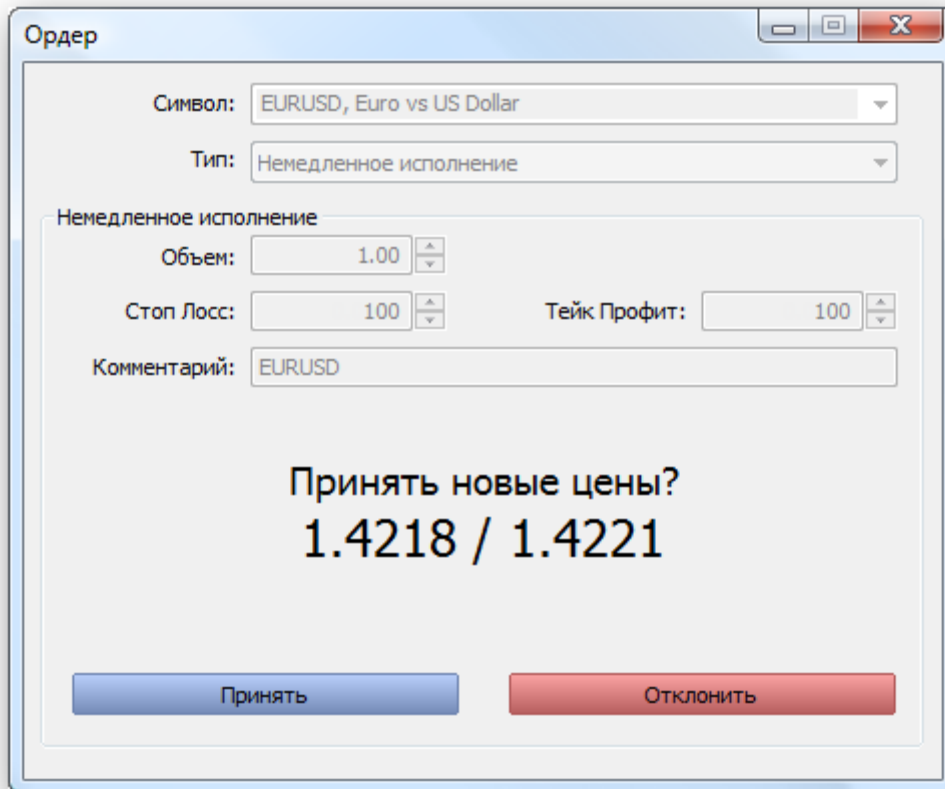


Рис. 3.6. Перекотування, якщо брокер повернув ціни

- В режимі ринкового виконання рішення про ціну виконання приймає брокер без додаткового узгодження з трейдером. Відправка ринкового ордера в такому режимі має на увазі дострокову згоду з ціною, за якою він буде виконаний.

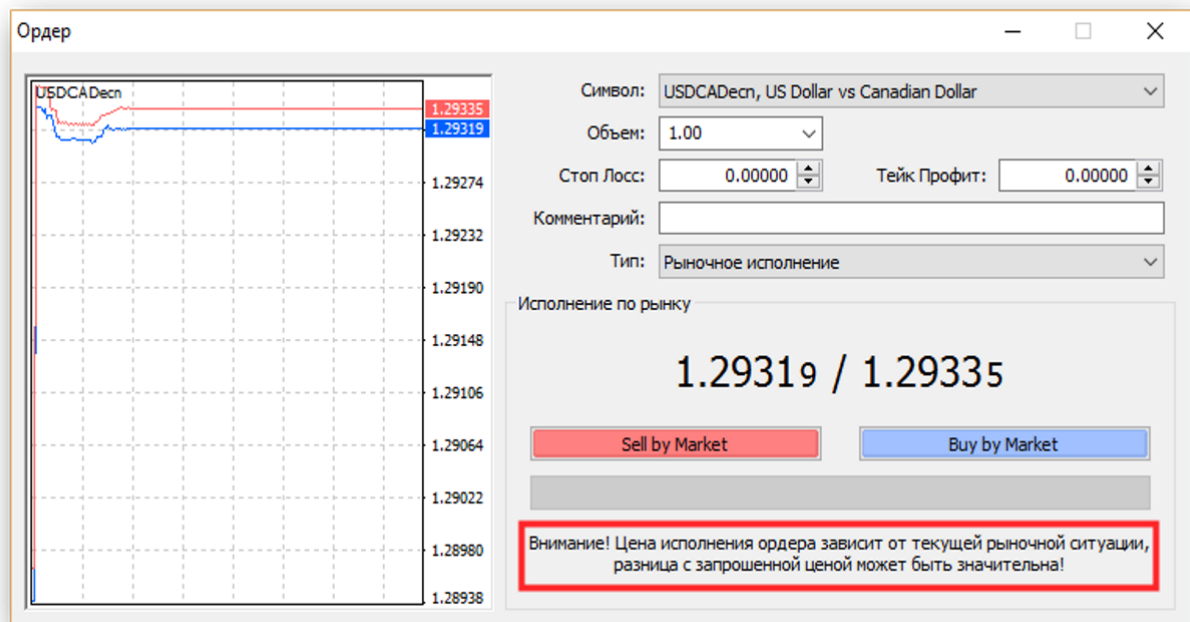


Рис. 3.7. Вікно відкриття ордеру з ринковим виконанням

- Відкладений ордер – це ордер, який буде виконаний у майбутньому за ціною, яку вказав клієнт, якщо вона дійде до значень зазначених в ордері. Наприклад, котирування EUR/USD знаходяться на рівні 1.20000, а трейдер хоче відкрити ордер на продаж за ціною 1.20550. Він може виставити відкладений ордер на цю ціну і якщо вона, дійде до значення 1.20550, цей ордер буде виконаний.

3. 2. 6. Валютний SWAP

SWAP – це валютна операція, що поєднує в собі покупку і продаж двох контрактів з різними датами валютування на одних і тих же умовах.

Мета угоди swap – продовжити наявну відкриту позицію і при цьому уникнути реального зарахування на рахунок купленої валюти по угоді. Таким чином, кожен раз відкриваючи нову угоду swap, ми продовжуємо дату зарахування на рахунок валюти.

Для перенесення відкритих позицій використовується swap типу TOM / NEXT:

- перенесення з понеділка на вівторок відбувається о 01:00 вівторка, відповідно ТОМ для вівторка – це середа, а NEXТ – четвер, тобто swar одинарний;
- перенесення з вівторка на середу проходить в 01:00 середи, відповідно ТОМ для середи – це четвер, а NEXТ – п'ятниця, тобто swar одинарний;
- перенесення із середи на четвер проходить о 01:00 четверга, відповідно ТОМ для четверга – це п'ятниця, а NEXТ – понеділка (субота і неділя – неробочі дні), тобто swar потрійний;
- перенесення з четверга на п'ятницю проходить о 01:00 п'ятниці, відповідно ТОМ для п'ятниці – це понеділок, а NEXТ – вівторок, тобто swar одинарний;
- перенесення з п'ятниці на понеділок проходить о 01:00 понеділка, відповідно ТОМ для понеділка – це вівторок, а NEXТ – середа, тобто swar одинарний.

3.3. Методика пошуку точок входу в ринок

Перш ніж розглядати алгоритм аналізу торгових інструментів, необхідно уточнити різницю між поняттями «точка входу» та «сигнал» в контексті даної дипломної роботи.

Торговий сигнал, як вже було сказано раніше, це інтерпретація індикаторів і самого графіка руху ціни, яка спонукає трейдера до відкриття позицій на ринку. Наприклад, перетин ліній індикатора Stochastic в зоні перепроданості передбачає, що трейдер повинен відкрити позицію на покупку. Такі сигнали не використовуються торговельною системою, як заклик до дії та ігноруються.

Точка входу – це сигнали від утворення патернів Price Action, які використовуються розробленим торговим роботом для відкриття позицій на ринку. Іншими словами, під час аналізу ринку, спочатку йде пошук патерна Price Action на деякій дистанції. Якщо такий був виявлений, аналізується

ринковий контекст і свідчення інших активних індикаторів для пошуку торгових сигналів від них.

3.3.1. Періодичність аналізу

При описі свічкового аналізу був відзначений той факт, що кожна свічка являє собою рух ціни за певний період – тобто інтервал часу, який вказаний в налаштуваннях графіка, іменованій як таймфрейм (від англ. Time frame).

Таймфрейми бувають декількох стандартних видів: однохвилинний, п'ятихвилинний, годинний, денний або тижневий і позначаються вони M1, M5, M15, H1, H4, D1, W1 і MN. Залежно від стилю торгівлі (скальпінг, короткострокова, середньострокова або довгострокова торгівля), в якості основного таймфрейма вибираються M5, H1, H4 або D1, наприклад. Це умовно, кожен трейдер вибирає таймфрейм самостійно та має можливість їх комбінувати. Однак в пункті 2.5.3. було наочно доведено, що статистично, чим старше таймфрейм, тим сильніше сигнали, які на ньому генеруються. Це пояснюється тим, що на молодших таймфреймах занадто багато ринкового шуму, за яким не видно дійсно важливих і сильних тенденцій та цінових рухів. Де-факто, стандартними таймфреймами є H1, H4 і D1, при чому D1 – найбільш визнаний і використовуваний серед великих гравців.

Оскільки в основі торгової системи лежить пошук і аналіз патернів Price Action, немає сенсу проводити аналіз торгового інструменту кожен раз, як оновлюються його котирування, так як система аналізує закінчені свічкові моделі і патерни. Тому аналіз проводиться в залежності від таймфрейму графіка, на якому запущений робот. Якщо таймфрейм дорівнює H4, то аналіз буде проводитися раз на 4 години на той момент, коли поточна свічка закінчила формуватися. Найкраще це зрозуміло на основі D1 – в цьому випадку, інструмент аналізується опівночі, за умови, що торговий термінал запущений.

Через те, що користувач може в момент поновлення періоду вимкнути свій термінал, аналіз інструменту відбувається і при запуску терміналу, проте в такому випадку, торгові сигнали можуть бути вже втрачені або неактуальні.

3.3.2. Торговий сигнал, як основний елемент аналізу

У попередньому розділі розглядалася проблема неоднозначної інтерпретації сигналів, коли один і той же індикатор може генерувати різні сигнали в одній певній ринкової ситуації. Звідси з'являється наступна проблема – необхідно не просто виявляти сигнали з різних джерел, а грамотно їх розпізнавати, щоб в результаті можна було визначити: це точка входу на покупку або на продаж. Саме тому основою для виявлення точки входу в ринок є утворення патерна Price Action і вже в момент його утворення використовуються сигнали інших індикаторів в якості підтвердження або спростування.

Кожен сигнал представлений у вигляді простого масиву з двох елементів, в якому міститься задана вага сигналу (описано далі в пункті 3.3.3) і тип (на продаж або на покупку). Сигнали на продаж в осередку типу містять значення «- 1», відповідно, сигнали на покупку містять значення «+1». При відкритті торгової позиції значення типу всіх виявлених сигналів складаються. Якщо сума більша нуля, відкривається позиція на покупку, якщо менше – на продаж.

У доповнення до цього, при підрахунку вагових коефіцієнтів, вага сигналів, які мають відмінний від точки входу тип сигналу, при обчисленні сили даної точки, віднімається, а не додається. В налаштуваннях можна включити опцію «Ігнорувати протилежні сигнали» (`ignore_enemy_signals`), тоді суперечливі сигнали не враховуватимуться.

3.3.3. Сила точки входу

У другому розділі були описані різні методи аналізу торгових інструментів: від побудови простих рівнів підтримки і опору до виявлення і використання в якості торгового сигналу патернів Price Action. Повертаючись до пункту 2.5.3. було наочно продемонстровано, що в залежності від різних умов, в яких проводиться оцінка графіка руху ціни, ефективність одних і тих же інструментів аналізу сильно відрізняється. Крім того, було доведено, що одні торговельні сигнали можуть підкріплюватися іншими – наприклад, відскік ціни

від рівня опору збігається з сигналом перетину ліній індикатора MACD в зоні перекупленості. В такому випадку, MACD є фільтром, який підтверджує або спростовує початковий торговий сигнал, а саме підхід і відскік ціни від лінії опору.

Звідси стає зрозуміло, що для більш точного входу в ринок потрібно грамотно розробити алгоритмічну систему, яка не просто буде визначати різні цінові рівні, тренди і виявляти сигнали. Алгоритм повинен містити механізм, який дозволить комбінувати кілька сигналів в один, фільтрувати суперечливі і, на основі заданих параметрів, виміряти «силу» точки входу, утворену патерном Price Action. Для цього був використаний механізм вагових коефіцієнтів.

Визначення вагового коефіцієнта наступне: це числовий коефіцієнт, параметр, що відображає значимість, відносну важливість, «вагу» даного чинника, показника в порівнянні з іншими чинниками, що впливають на досліджуваний процес.

З точки зору завдання цього дипломного проекту, кожен торговий сигнал від кожного індикатора, об'єкта чи патерна Price Action має свою «вагу», в залежності від таймфрейма. Під час аналізу підраховується вага всіх торгових сигналів, які були виявлені, потім множаться на коефіцієнт активного таймфрейма і на коефіцієнт пройденої відстані. У вигляді формули це можна представити таким чином:

$$S_s = \sum VS_i * \left(1 + \frac{D}{w/10}\right) * cT, \quad (3.1)$$

де $\sum VS_i$ – сума вагових коефіцієнтів;

D – діапазон пошуку сигналу (за замовчуванням 1000);

w – відстань від поточної ціни до ціни закриття свічки, яка утворила потенційну точку входу;

cT – коефіцієнт таймфрейма (для H1 дорівнює 2, H4 – 3, D1 – 4).

Вагові коефіцієнти для патернів Price Action обчислюються в діапазоні від 1 до 5, для інших джерел сигналів від 1 до 2,5. Для відкриття позиції необхідно, щоб сила точки входу пододала певний числовий поріг (за замовчуванням 700).

Нижче в таблиці 3.1. представлена вага торгових сигналів в залежності від таймфреймів H1, H4, D1 для рівнів П / С.

Таблиця 3.1.

Вага торгових сигналів для рівнів П/С в залежності від таймфрейма і ринкової ситуації.

Джерело сигналу	H1	H4	D1
Пробой по тренду	1.1	1.4	1.9
Отбой по тренду	1.2	1.45	2
Пробой против тренда	1.05	1.325	1.8
Отбой против тренда	1.15	1.425	1.95
Ложный пробой по тренду	1.225	1.475	2.025
Ложный пробой против тренда	1.3	1.5	2.2
Пробой зоны консолидации	1.125	1.435	1.925
Отбой от границ зоны консолидации	1.2	1.45	2
Ложный пробой границы зоны консолидации	1.25	1.485	2.25

Таким чином, були складені коефіцієнти і для інших використовуваних інструментів технічного і опціонного аналізу, а також для сигналів від патернів Price Action. Це дозволяє використовувати їх комбінації для визначення більш точного входу в ринок.

Наприклад, змодельюємо наступну ситуацію: утворився патерн «Пін-бар» поруч з рівнем опору вище ковзних середніх за 100 і 200 днів (висхідний тренд). При цьому значення підключеного індикатора MACD знаходяться в

зоні перекупленості, що в контексті утворення «Пін-бару» може вказувати на зміну тренда. У підсумку існує 3 торгових сигнали: утворення пін-бару, відбій проти тренда від рівня п/с і MACD в зоні перекупленості. Для цих сигналів було запропоновано такі коефіцієнти: 5, 1.95, 1.8. На момент виявлення сигналів, котирування пішли від ціни закриття на 480 пунктів. Робот запущений на таймфрейме D1.

Підставивши ці дані в формулу отримаємо:

$$Ss = (5 + 1.95 + 1.8) * 4 * \left(1 + \frac{1000}{480/10}\right) = 8.75 * 4 * 21.8 = 764 \quad (3.2)$$

Таким чином виходить, що сигнал достатньо сильний і робот відкриє угоду на продаж, так як до цієї точки входу підв'язаний тип позиції Sell.

3.4. Структура і компоненти торгової системи

Виходячи з пунктів 3.2 та 3.3, можна визначити 2 основні групи завдань, що стоять перед торговим роботом:

1. Оцінка торгового інструменту, для якого був запущений даний робот. Використання комбінацій різних видів технічного аналізу для пошуку точки входу в ринок.
2. Визначення поточного стану торгового рахунку для визначення потенційно можливого ризику при відкритті нових позицій, фіксації поточного прибутку або збитку.

На основі цих груп можна виділити структурні компоненти автоматизованої торгової системи (рисунок 3.4).

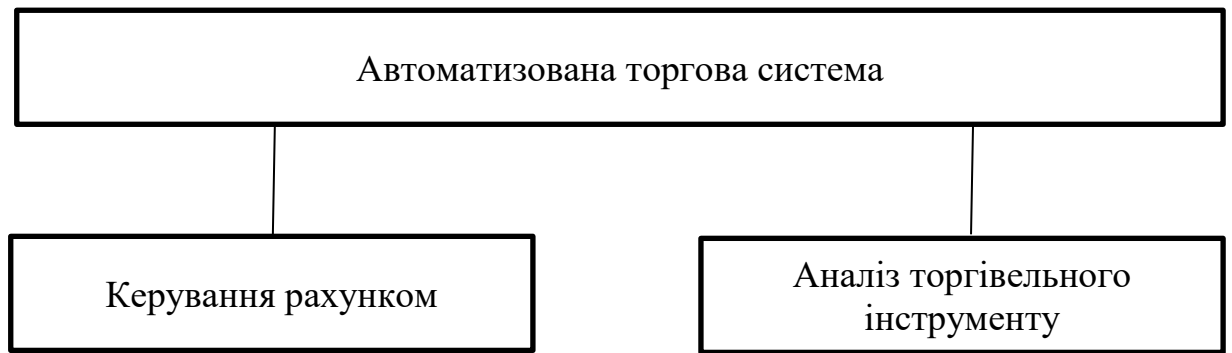
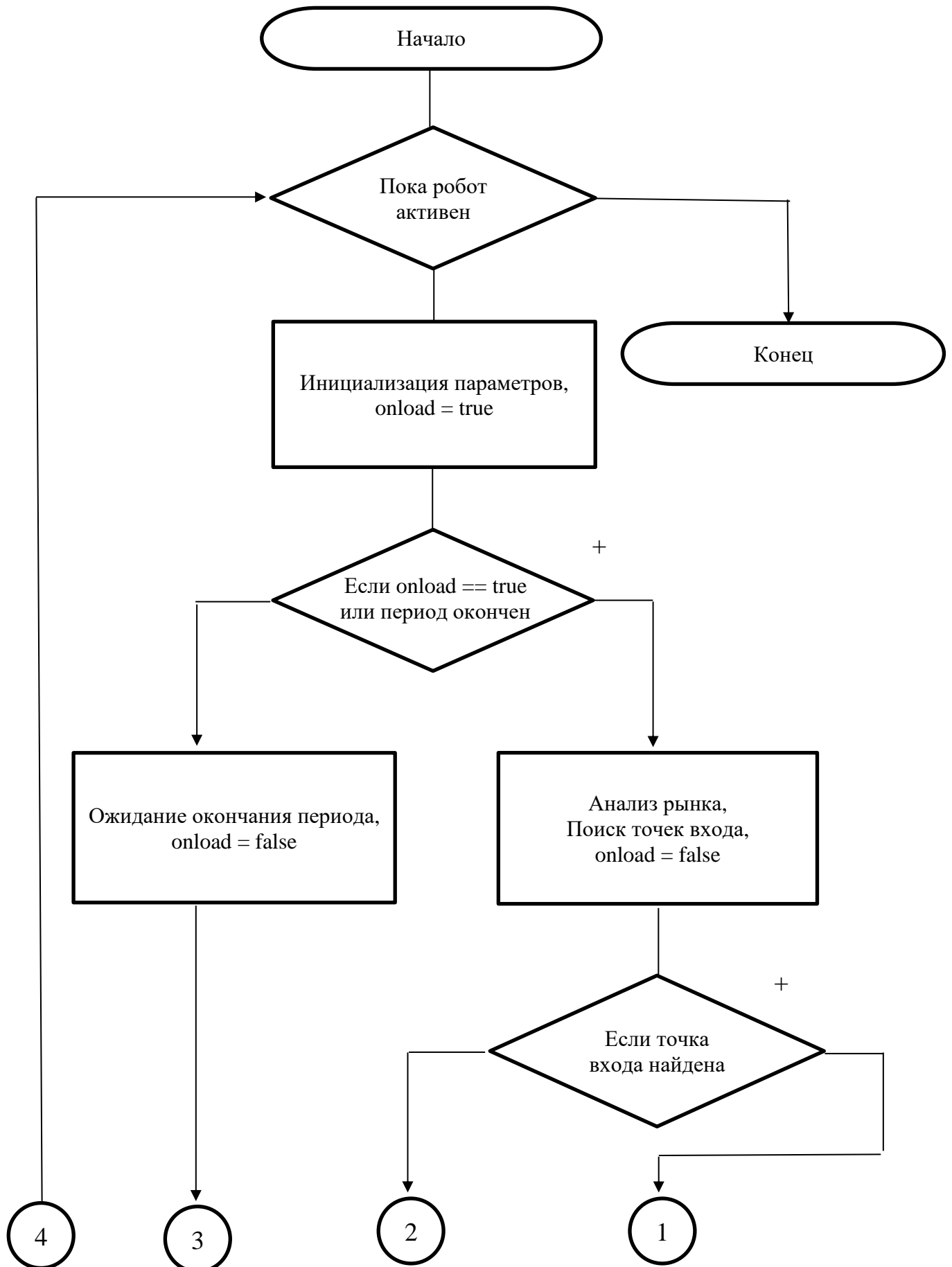


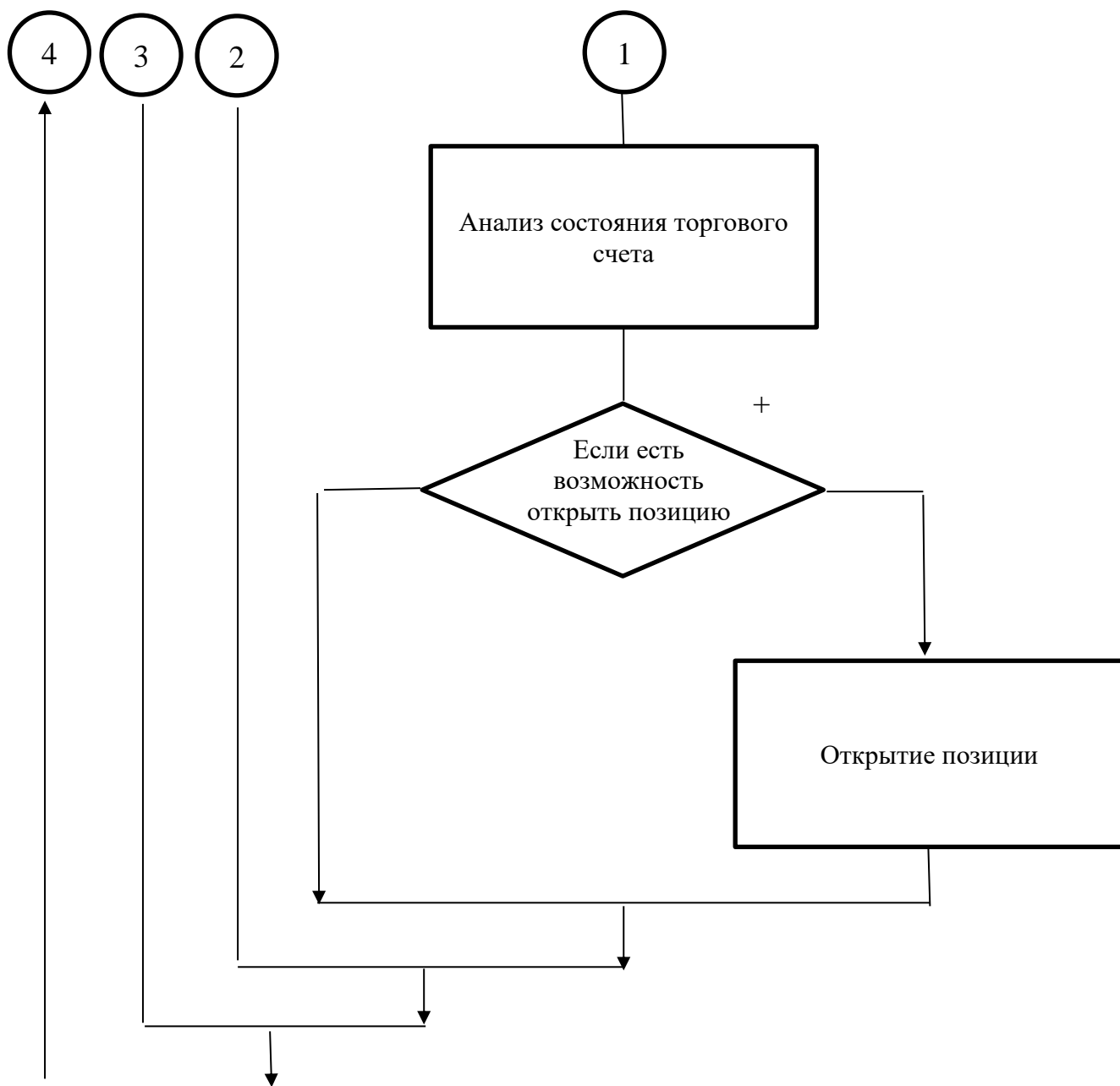
Рис. 3.4. Компоненти торгової системи

Оскільки, мова MQL4 процедурна, то блок-схема, що описує роботу і взаємодію компонентів автоматизованої торгової системи між собою в узагальненому вигляді, виглядає таким чином (блок-схема 3.1).

Блок-схема 3.1. Загальний алгоритм автоматизованої торгової системи з використанням пошуку патернів Price Action.



Продовження блок-схеми 3.1.



3.4.1. Реалізація компоненту аналізу торгового інструменту

Компонент аналізу торгового інструменту містить в собі набір попередньо інтегрованих і призначених для користувача функцій щодо здійснення пошуку точок входу в ринок і подальшого аналізу, в разі знаходження такої точки. З пункту 3.3. відомо, що під точкою входу мається на увазі торговий сигнал від утворення патерну Price Action.

Пошук патернів Price Action здійснюється від попередньої свічки в заданому в налаштуваннях діапазоні. Для визначення кожного простого патерну, що складається з однієї свічки, розроблена функція, яка порівнює параметри свічки відповідно до параметрів патерну. Кожна функція має тип bool і повертає true у випадку, якщо патерн був знайдений. Наприклад, наступний код визначає патерн «Молот», який є різновидом розворотних патернів типу «Пін-бар» і зустрічається в висхідному тренді (лістинг 3.1).

Лістинг 3.1. Функція визначення патерну «Молот»

```
bool IsHammer(int ai_0) {
    if (GetAllHeight(ai_0) >= 10.0 * Point &&
        GetUpperShadowHeight(ai_0) < GetAllHeight(ai_0) / 5.0 &&
        GetLowerShadowHeight(ai_0) > 2.0 * GetBodyHeight(ai_0) &&
        GetUpperShadowHeight(ai_0) > 2.0 * Point)
        if (IsHigher(ai_0)) return (TRUE);
    return (FALSE);
}
```

Для визначення параметрів свічок написані окремі функції, які повертають розмір тіла свічки, її верхню тінь, нижню тінь тощо. Приклади використання цих функцій можна побачити в лістингу 3.1. Нижче наведений приклад однієї з таких функцій, а саме GetUpperShadowHeight, що повертає висоту верхньої тіні свічі (лістинг 3.2).

Лістинг 3.2. Функція по обчисленню розміру верхньої тіні

```
double GetUpperShadowHeight(int ai_0) {
    return (MathAbs(High[ai_0] - MathMax(Close[ai_0],
        Open[ai_0])));
}
```

В цих функція єдиним параметром є `ai_0` – ідентифікатор переданої свічки.

Визначення більш складних паттернів, що складаються з двох свічок і більш, знаходиться в загальній функції `MMCFCondition`, де виконується виклик функцій по розпізнаванню простих паттернів. Наприклад, код, наведений в лістингу 3.3 визначає патерн «Модель поглинання».

Лістинг 3.3. Визначення патерну «Модель поглинання»

```
//Модель поглинання
if (!IsDoji(ai_0 + 2)) {
    if (IsYang(ai_0 + 2) != IsYang(ai_0 + 1)) {
        if (MathMax(Close[ai_0 + 1], Open[ai_0 + 1]) >
MathMax(Close[ai_0 + 2], Open[ai_0 + 2]) && MathMin(Close[ai_0 +
1], Open[ai_0 + 1]) < MathMin(Close[ai_0 + 2], Open[ai_0 +
2])) {
            if (IsLower(ai_0 + 2) || IsLower(ai_0 + 1) &&
IsYang(ai_0 + 1)) AddintoCFArray(gi_192, gi_152);
            if (IsHigher(ai_0 + 2) || IsHigher(ai_0 + 1) &&
IsYing(ai_0 + 1)) AddintoCFArray(gi_192, gi_156);
            if (GetBodyHeight(ai_0 + 2) >= 15.0 * Point ||
IsLower(ai_0 + 1) && IsYang(ai_0 + 1)) AddintoCFArray(gi_192,
gi_152);
            if (GetBodyHeight(ai_0 + 2) >= 15.0 * Point ||
IsHigher(ai_0 + 1) && IsYing(ai_0 + 1)) AddintoCFArray(gi_192,
gi_156);
        }
    }
}
```

Так як була необхідність перевірити розпізнавання паттернів на історії котирувань, всі виявлені патерни заносяться в масиви `gia_172` і `gia_176` за допомогою функції `AddintoCFArray`, які зберігають назву патерну і тип сигналу (купівля або продаж). В результаті роботи цієї функції виводяться всі потенційні сигнали від усіх виявлених паттернів (рисунок 3.8).

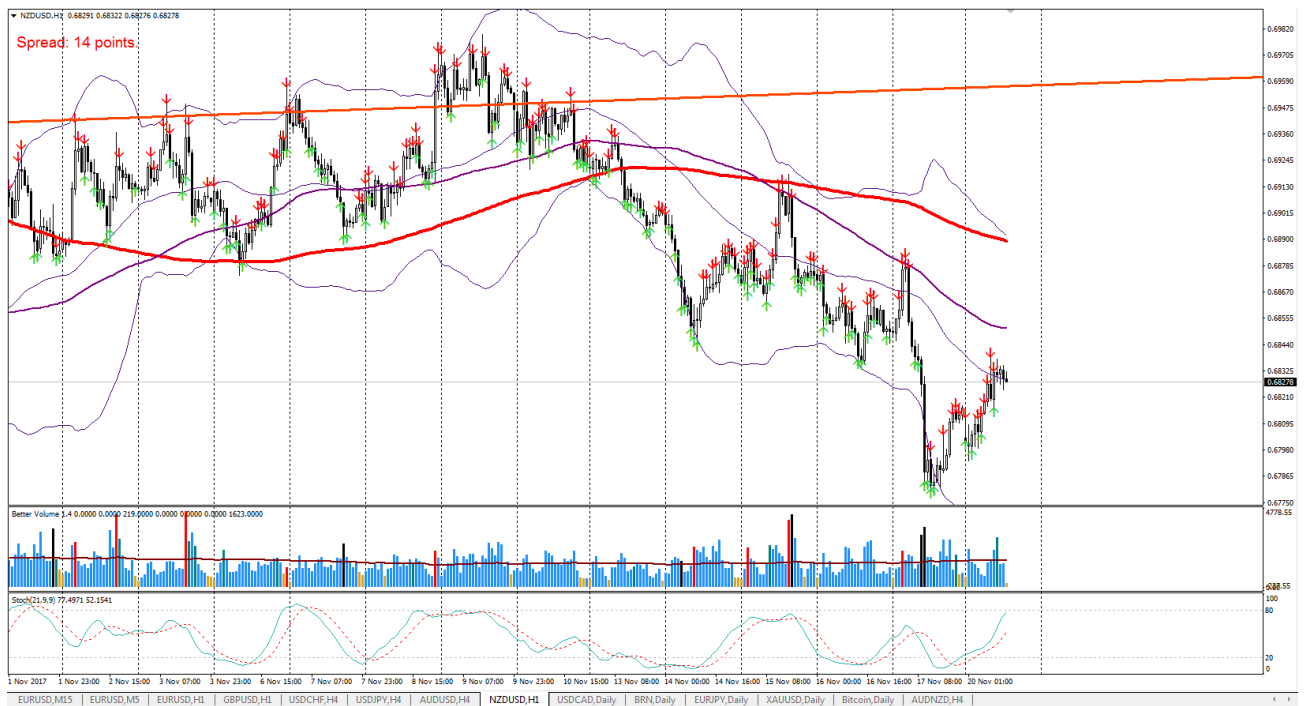


Рис. 3.8. Результат пошуку патернів Price Action і точки входу, які вони генерують, на графіку валютної пари NZD/USD H1.

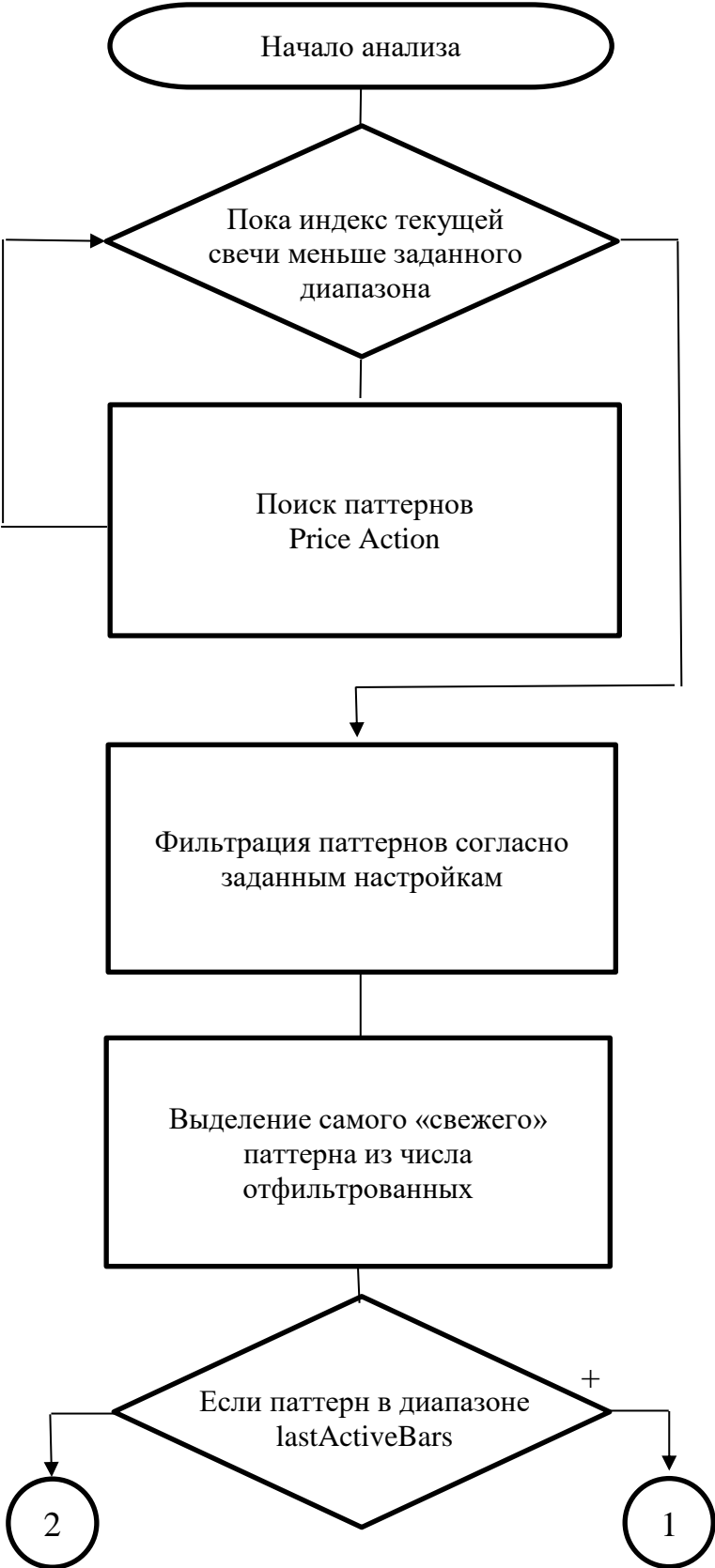
З рисунка 3.8 видно, що виявлених патернів досить багато, саме тому за замовчуванням рекомендується використовувати тільки найбільш сильні патерни, наприклад такі як «Пін-бар» і його різновиди, «Поглинання» або «Розворотне марібозу». Проте в алгоритмі реалізовано визначення 38 патернів Price Action, щоб користувач мав можливість вибору з широкого спектру різних торгових комбінацій.

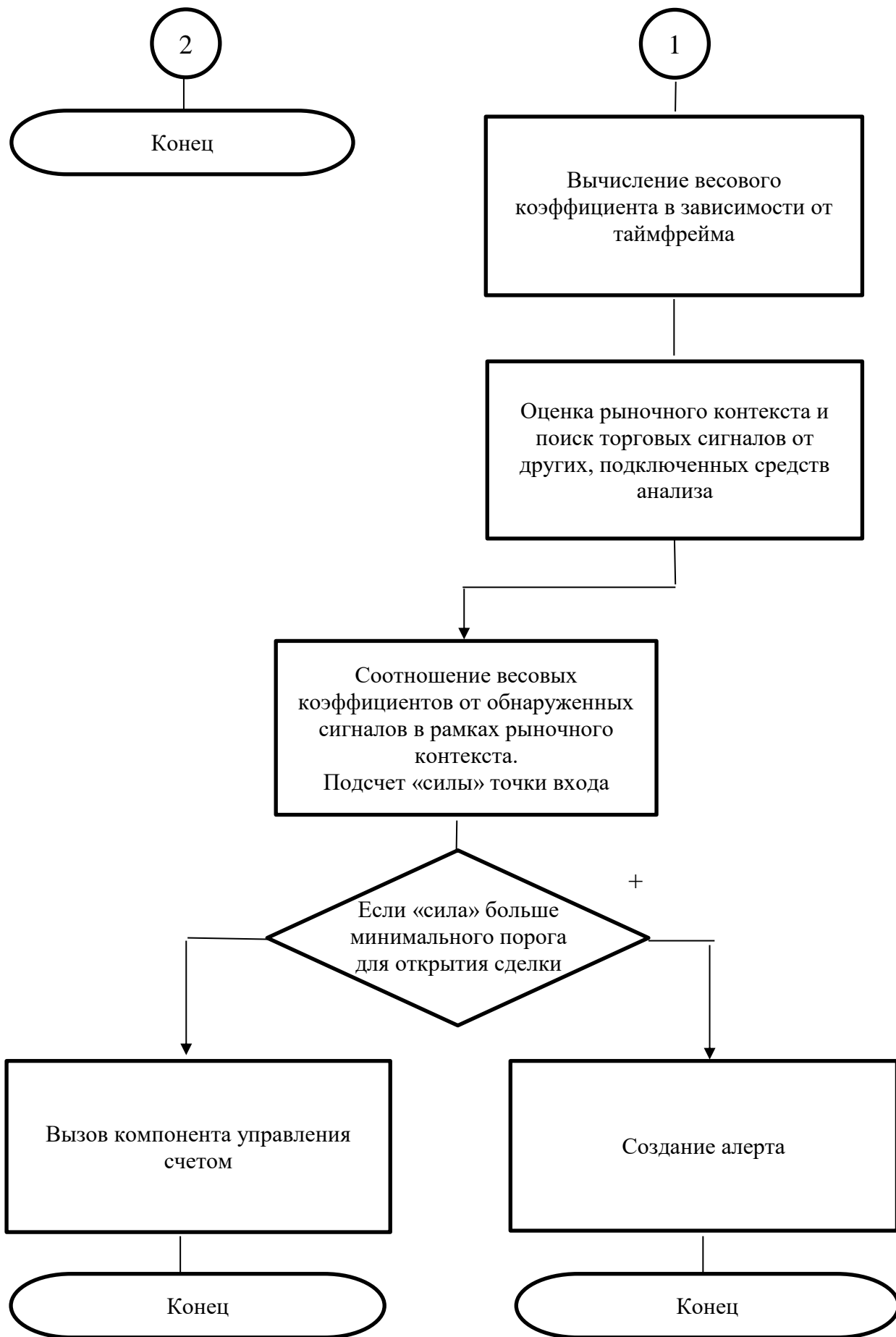
На додаток до вищесказаного, в черговий раз стає очевидною необхідність фільтрації точок входу, які генеруються появою патернів, саме тому, після виявлення патерна, використовуються інші види аналізу для його підтвердження або спростування, і за допомогою механізму вагових коефіцієнтів, описаного в пункті 3.3.3, визначається «сила» точки входу, яка, долаючи певний, заданий за замовчуванням або користувачем, поріг, вважається досить сильною для відкриття позиції.

Виходячи з наданої вище інформації, узагальнена блок-схема, що описує алгоритм пошуку і підтвердження точок входу від моменту сканування графіка

на наявність патернів Price Action до створення готового торгового наказу, виглядає наступним чином (блок-схема 3.2):

Блок-схема 3.2. Алгоритм виявлення патернів Price Action





3. 4. 2. Реалізація компоненту управління рахунком

Компонент управління рахунком має на увазі виконання функцій по відстеженню різних параметрів торгового рахунку, які були визначені раніше, а саме:

- загального балансу;
- доступних засобів;
- відкритих позицій;
- виставлених ордерів;
- поточного прибутку/збитку;
- зарезервованої маржі.

На основі отриманої інформації розраховуються:

- доступний обсяг при віддачі торгових наказів на відкриття позицій або виставлення ордерів;
- визначення рівнів take profit та stop loss для них, на основі обчисленого об'єму;
- побудова сітки ордерів, якщо торгова ситуація по інструменту вважається збитковою;
- закриття сітки відкритих позицій і виставлених ордерів.

Доступний обсяг визначається при налаштуванні торгової системи. Трейдер вказує відсоток від вільної маржі, яку може використовувати робот. За замовчуванням 50%.

Рівні take profit та stop loss можуть не встановлюватись, мати певне значення (наприклад 200 пунктів) або розраховуватись автоматично, виходячи з обсягу позиції.

Якщо використовується сітка, то відстань між ордерами може вказуватись вручну, або розраховуватись автоматично. Крім того, є опція, при активації якої, наступний ордер відкриється лише при появі нової точки входу.

3. 5. Аналіз результатів роботи автоматизованої торгової системи

Тестування автоматизованої торгової системи в режимі реального часу потребує її ініціалізації в терміналі МТ4. Здійснення аналізу проводиться за умови, що термінал МТ4 підключений до відкритого торговельного рахунку брокера, який є постачальником ринкових котирувань в режимі реального часу.

Для запуску автоматизованої торгової системи у терміналі МТ4 необхідно виконати наступні кроки:

1. Помістити скопійований файл торгової системи в каталог даних МТ4, де зберігаються експерти.
2. Відкрити графік котирувань, на якому буде запусканий робот, наприклад, EUR/USD.
3. Активувати режим автоматичної торгівлі в терміналі (рисунок 3.09).



Рис. 3.09. Запуск режиму авто-торгівлі в терміналі МТ4.

4. Відкрити «Навігатор» та на вкладці «Советники» вибрати «Price Action Trade System» (рисунок 3.10).



Рис. 3.10. Вибір розробленої торгової системи

- Двійним натисненням ЛКМ запустити вікно налаштуванням та кнопкою «ОК» активувати радника (рисунок 3.11).

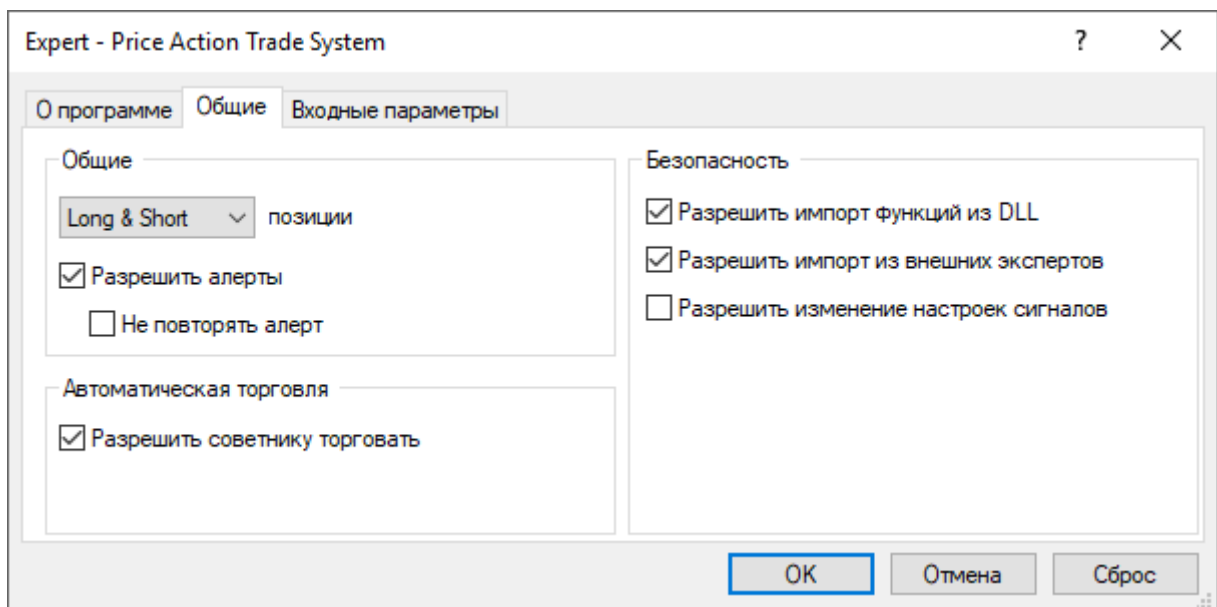


Рис. 3.11. Вікно налаштувань торгівельного радника у МТ4

- Назва радника з'явиться в правому верхньому куту вибраного графіка торгового інструменту. Якщо поряд з назвою відображається

«веселий смайлик», це означає, що радник налаштований та працює правильно (рисунок 3.12).



Рис. 3.12. Вдалилий запуск автоматизованої торгової системи в терміналі MT4

Для тестування був відкритий демо-рахунок з початковим депозитом 1000 USD. Були використані наступні налаштування:

- Доступний відсоток коштів: 100%;
- Обсяг: 0.1;
- Тайм менеджмент: 00:00 – 23:59;
- Період аналізу: 5;
- Використання сітки: true;
- Максимальний розмір сітки: 5;
- Рівень take profit: auto;
- Рівень stop loss: auto;
- Таймфрейм: D1;

- Патерни Price Action: пінбар, поглинання, внутрішній бар;
- Використання наступного набору сигналів: увесь набір;
- Ігнорувати протилежні сигнали: false;
- Сила сигналу: 600;
- Використання п/с: true;
- Спосіб побудови п/с: фрактали;
- Використання МА: true;
- Періоди МА: 100, 200;
- Використання смуг Болинджеру: true;
- Налаштування смуг Болинджеру: 50, 2.5;
- Використання смуг Болинджеру для визначення тренду: true;
- Визначення тренду: true;
- Торгівля по тренду: false;
- Опціонний аналіз: false;
- Опціонні рівні: false;
- Використання сітки Фібоначчі: false;
- Використання для підтвердження сигналу дані осцилятора: true;
- Використання MACD: false;
- Використання RSI: false;
- Використання Stochastic: true;
- Налаштування Stochastic: 14, 8;

У цьому списку перераховані основні налаштування, які мають вплив на алгоритм робота.

Торгова система тестувалась з 11.09.2017 року до 14.01.2018 повністю в автоматичному режимі (без втручання трейдера). Результат тестування повністю збігається з тестовим звітом за аналогічний період (рисунок 3.13).

Strategy Tester Report

Price Action Trade System

ForexClub-MT4 Demo Server (Build 1090)

Символ	EURUSD (Euro vs US Dollar)				
Период	День (D1) 2017.09.11 00:00 - 2018.01.12 00:00 (2017.09.11 - 2018.01.14)				
Модель	Все тики (наиболее точный метод на основе всех наименьших доступных таймфреймов)				
Параметры	PercentCash=100; Lots=0.1; TimeStart=00:00; TimeEnd=23:59; Period=5; TableSystem=true; MaxSizeTableSystem=5; TakeProfit=auto; StopLoss=auto; TimeFrame=D1; Pattern_pinbar=true; Pattern_insidebar=true; Pattern_intake=true; Pattern_dodji=false; Pattern_3toTop=false; Pattern_3toBot=false; Pattern_morningstar=false; Pattern_hammer=false; UseAllSignalsFromOtherSource=true; IgnoreEnemySignal=false; PowerOfSignal=600; UsePS=true; TypePS=fractal; UseMA=true; PeriodsMA=100,200; Bollynger=true; BollyngerSet=50,2.5; BollyngerTrend=true; UseTrendData=true; TradeOnlyOnTrend=false; OptionAnalyse=false; BuildOptionLevels=false; UseFibo=false; UseAddIndicators=true; UseMACD=false; MACDSet=0,0; UseRSI=false; RSISet=0; UseStoch=true; StochSet=14,8;				
Баров в истории	1399	Смоделировано тиков	33066576	Качество моделирования	n/a
Ошибки рассогласования графиков	3826				
Начальный депозит	10000.00	Спред		Текущий (8)	
Чистая прибыль	259.01	Общая прибыль	270.93	Общий убыток	-11.92
Прибыльность	22.73	Матожидание выигрыша	2.94		
Абсолютная просадка	230.41	Максимальная просадка	241.41 (2.41%)	Относительная просадка	2.41% (241.41)
Всего сделок	88	Короткие позиции (% выигравших)	48 (100.00%)	Длинные позиции (% выигравших)	40 (90.00%)
		Прибыльные сделки (% от всех)	84 (95.45%)	Убыточные сделки (% от всех)	4 (4.55%)
		Самая большая прибыльная сделка	5.09	Убыточная сделка	-10.22
		Средняя прибыльная сделка	3.23	Убыточная сделка	-2.98
		Максимальное количество непрерывных выигрышей (прибыль)	60 (201.06)	непрерывных проигрышей (убыток)	1 (-10.22)
		Макс. непрерывная прибыль (число выигрышей)	201.06 (60)	непрерывный убыток (число проигрышей)	-10.22 (1)
		Средний непрерывный выигрыш	17	непрерывный проигрыш	1



Рис. 3.13. Тестовий звіт результатів роботи автоматизованої торгової системи.

У результаті тестування протягом 4 місяців торгова система показала себе досить стабільно, що є найважливішою перевагою. Використовувалися три найбільш ефективні патерни Price Action для визначення точки входу, а також для підвищення ефективності визначався тренд за допомогою МА з періодами 100, 200 та МА зі смуг Болинджеру 50. Для підтвердження сигналу використовувались дані індикатору Stochastic. Як видно з скріншоту, депозит був збільшений з початкових 1000 у. є. до 1259 у. є., тобто на 25%, що, при відсутності великих просадок, є дуже добрим результатом.

Слід зазначити, що такі інструменти аналізу, як опціонний аналіз, побудова рівнів корекції Фібоначі та використання цих рівнів в якості визначення важливих цінових зон відкритого інтересу, не були використані. Також не використовувались більшість із патернів Price Action, тому що при тестуванні було визначено, що чим більше патернів використовується, тим більше помилкових сигналів генерується, а це впливає на якість роботи та стабільність торгового рахунку, тому були обрані лише три, зазначені вище, патерни, ефективність яких була доказана у пункті 2.6.2 другого розділу цієї роботи.

Підводячи підсумки, треба відмітити той факт, що автоматизована торгова система з використанням патернів Price Action, як основного джерела аналізу, тестувалася на реальному рахунку з втручанням трейдера. Оскільки це реальний рахунок, з особистих інтересів автора цифри оголошуватися не будуть, але прибуток в процентному співвідношенні до початкового депозиту склав 280% за той же період часу, що і при тестуванні на демо-рахунку, з максимальною просадкою приблизно в 18.5%.

Ці приклади лише доводять той факт, що сигнали, які виявляє алгоритм торгового роботу є в більшості випадків, ефективними. До того ж приклад з використанням реального рахунку показує, що грамотна участь трейдера може покращити показники ефективності торгової системи у декілька разів, що відкриває великі можливості для вдалого заробітку на світових фінансових ринках, включаючи широко розповсюджений валютний ринок FOREX.

Висновки

1. MetaTrader 4 – це інформаційно-торгова платформа, яка складається з декількох програм, і є однією з найпоширеніших платформ у світі для організації торгової діяльності у сфері ритейлового FOREX.
2. MQL 4 – мова програмування, що вбудована в редактор коду MetaEditor, який поставляється разом з клієнтським терміналом МТ4. Дозволяє розробляти різноманітні види програмних продуктів, а саме скрипти, індикатори та експертів, які можуть виконувати різні типи задач.
3. Автоматизовані торгові системи, які запуснені з-під торговельного терміналу, мають однаковий набір функцій та прав, як і трейдер. Проте для здійснення торгової діяльності необхідно, щоб клієнтський термінал був запуснений та підключений до Інтернету.
4. Будь-яка якісна торгова система повинна мати вбудовані механізми з керування стана торгового рахунку, а саме балансу, маржи, коштів, прибутку тощо.
5. Точка входу в ринок – це сигнал, який генерується від утворення патерну Price Action. Сигнали інших методів технічного аналізу використовуються лише для підтвердження або опровердження точки входу.
6. Для покращення результатів аналізу ринкової динаміки, кожен сигнал має свою вагу. При використанні комбінації декількох сигналів, їх вага складається, та при подоланні встановленого рівня вважається, що сформована точка входу в ринок є достатньо надійною.
7. Проаналізувавши результати тестування автоматизованої торгової системи на історичних даних та в режимі реального часу, на демо-рахунку та на реальному і в повністю автономному режимі та с втручанням трейдера стає можливим зробити висновок, що розроблений робот є ефективним та може приносити стабільний прибуток.

РОЗДІЛ 4. ЕКОНОМІЧНА ЧАСТИНА

4. 1. Визначення трудомісткості розробки програмного забезпечення

Вихідні дані:

1. q – передбачуване число операторів – 1632;
2. C – коефіцієнт складності програми – 1,4;
3. p – коефіцієнт корекції програми в ході її розробки – 0,06;
4. B – коефіцієнт збільшення витрат – 1,2;
5. k - коефіцієнт кваліфікації програміста – 1;
6. $C_{ПР}$ - середня годинна заробітна плата програміста, грн/година.
Згідно з середнього значення погодинної оплати $C_{ПР} = 35,9$ грн/година
7. B_k - число виконавців;
8. F_p - місячний фонд робочого часу (при 40 годинному робочому тижні $F_p=176$ годин).
9. $C_{МЧ}$ - вартість машино-години ЕОМ, грн/год (13,8 грн/година)

Нормування праці в процесі створення ПЗ істотно ускладнено в силу творчого характеру праці програміста. Тому трудомісткість розробки ПЗ може бути розрахована на основі системи моделей з різною точністю оцінки. Трудомісткість розробки ПЗ можна розрахувати за формулою:

$$t = t_0 + t_u + t_a + t_n + t_{oml} + t_d \quad (3.1)$$

де t_0 – витрати праці на підготовку й опис поставленої задачі (приймається 50);

t_u – витрати праці на дослідження алгоритму рішення задачі;

t_a – витрати праці на розробку блок-схеми алгоритму;

t_n – витрати праці на програмування по готовій блок-схемі;

t_{oml} – витрати праці на налагодження програми на ЕОМ;

t_d – витрати праці на підготовку документації.

Складові витрати праці визначаються через умовне число операторів у ПЗ, яке розробляється.

Умовне число операторів (підпрограм):

$$Q = q \times C \times (1 + p) \quad (3.2)$$

де, C – коефіцієнт складності програми – характеризує відносну складність програми по відношенню до так званої типової задачі, що реалізує стандартні методи рішення, складність якої прийнята рівній одиниці (величина C лежить в межах від 1,25 до 2);

q – передбачуване число операторів;

p – коефіцієнт корекції програми в ході її розробки – збільшення обсягу робіт за рахунок внесення змін до алгоритм або програму за результатами уточнення постановок і описів її, зміни складу і структури інформації, а також уточнень, внесених розробниками для поліпшення якості самої програми без зміни постановки завдання (величина p знаходиться в межах 0,05 ... 0,1).

Згідно з визначеними даними умовне число операторів:

$$Q = 1632 \times 1,4 \times (1 + 0,06) = 2422$$

Витрати праці на вивчення опису задачі t_u визначається з урахуванням уточнення опису і кваліфікації програміста:

$$t_u = \frac{Q \times B}{(75..85) \times k}, \text{ людино-годин,} \quad (3.3)$$

де, B - коефіцієнт збільшення витрат праці внаслідок недостатнього опису задачі – якість постановки задачі, виданої для розробки, у зв'язку з тим, що завдання, як правило, вимагають уточнення і деякого доопрацювання (цей коефіцієнт в залежності від складності завдання приймається від 1,2 до 1,5);

k - коефіцієнт кваліфікації програміста, обумовлений від стажу роботи з даної спеціальності – ступінь підготовленості виконавця до дорученої йому роботи (він визначається залежно від стажу роботи і становить: для працюючих до двох років - 0,8);

$$t_u = \frac{2422 \times 1,2}{83 \times 1} = 35, \text{ людино-годин,}$$

Витрати праці на розробку алгоритму рішення задачі:

$$t_a = \frac{Q}{25 \times k} = \frac{2422}{25 \times 1} = 96,8, \text{ людино-годин,} \quad (3.4)$$

Витрати на складання програми по готовій блок-схемі:

$$t_n = \frac{Q}{25 \times k} = \frac{2422}{25 \times 1} = 96,8, \text{ людино-годин,} \quad (3.5)$$

Витрати праці на налагодження програми на ЕОМ:

- за умови автономного налагодження одного завдання:

$$t_{oml} = \frac{Q}{(4..5) \times k} = \frac{2422}{5 \times 1} = 484,4, \text{ людино-годин,} \quad (3.6)$$

- за умови комплексного налагодження завдання:

$$t_{oml}^k = t_{oml} \times 1,5 = 484,4 \times 1,5 = 726,6, \text{ людино-годин,} \quad (3.7)$$

Витрати праці на підготовку документації:

$$t_{\partial} = t_{\partial p} + t_{\partial o}, \text{ людино-годин,} \quad (3.8)$$

де $t_{\partial p}$ – трудомісткість підготовки матеріалів і рукопису.

$$t_{\partial p} = \frac{Q}{(15..20) \times k} = \frac{2422}{20 \times 1} = 121, \text{ людино-годин} \quad (3.9)$$

$t_{\partial o}$ – трудомісткість редагування, печатки й оформлення документації

$$t_{\partial o} = 0,75 \times t_{\partial p} = 0,75 \times 121 = 90,75, \text{ людино-годин} \quad (3.10)$$

Згідно розрахованих значень $t_{\partial p} = 121$, та $t_{\partial o} = 90,75$;

$$t_{\partial} = 121 + 90,75 = 211,75, \text{ людино-годин}$$

Знаючи усі необхідні значення, трудомісткість розробки ПЗ складає:

$$t = 50 + 35 + 96,8 + 96,8 + 484,4 + 211,75 = 974,75$$

4. 2. Витрати на створення програмного забезпечення

Витрати на створення ПЗ $K_{ПО}$ включають витрати на заробітну плату виконавця програми $Z_{ЗП}$ і витрат машинного часу, необхідного на налагодження програми на ЕОМ

$$K_{ПО} = Z_{ЗП} + Z_{МВ}, \text{ грн.} \quad (3.11)$$

Заробітна плата виконавців визначається за формулою:

$$ЗП = t \times C_{ПП}, \text{ грн,} \quad (3.12)$$

де: t - загальна трудомісткість, людино-годин;

$$ЗП = 974,2 \times 35,9 = 34993.52, \text{ грн}$$

Вартість машинного часу, необхідного для налагодження програми на ЕОМ:

$$З_{МВ} = t_{омл} \times C_{Мч}, \text{ грн}, \quad (3.13)$$

де $t_{омл}$ - трудомісткість налагодження програми на ЕОМ, год.

$$З_{МВ} = 484,4 \times 13,8 = 6684.72, \text{ грн}$$

Згідно розрахованих значень витрати на створення ПЗ

$$К_{ПО} = 34993.52 + 6684.72 = 41678.24, \text{ грн}$$

Визначені в такий спосіб витрати на створення програмного забезпечення є частиною одноразових капітальних витрат на створення АСУП.

4. 3. Тривалість розробки програмного забезпечення

Очікуваний період створення ПЗ розрахуємо за формулою:

$$T = \frac{t}{B_k \times F_p}, \text{ міс} \quad (3.14)$$

де B_k – число виконавців ($B_k = 1$).

Очікуваний період створення ПЗ:

$$T = \frac{974.75}{176} = 5,53, \text{ міс}$$

Таким чином, очікувана тривалість розробки складе близько 5,5 місяців, а витрати на створення програмного забезпечення – 41678.24, грн

4. 4. Маркетингові дослідження ринку збуту розробленого програмного продукту

Головною метою під час розробки програмного забезпечення є поліпшення якості життя або роботи певної категорії споживачів, на яких цей продукт був націлений. Для того, щоб програмний продукт користувався попитом, необхідно добре вивчити сферу майбутньої реалізації, а саме потреби споживачів та пропозиції конкурентів.

Задовольнити запити споживачів – непроста задача. Насамперед, потрібно добре вивчити споживача, тобто відповісти на питання: хто купує, яку кількість, за якою ціною, з якою метою, для задоволення яких потреб, де купує. Забезпечити, якщо це необхідно, сервіс.

Відповідаючи на ці питання, стає можливим вивчити запити споживачів та дізнатися, які з них є незадоволеними. Якщо вирішити проблеми, які не в змозі були розв'язати існуючі продукти, що вже обертаються на ринку, то розробка програмного продукту буде економічно обгрунтована.

Для цього були вирішені наступні питання:

1. Сфера реалізації ПЗ та її особливості.
2. Недоліки існуючих конкурентних пропозицій.
3. Які потреби кінцевого споживача є незадоволеними під час проектування ПП.

Проект дипломної роботи був націлений на вирішення проблеми аналізу ціноутворення на світових фінансових ринках та автоматизації торгівельної діяльності приватними трейдерами. Була зроблена спроба створити торгового радника, який є орієнтованим на широкого та неосвіченого в питаннях світової економіки, споживача.

В результаті розроблена автоматизована торгова система, яка використовує комбінований підхід до аналізу ринкової динаміки та патерни Price Action у якості основного елементу аналізу ціноутворення.

1. Сферою реалізації ПП є ритейловий FOREX. Особливості даної індустрії були описані в пункті 1.2.4 цієї роботи. Раніше відмічався той факт, що між трейдерами, які ведуть реальну торгівельну діяльність в інвестиційних компаніях, та трейдерами, які торгують приватно через брокерів задля отримання додаткового джерела прибутку, є суттєва різниця.

Автоматизація торгівельної діяльності поширена у великих гравців ринку, тому що вона має низку переваг, однією з яких є відсутність емоцій та обсяг роботи, яку має можливість виконувати торгова система. Не дивно, що серед приватних трейдерів, які і є основними споживачами розробленого ПП, з'явився

інтерес до торгових роботів, тому що основна аудиторія ритейлового FOREX є не підготовленою, як емоціонально і психологічно, так і низьким рівнем фундаментальних знань про фінансові ринки.

Звідси і виникла індустрія торгових роботів для приватних трейдерів, однак, незважаючи на те, що вона існує вже достатньо довго, торгові роботи, які зараз поширюються в інтернеті, як правило, не тільки не вирішують поставлених проблем, а ще і створюють нові.

2. Основними проблемами існуючих торгових систем, доступних для придбання через інтернет є:

- низька ефективність. Більшість з існуючих безкоштовних торгових радників реалізують дуже примитивні та прості методи аналізу цінового руху. Це не дозволяє таким ПП бути ефективними при використанні та приносити стабільний прибуток трейдеру;
- висока ціна. Деякі програмні роботи продаються за цінами, які є недоступними для більшої частини аудиторії ритейлового FOREX;
- відсутність законодавчого регулювання. Вся сфера ритейлового FOREX ні як не регулюється чинним законодавством. Це поширюється і на індустрію торгових роботів. Коли трейдер купує автоматизовані торгові системи в інтернеті, він діє на свій страх та ризик і цілком може придбати «повітря»;
- відсутність можливості тестування торгової системи. У разі придбання торгової системи в інтернеті, трейдер не має можливості протестувати її ефективність на демо-рахунку. Поєднуючи цей факт з попереднім, більшість споживачів відмовляються від придбання автоматизованих торгових систем.

Підводячи підсумок можна сказати, що безкоштовні роботи є слабо ефективними, а придбання «якісного радника» для приватного трейдера може стати не тільки дорогим «задоволенням», а ще і дуже ризиковим.

3. Знаючи недоліки пропозицій конкурентів, можна виділити ті інтереси потенційних споживачів, які спроможна задовольнити автоматизована торгова

система на основі патернів Price Action, що була розроблена під час виконання цієї магістерської роботи, а саме:

- висока ефективність прогнозування ринкового ціноутворення;
- прибутковість у разі роботи в автоматичному режимі;
- можливість тестування роботи автоматизованої системи на демо-рахунку.

Конкурентноспроможність програмного забезпечення визначається його особливостями та потребами, які воно задовольняє. Якщо є можливість вирішити поставлені вище задачі, то розробка ПЗ є обгрунтованою.

В першу чергу необхідно, щоб торгова система була спроможна вести прибуткову торгівельну діяльність без втручання трейдера. Це можливо завдяки використанню комбінації високоефективних методик з аналізу ринкової динаміки.

Висока точність торгових сигналів, які генерує торгова система, в з'єднанні з закладеною системою управління капіталом дозволяє неосвідченому трейдеру інсталиювати даний продукт та використовувати задані налаштування для роботи на фінансових ринках, отримавши при цьому прибуток.

У разі розповсюдження автоматизованої торгової системи, буде використана азіатська модель монетизації – оплачувана щомісячна підписка. Вибір азіатської моделі обумовлений наступними перевагами:

- перший місяць безкоштовний. За цей період користувач буде взмозі протестувати ефективність програмного продукту;
- вартість використання за місячний період значно нижча ніж придбання продукту в постійне користування, що дозволяє заощаджувати кошти;
- користувач оплачує лише за той час, коли він веде торгову діяльність.

Таким чином, аналіз сфери ритейлового FOREX, визначення основних особливостей та проблем ринку збуту і вирішення питань, які задовольняють інтереси кінцевого споживача – приватного трейдера, дозволяє зробити висновок, що розробка автоматизованої торгової системи з використанням

патернів Price Action є обґрунтованою і спроможною задовільнити потреби аудиторії та принести стабільний прибуток, у разі розповсюдження та монетизації.

4.5. Оцінка економічної ефективності впровадження програмного забезпечення

Таблиця 4.1. – Розрахунок чистих грошових надходжень від розробки ПЗ.

Показники, грн	За роками						Усього за 5 років	Середнє за 5 років
	0	1	2	3	4	5		
1. Інвестиції на ПЗ	41678	-	-	-	-	-	-	-
2. Витрати до впровадження ПЗ,	-	20000	23200	25200	16600	26900	111900	22380
- на тестування торгової системи	-	20000	23200	25200	16600	26900	111900	22380
3. Витрати після впровадження ПО	-	1050	1050	1050	1050	1050	5250	1050
- на аренду VPS-серверу	-	1050	1050	1050	1050	1050	5250	1050
4. Економія	-	18950	22150	24150	15550	25850	106650	21330
5. Амортизація	-	8335	8335	8335	8335	8335	41678	8335
6. Чисті грошові надходження	-	10615	13815	15815	7215	17515	57825	11565
7. Коефіцієнт дисконтування	-	0,87	0,756	0,658	0,572	0,497	-	-
Дисконтові грошові надходження	-	9325	10444	10406	4162	8704	43041	8608

Коефіцієнти економічної ефективності

Чиста поточна вартість доходів:

$$NPU = 43041 - 41678 = 1363 \text{ грн.} > 0$$

Строк окупності:

$$T=41678/8608= 4,8 \text{ років}$$

Індекс прибутковості:

$$ИД=43041/41678=1,03$$

Показник економічної ефективності (NPU - чиста поточна вартість доходів за роки реалізації впровадження (3-5 років) складе 1363 грн тобто відповідає умовам ефективності, тому що $NPU > 0$.

Середній строк окупності капвкладень складе 4,8 роки.

Індекс прибутковості за 5 років складе 1,03, тобто $ИД > 1$, проект варто прийняти.

Таким чином, показник ефективності свідчить про те, що дане впровадження є економічно вигідним.

ВИСНОВКИ

Головною метою під час розробки програмного забезпечення є поліпшення якості життя або роботи певної категорії споживачів, на яких цей продукт був націлений. Для того, щоб програмний продукт користувався попитом, необхідно добре вивчити сферу майбутньої реалізації, а саме потреби споживачів та пропозиції конкурентів, методики проектування існуючих програмних рішень.

Відповідаючи на ці питання, стає можливим вивчити запити споживачів та дізнатися, які з них є незадоволеними. Якщо вирішити проблеми, які не в змозі були розв'язати існуючі продукти, що вже обертаються на ринку, то розробка програмного продукту буде обґрунтована.

Для цього були вирішені наступні питання:

1. Сфера реалізації ПЗ та її особливості.
2. Недоліки існуючих конкурентних пропозицій.
3. Які потреби кінцевого споживача є незадоволеними під час проектування ПЗ.

Проект дипломної роботи був націлений на вирішення проблеми аналізу ціноутворення на світових фінансових ринках та автоматизації торгівельної діяльності приватними трейдерами. Була зроблена спроба створити торгового радника, який є орієнтованим на широкого та неосвіченого в питаннях світової економіки, споживача.

В результаті розроблена автоматизована торгова система, яка використовує комбінований підхід до аналізу ринкової динаміки та патерни Price Action у якості основного елементу аналізу ціноутворення. В алгоритмі пошуку торгівельних сигналів були використані різноманітні види технічного аналізу. Сигнали, що генерують інструменти технічного аналізу, були оцінені за їх ефективністю та контекстом використання в поточній ринковій ситуації.

Розроблена методика присвоєння ваги різноманітним торговим сигналам, дозволяє під час утворення точки входу в ринок, оцінити поточну ринкову ситуацію з різних інтерпретацій ціноутворення та відокремити найбільш

вірогідні та надійні сигнали, які спроможні принести прибуток у разі відкриття ринкової позиції.

У разі розповсюдження автоматизованої торгової системи, буде використана азіатська модель монетизації – оплачувана щомісячна підписка. Це дозволяє при зменшенні вартості продукту, використовувати торгову систему максимально великій кількості споживачів.

Таким чином, аналіз сфери ритейлового FOREX, визначення основних особливостей та проблем ринку збуту і вирішення питань, які задовольняють інтереси кінцевого споживача – приватного трейдера, дозволяє зробити висновок, що розробка автоматизованої торгової системи з використанням патернів Price Action є обґрунтованою і спроможною задовільнити потреби аудиторії та принести стабільний прибуток, у разі розповсюдження та монетизації.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Adamatzky A. (Ed.) *Advances in Unconventional Computing. Volume 2. Prototypes, Models and Algorithms.* – Springer International Publishing, 2017. – 812 p.
2. Albers V., Still B. (Eds.) *Usability of Complex Information Systems: Evaluation of User Interaction.* – CRC Press, 2011. – 392 p.
3. Arlow Jim, Neustadt Ila. *Enterprise Patterns and MDA: Building Better Software with Archetype Patterns and UML.* – Addison–Wesley Professional, 2004. 528 p.
4. Aslaksen E.W. *Designing Complex Systems: Foundations of Design in the Functional Domain.* – Auerbach Publications; 1 edition (October 27, 2008). – 176 p.
5. Barrera D.G., Diaz M. *Communicating Systems with UML 2: Modeling and Analysis of Network Protocols.* – ISTE Ltd. – Wiley, 2011. – 268 p.
6. Baskerville R., De Marco M., Spagnoletti P. (Eds.) *Designing Organizational Systems: An Interdisciplinary Discourse.* – Springer, 2013 – 342 p.
7. Boehm Barry W. *A Spiral Model of Software Development and Enhancement.* – Object Management Group, 2010. – 180 p.
8. Borland. *Delphi 7: Developer's Guide.* Author and Publishers: Borland Software Corporation, 2002. – 1108 p.
9. Braun A.W. *Enterprise Software Delivery.* – Addison–Wesley, 2012. – 291 p.
10. Camp O., Filipe J.B.L., Hammoudi S., Piattini M. *Enterprise Information Systems V.* – Kluwer, 2005. –339 p.
11. Cantù M. *Delphi 2010 Handbook: A Guide to the New Features of Delphi 2010.* – CreateSpace Independent Publishing Platform, 2010. – 318 p.
12. Cantu M. *Mastering Delphi 7.* – Sybex, 2003. – 992 p.
13. Cheng B.H.C., de Lemos R., Paola H.G., Magee I.J. (eds.) *Software Engineering for Self–Adaptive Systems.* – Springer, 2009. –270 p.

14. Cruz–Cunha M.M. Social, Managerial, and Organizational Dimensions of Enterprise Information Systems. – IGI Global, 2010. –604 p.
15. Cruz–Cunha M.M., Varajago J. (Eds.) Enterprise Information Systems Design, Implementation and Management: Organizational Applications. – Information Science Reference, 2010. – 622 p.
16. for Reverse Engineering Technologies: Strategic Directions and System Evolution. – Premier Reference Source, 2010. – 386 p.
17. Frankel D. Model Driven Architecture: Applying MDA to Enterprise Computing. – Wiley, 2003, 352 p.
18. Grassle P. et al. UML 2.0 in action: a project based tutorial. – Packt Publishing, 2005. – 232 p.
19. Gunasekaran A. (Ed.) Modeling and analysis of enterprise information systems. – IGI Global, 2007. – 392 p.
20. Hrubby P. Model–Driven Design Using Business Patterns. – Springer, 2006. – 359 p.
21. Podeswa H. UML for the IT Business Analyst: A Practical Guide to Requirements Gathering Using the Unified Modeling Language. – Course Technology PTR, 2009. – 400 p.
22. Rebovich G.J. Enterprise Systems Engineering: Advances in the Theory and Practice. – CRC Press, 2010. – 477 p.
23. Saaty T.L., Vargas L.G. Models, Methods, Concepts & Applications of the Analytic Hierarchy Process. – Springer US, 2001. – 333 p.
24. Teti D. Delphi Cookbook: 50 hands–on recipes to master the power of Delphi for cross–platform and mobile development on Windows, Mac OS X, Android, and iOS. Packt Publishing, 2014. – 329 p.
25. Vogel O. Software Architecture: A Comprehensive Framework and Guide for Practitioners. – Springer, 2011. – 550 p.
26. Ward John, Peppard Joe. Strategic Planning for Information Systems. – John Wiley & Sons, 2002. – 641 p.

27. Акимов В.А., Воронов Е.М., Крыжановская Т.Г. Проектирование информационных систем. – М.: МГМУ «МАМИ» (Университет машиностроения), 2013. – 126 с.
28. Арунянц Г.Г. Проектирование информационных систем. – Калининград: Балтийский институт экономики и финансов, 2010. – 254 с.
29. Архангельский А.Я. Delphi 2006. Справочное пособие. – М.: ООО «Бином–Пресс», 2006. – 1152 с.
30. Архангельский А.Я. Программирование в Delphi для Windows. Версии 2006, 2007, Turbo Delphi. – М.: ООО "Бином–Пресс", 2007. – 1248 с.
31. Брукс П. Проектирование процесса проектирования. – М.: Вильямс, 2013. – 464 с.
32. Гвоздева Т.В., Баллод Б.А. Проектирование информационных систем. – Ростов н/Д.: Феникс, 2009. – 512 с.
33. Головчинер М.Н. Проектирование информационных систем. – Томск: ТГУ, 2015. – 110 с.
34. Гома Х. UML–проектирование систем реального времени параллельных и распределенных приложений. – М.: ДМК–Пресс, 2011. – 704 с
35. Грекул В.И., Денищенко Г.Н., Коровкина Н.Л. Проектирование информационных систем. – М.: Интернет–Университет Информационных Технологий «Интуит», 2016. – 570 с.
36. Грибачев К.Г. Delphi и Model Driven Architecture. Разработка приложений баз данных.– СПб.: Изд–во Питер, 2004. – 352 с.
37. Коцюба И.Ю., Чунаев А.В., Шиков А.Н. Основы проектирования информационных систем. Учебное пособие. – СПб.: Университет ИТМО, 2015. – 206 с.
38. Кузнецов М.Б. Трансформация UML–моделей и ее применение в технологии MDA. – М.: Институт системного программирования РАН, 2005. – 13 с.

39. Культин Н. Основы программирования в Delphi 2010. – СПб.: БХВ–Петербург, 2010. – 438 с.
40. Ларман Крэг. Применение UML 2.0 и шаблонов проектирования. 3–е изд. – М.: Вильямс, 2013. – 736 с.
41. Новикова Г.М. Основы разработки корпоративных инфокоммуникационных систем. Учеб. пособие. М.: РУДН, 2008. – 152 с.
42. Осипов Д.Л. Базы данных и Delphi. Теория и практика. – СПб.: БХВ–Петербург, 2011. – 855 с.
43. Осипов Д. Delphi XE2. – СПб.: БХВ–Петербург, 2012. – 912 с.
44. Осипов Д. Delphi. Программирование для Windows, OS X, iOS и Android. – СПб.: БХВ–Петербург, 2014. – 464 с.
45. Рубанцев Валерий. Delphi на пальцах. Осваиваем новые технологии Windows 8: Touch и Gesture. Учебно–популярное издание. – М.: Издательство RVGames, 2013 – 81 с.
46. Скиена С. Алгоритмы. Руководство по разработке. 2–е изд.: Пер. с англ. – СПб.: БХВ–Петербург. 2011. – 720 с.
47. Сухарев М.В. Основы Delphi. Профессиональный подход. – СПб.: Наука и техника, 2004. – 600 с.
48. Хомоненко А.Д., Гофман В.Э., Мещеряков Е.В. Delphi 7. – СПб.: БХВ–Петербург, 2010. – 1136 с.
49. Якунин Ю.Ю. Технологии разработки программного обеспечения. – Красноярск: ИПК СФУ, 2008. – 225с.
50. Бойко В.В. Экономика предприятий Украины. Основной курс: Учебник для вузов. - Д.: Пороги, 1997. - 312 с.
51. Андрусенко Г.О. Основы маркетинга. – Е.:НМК ВО, 1992. – 143 с
52. Баркан Д.И. Маркетинг для всех. – Л.: «Культинформпрес», 1991. – 256 с.
53. Бусыгин А.В. Предпринимательство. Основной курс: Учебник для вузов. – М.: ИНФРА-М, 1997. – 608 с.

54. Завялов П.С., Демидов В.Е. Формула успеха – маркетинг. – М.: Междунар. Отношения, 1991.
55. Котлер Ф. Основы маркетинга: Пер. с англ. / Общ. Ред. и вступ. ст. Е.М. Пеньковой – М.: Прогресс, 1990. – 636с.
56. Скворцов Н.Н. Бизнес-план предприятия. – К.: Вища школа, 1995. – 187 с.
57. Мескон М.Х., Альберт М., Хедоури Ф. Основы менеджмента. – М.: Дело, 1992. - 702 с.
58. Мете А.Ф., Штец К.А., Бельгольский Б.П. и др. Организация и планирование предприятий. - М.: Металлургия, 1986. - 560 с.
59. Основы инновационного менеджмента: Теория и практика: Учеб. пособие / Под ред. П.Н. Завлина и др. – М: ОАО Издательство «Экономика». 2000. – 475 с.

Додаток А

ТЕКСТ ПРОГРАМИ

```
#property indicator_chart_window
#property indicator_buffers 2
#property indicator_color1 Red
#property indicator_color2 LimeGreen

extern string PivotRangeStart = "00:30";
extern string PivotRangeEnd = "00:30";
extern bool DisplayPivotPoint = true;
extern bool DisplayPreviousHighLow = true;
extern bool DisplayMAS = false;

extern bool ShowStrongReversalPatterns = TRUE;
extern bool ShowWeakReversalPatterns = FALSE;
extern bool ShowContinuationPatterns = FALSE;
extern bool ShowUnclassified = FALSE;
extern int ViewBars = 350;
extern bool SoundAlert = TRUE;
extern bool EmailAlert = FALSE;
extern color SellColor = Red;
extern color BuyColor = LimeGreen;
extern color TextColor = Yellow;

extern int PpercentCash = 100;
extern float Lots = 0.1;
extern string TimeStart = "00:00";
extern string TimeEnd = "23:59";
extern int CustomPeriod = 5;
extern bool TableSystem = true;
extern int MaxSizeTableSystem = 5;
extern string TakeProfit = "auto";
extern string StopLoss = "auto";
extern string TimeFrame = "D1";
extern bool Pattern_pinbar = true;
extern bool Pattern_insidebar = true;
extern bool Pattern_intake = true;
extern bool Pattern_dodji = false;
extern bool Pattern_3toTop = false;
extern bool Pattern_3toBot = false;
extern bool Pattern_morningstar = false;
extern bool Pattern_hammer = false;
extern bool UseAllSignalsFromOtherSource = true;
extern bool IgnoreEnemySignal = false;
extern int PowerOfSignal = 600;
extern bool UsePS = true;
extern string TypePS = "fractal";
extern bool UseMA = true;
extern float PeroidsMA = 100,200;
extern bool Bollynger = true;
extern float BollyngerSet = 50,2.5;
```

```

extern bool BollyngerTrend = true;
extern bool UseTrendData = true;
extern bool TradeOnlyOnTrend = false;
extern bool OptionAnalyse = false;
extern bool BuildOptionLevels = false;
extern bool UseFibo = false;
extern bool UseAddIndicators = true;
extern bool UseMACD = false;
extern float MACDSet = 0,0;
extern bool UseRSI = false;
extern int RSISet = 0;
extern bool UseStoch = true;
extern float StochSet = 14,8;

```

```

double Buffer1[];
double Buffer2[];
double Buffer3[];
double Buffer4[];
double Buffer5[];
double Buffer6[];
double Buffer7[];
double Buffer8[];

```

```

double pivots[50];

```

```

double pivotRangeHigh;
double pivotRangeLow;
double pivotRangeClose;
double pivotPoint;
double pivotDiff;

```

```

double pivotTop=0;
double pivotBottom=0;

```

```

double pivot14MA;
double pivot30MA;
double pivot50MA;

```

```

int openBar;

```

```

double g_ibuf_116[];
double g_ibuf_120[];
datetime g_time_124;
bool gi_128 = FALSE;
string gs_unused_132 = "10.";
string gs_unused_140 = "20";
int gi_148 = -1;
int gi_152 = 0;
int gi_156 = 1;
int gi_160 = 2;
int gi_164 = 3;
int gi_unused_168 = 39;

```

```

int gia_172[];
int gia_176[];
string gs_myMMObj_180 = "MyMMObj";
int gi_188 = 0;
int gi_192 = 1;
int gi_196 = 2;
int gi_200 = 3;
int gi_204 = 4;
int gi_208 = 5;
int gi_212 = 6;
int gi_216 = 7;
int gi_220 = 8;
int gi_224 = 9;
int gi_228 = 10;
int gi_232 = 11;
int gi_236 = 12;
int gi_240 = 13;
int gi_244 = 14;
int gi_248 = 15;
int gi_252 = 16;
int gi_256 = 17;
int gi_260 = 18;
int gi_264 = 19;
int gi_268 = 20;
int gi_272 = 21;
int gi_276 = 22;
int gi_280 = 23;
int gi_284 = 24;
int gi_288 = 25;
int gi_292 = 26;
int gi_296 = 27;
int gi_300 = 28;
int gi_304 = 29;
int gi_308 = 30;
int gi_312 = 31;
int gi_316 = 32;
int gi_320 = 33;
int gi_324 = 34;
int gi_328 = 35;
int gi_332 = 36;
int gi_336 = 37;
int gi_340 = 38;
string gsa_344[39];
int gia_348[39];
int gia_352[39];

int initDefine() {
    gsa_344[gi_188] = "Молот";
    gia_352[gi_188] = 0;
    gsa_344[gi_304] = "Повешенный";
    gia_352[gi_304] = 0;
    gsa_344[gi_192] = "Модель поглощения";
    gia_352[gi_192] = 0;
}

```

gsa_344[gi_196] = "Утренняя звезда";
gia_352[gi_196] = 0;
gsa_344[gi_200] = "Вечерняя звезда";
gia_352[gi_200] = 0;
gsa_344[gi_212] = "Завеса из темных облаков";
gia_352[gi_212] = 0;
gsa_344[gi_216] = "Просвет в облаках";
gia_352[gi_216] = 0;
gsa_344[gi_204] = "Падающая звезда";
gia_352[gi_204] = 0;
gsa_344[gi_208] = "Перевернутый молот";
gia_352[gi_208] = 0;
gsa_344[gi_220] = "Харами";
gia_352[gi_220] = 1;
gsa_344[gi_224] = "Вершина Пинцет";
gia_352[gi_224] = 1;
gsa_344[gi_228] = "Основание Пинцет";
gia_352[gi_228] = 1;
gsa_344[gi_232] = "Захват за Пояс";
gia_352[gi_232] = 1;
gsa_344[gi_236] = "Две взлетевшие вороны";
gia_352[gi_236] = 1;
gsa_344[gi_244] = "Три вороны";
gia_352[gi_244] = 1;
gsa_344[gi_240] = "Удержание на Татами";
gia_352[gi_240] = 2;
gsa_344[gi_248] = "Контратака";
gia_352[gi_248] = 3;
gsa_344[gi_252] = "Разделение";
gia_352[gi_252] = 2;
gsa_344[gi_256] = "Доджи-надгробие";
gia_352[gi_256] = 3;
gsa_344[gi_260] = "Длинноногий Доджи";
gia_352[gi_260] = 3;
gsa_344[gi_264] = "Медвежий Доджи";
gia_352[gi_264] = 0;
gsa_344[gi_324] = "Бычий Доджи";
gia_352[gi_324] = 0;
gsa_344[gi_268] = "Разрыв Тасуки";
gia_352[gi_268] = 2;
gsa_344[gi_272] = "Смежные белые свечи";
gia_352[gi_272] = 2;
gsa_344[gi_276] = "Три метода";
gia_352[gi_276] = 2;
gsa_344[gi_280] = "Три реки, основание";
gia_352[gi_280] = 3;
gsa_344[gi_284] = "Окно вверх";
gia_352[gi_284] = 3;
gsa_344[gi_288] = "Окно Вниз";
gia_352[gi_288] = 3;
gsa_344[gi_292] = "Три белых солдата";
gia_352[gi_292] = 3;
gsa_344[gi_296] = "Отбитое наступление";

```

gia_352[gi_296] = 1;
gsa_344[gi_300] = "Торможение";
gia_352[gi_300] = 3;
gsa_344[gi_316] = "Доджи Поглощение";
gia_352[gi_316] = 3;
gsa_344[gi_320] = "Перевернутый Молот";
gia_352[gi_320] = 3;
gsa_344[gi_328] = "Верхний разрыв метода трех";
gia_352[gi_328] = 2;
gsa_344[gi_332] = "Нижний разрыв метода трех";
gia_352[gi_332] = 2;
gsa_344[gi_336] = "Тройной удар";
gia_352[gi_336] = 1;
gsa_344[gi_340] = "У линии шеи";
gia_352[gi_340] = 1;
gia_348[gi_316] = -150;
gia_348[gi_188] = -100;
gia_348[gi_304] = -100;
gia_348[gi_192] = -100;
gia_348[gi_196] = -100;
gia_348[gi_200] = -100;
gia_348[gi_212] = -100;
gia_348[gi_216] = -100;
gia_348[gi_204] = -90;
gia_348[gi_208] = -90;
gia_348[gi_220] = -80;
gia_348[gi_224] = -50;
gia_348[gi_228] = -50;
gia_348[gi_232] = -80;
gia_348[gi_236] = -80;
gia_348[gi_244] = -90;
gia_348[gi_240] = 0;
gia_348[gi_248] = -80;
gia_348[gi_252] = 0;
gia_348[gi_256] = 0;
gia_348[gi_260] = 0;
gia_348[gi_264] = 0;
gia_348[gi_324] = 0;
gia_348[gi_268] = 0;
gia_348[gi_272] = 0;
gia_348[gi_276] = 0;
gia_348[gi_280] = -90;
gia_348[gi_284] = 100;
gia_348[gi_288] = 100;
gia_348[gi_292] = 100;
gia_348[gi_296] = -80;
gia_348[gi_300] = -80;
return (0);
}

double GetUpperShadowHeight(int ai_0) {
    return (MathAbs(High[ai_0] - MathMax(Close[ai_0],
Open[ai_0])));
}

```



```

}

double GetLowerShadowHeight(int ai_0) {
    return (MathAbs(MathMin(Close[ai_0], Open[ai_0]) - Low[ai_0]));
}

double GetBodyHeight(int ai_0) {
    return (MathAbs(Close[ai_0] - Open[ai_0]));
}

double GetAllHeight(int ai_0) {
    return (MathAbs(High[ai_0] - Low[ai_0]));
}

bool IsHigher(int ai_0) {
    if (High[ai_0] >= High[ai_0 + 1] + 2.0 * Point && High[ai_0] >=
High[ai_0 + 2] + 2.0 * Point && High[ai_0] >= High[ai_0 + 3] + 2.0
* Point) return (TRUE);
    return (FALSE);
}

int IsBodyHigher(int ai_0) {
    if (Close[ai_0] > GetHighCloseOpen(ai_0 + 1) + 2.0 * Point &&
Close[ai_0] > GetHighCloseOpen(ai_0 + 2) + 2.0 * Point) return
(1);
    return (0);
}

bool IsLower(int ai_0) {
    if (Low[ai_0] + 2.0 * Point < Low[ai_0 + 1] && Low[ai_0] + 2.0
* Point < Low[ai_0 + 2] && Low[ai_0] + 2.0 * Point < Low[ai_0 +
3]) return (TRUE);
    return (FALSE);
}

int IsYing(int ai_0) {
    if (Close[ai_0] < Open[ai_0]) return (1);
    return (0);
}

int IsYang(int ai_0) {
    if (Close[ai_0] > Open[ai_0]) return (1);
    return (0);
}

int RemoveObjects() {
    for (int li_0 = GetFirstMyObjectandText(); li_0 > -1; li_0 =
GetFirstMyObjectandText()) ObjectDelete(ObjectName(li_0));
    return (0);
}

int GetFirstMyObjectandText() {
    for (int li_ret_0 = 0; li_ret_0 < ObjectsTotal(); li_ret_0++)

```

```

        if (StringFind(ObjectName(li_ret_0), "My", 0) >= 0) return
(li_ret_0);
    return (-1);
}

double GetLowCloseOpen(int ai_0) {
    return (MathMin(Close[ai_0], Open[ai_0]));
}

double GetHighCloseOpen(int ai_0) {
    return (MathMax(Close[ai_0], Open[ai_0]));
}

string GetNextObjectName(string as_0) {
    return (as_0 + DoubleToStr(ObjectsTotal(), 0));
}

int testBodyHeight(int ai_0, int ai_4) {
    if (GetBodyHeight(ai_0) >= ai_4 * Point) return (1);
    return (0);
}

int AlmostSameBodyHeight(int ai_0, int ai_4) {
    if (MathAbs(GetBodyHeight(ai_0) - GetBodyHeight(ai_4)) < 5.0 *
Point) return (1);
    return (0);
}

void CreateTextObject(int a_datetime_0, double a_price_4, color
a_color_12, string a_text_16) {
    string l_name_24 = GetNextObjectName(gs_mymmobj_180);
    ObjectCreate(l_name_24, OBJ_TEXT, 0, a_datetime_0, a_price_4);
    ObjectSetText(l_name_24, a_text_16, 7);
    ObjectSet(l_name_24, OBJPROP_COLOR, a_color_12);
}

int AddintoCFArray(int ai_0, int ai_4) {
    int l_arr_size_8 = ArraySize(gia_172);
    ArrayResize(gia_172, l_arr_size_8 + 1);
    ArrayResize(gia_176, l_arr_size_8 + 1);
    gia_172[l_arr_size_8] = ai_0;
    gia_176[l_arr_size_8] = ai_4;
    return (0);
}

bool IsHammer(int ai_0) {
    if (GetAllHeight(ai_0) >= 10.0 * Point &&
GetUpperShadowHeight(ai_0) < GetAllHeight(ai_0) / 5.0 &&
GetLowerShadowHeight(ai_0) > 2.0 * GetBodyHeight(ai_0) &&
GetUpperShadowHeight(ai_0) > 2.0 * Point)
        if (IsHigher(ai_0)) return (TRUE);
    return (FALSE);
}

```

```

bool IsHammerCFM(int ai_0) {
    if (IsHammer(ai_0 + 1) && Open[ai_0] < GetLowCloseOpen(ai_0 +
2) && Close[ai_0] < GetLowCloseOpen(ai_0 + 2)) return (TRUE);
    return (FALSE);
}

bool IsHangMan(int ai_0) {
    if (GetAllHeight(ai_0) >= 10.0 * Point &&
GetUpperShadowHeight(ai_0) < GetAllHeight(ai_0) / 5.0 &&
GetLowerShadowHeight(ai_0) > 2.0 * GetBodyHeight(ai_0))
        if (IsHigher(ai_0)) return (TRUE);
    return (FALSE);
}

bool IsHangManCFM(int ai_0) {
    if (IsHangMan(ai_0 + 1) && Open[ai_0] > GetHighCloseOpen(ai_0 +
2) && Close[ai_0] > GetHighCloseOpen(ai_0 + 2)) return (TRUE);
    return (FALSE);
}

bool IsDoji(int ai_0) {
    if (MathAbs(Open[ai_0] - Close[ai_0]) < 3.0 * Point) return
(TRUE);
    return (FALSE);
}

bool IsInvertHammer(int ai_0) {
    if (GetLowerShadowHeight(ai_0) < GetAllHeight(ai_0) / 5.0) {
        if (GetUpperShadowHeight(ai_0) > 2.0 * GetBodyHeight(ai_0))
            if (IsLower(ai_0)) return (TRUE);
    }
    return (FALSE);
}

bool IsInvertHammerCFM(int ai_unused_0) {
    return (FALSE);
}

bool IsThree_Crows(int ai_0) {
    if (IsHigher(ai_0 + 2) && High[ai_0 + 1] > High[ai_0 + 2] &&
IsYing(ai_0 + 2) && IsYing(ai_0 + 1) && IsYing(ai_0) && Open[ai_0
+ 1] < Open[ai_0 + 2] && Close[ai_0 +
1] < Close[ai_0 + 2] && Open[ai_0] < Open[ai_0 + 1] &&
Close[ai_0] < Close[ai_0 + 1]) return (TRUE);
    return (FALSE);
}

bool IsThree_White_Soldiers(int ai_0) {
    if (IsYang(ai_0 + 2) && IsYang(ai_0 + 1) && IsYang(ai_0 + 0) &&
Open[ai_0 + 1] > Open[ai_0 + 2] + GetBodyHeight(ai_0 + 2) / 2.0 &&
Open[ai_0 + 0] > Open[ai_0 + 1] +

```

```

        GetBodyHeight(ai_0 + 1) / 2.0 && Close[ai_0 + 1] >
Close[ai_0 + 2] && Close[ai_0 + 0] > Close[ai_0 + 1] && High[ai_0
+ 1] > High[ai_0 + 2] && High[ai_0 + 0] > High[ai_0 + 1] &&
testBodyHeight(ai_0 + 2, 20) && testBodyHeight(ai_0 + 1, 20) &&
testBodyHeight(ai_0 + 0, 20) && AlmostSameBodyHeight(ai_0 + 2,
ai_0 + 1) && AlmostSameBodyHeight(ai_0 + 1, ai_0 + 0)) return
(TRUE);
    return (FALSE);
}

int MMCFCCondition(int ai_0) {
    int l_count_4;
    if (!IsDoji(ai_0 + 2)) {
        if (IsYang(ai_0 + 2) != IsYang(ai_0 + 1)) {
            if (MathMax(Close[ai_0 + 1], Open[ai_0 + 1]) >
MathMax(Close[ai_0 + 2], Open[ai_0 + 2]) && MathMin(Close[ai_0 +
1], Open[ai_0 + 1]) < MathMin(Close[ai_0 + 2], Open[ai_0 +
2])) {
                if (IsLower(ai_0 + 2) || IsLower(ai_0 + 1) &&
IsYang(ai_0 + 1)) AddintoCFArray(gi_192, gi_152);
                if (IsHigher(ai_0 + 2) || IsHigher(ai_0 + 1) &&
IsYing(ai_0 + 1)) AddintoCFArray(gi_192, gi_156);
                if (GetBodyHeight(ai_0 + 2) >= 15.0 * Point ||
IsLower(ai_0 + 1) && IsYang(ai_0 + 1)) AddintoCFArray(gi_192,
gi_152);
                if (GetBodyHeight(ai_0 + 2) >= 15.0 * Point ||
IsHigher(ai_0 + 1) && IsYing(ai_0 + 1)) AddintoCFArray(gi_192,
gi_156);
            }
        }
    }
    if (IsDoji(ai_0 + 2)) {
        if (MathMax(Close[ai_0 + 1], Open[ai_0 + 1]) >
MathMax(Close[ai_0 + 2], Open[ai_0 + 2]) && MathMin(Close[ai_0 +
1], Open[ai_0 + 1]) < MathMin(Close[ai_0 + 2], Open[ai_0 +
2])) {
            if (IsLower(ai_0 + 2) || IsLower(ai_0 + 1) && IsYang(ai_0
+ 1)) AddintoCFArray(gi_316, gi_152);
            if (IsHigher(ai_0 + 2) || IsHigher(ai_0 + 1) &&
IsYing(ai_0 + 1)) AddintoCFArray(gi_316, gi_156);
        }
    }
    if (IsYang(ai_0 + 2) && IsYing(ai_0 + 1) && IsHigher(ai_0 + 2)
&& GetBodyHeight(ai_0 + 2) >= 20.0 * Point && Open[ai_0 + 1] >
High[ai_0 + 2] && Close[ai_0 + 1] < Open[ai_0 +
2] + (Close[ai_0 + 2] - (Open[ai_0 + 2])) / 2.0 &&
GetLowCloseOpen(ai_0 + 1) > GetLowCloseOpen(ai_0 + 2))
AddintoCFArray(gi_212, gi_156);
    if (MathAbs(Close[ai_0 + 2] - (Open[ai_0 + 2])) > 5.0 * Point)
{
        if (MathMax(Close[ai_0 + 1], Open[ai_0 + 1]) <
MathMax(Close[ai_0 + 2], Open[ai_0 + 2]) && MathMin(Close[ai_0 +
1], Open[ai_0 + 1]) > MathMin(Close[ai_0 + 2], Open[ai_0 +

```

```

2])) {
    if ((IsYang(ai_0 + 2) && IsHigher(ai_0 + 2)) ||
(IsYang(ai_0 + 2) && GetBodyHeight(ai_0 + 2) >= 15.0 * Point))
AddintoCFArray(gi_220, gi_156);
    if ((IsYing(ai_0 + 2) && IsLower(ai_0 + 2)) ||
(IsYing(ai_0 + 2) && GetBodyHeight(ai_0 + 2) >= 15.0 * Point))
AddintoCFArray(gi_220, gi_152);
}
}
if (IsHammer(ai_0 + 1)) AddintoCFArray(gi_188, gi_160);
if (IsHangMan(ai_0 + 1)) AddintoCFArray(gi_304, gi_164);
if (IsHammerCFM(ai_0 + 1)) AddintoCFArray(gi_308, gi_152);
if (IsHangManCFM(ai_0 + 1)) AddintoCFArray(gi_312, gi_156);
if (IsInvertHammer(ai_0 + 1)) AddintoCFArray(gi_208, gi_160);
if (IsInvertHammerCFM(ai_0 + 1)) AddintoCFArray(gi_320,
gi_152);
if (IsYang(ai_0 + 1) && IsYing(ai_0 + 2) && GetBodyHeight(ai_0
+ 2) >= 10.0 * Point && Open[ai_0 + 1] < Low[ai_0 + 2] &&
Close[ai_0 + 1] > Close[ai_0 + 2] + (Open[ai_0 +
2] - (Close[ai_0 + 2])) / 2.0 && GetHighCloseOpen(ai_0 + 1)
< GetHighCloseOpen(ai_0 + 2)) AddintoCFArray(gi_216, gi_152);
if (IsYing(ai_0 + 3) && IsYang(ai_0 + 1) && IsLower(ai_0 + 3))
{
    if (GetLowCloseOpen(ai_0 + 3) > GetHighCloseOpen(ai_0 + 2)
|| GetLowCloseOpen(ai_0 + 1) > GetHighCloseOpen(ai_0 + 2) &&
GetHighCloseOpen(ai_0 + 1) > GetLowCloseOpen(ai_0 +
3) && GetBodyHeight(ai_0 + 2) <= 10.0 * Point &&
GetBodyHeight(ai_0 + 3) >= 8.0 * Point && GetBodyHeight(ai_0 + 1)
>= 8.0 * Point) AddintoCFArray(gi_196, gi_152);
}
if (IsYang(ai_0 + 3) && IsYing(ai_0 + 1) && IsHigher(ai_0 + 3))
{
    if (GetHighCloseOpen(ai_0 + 3) < GetLowCloseOpen(ai_0 + 2)
|| GetHighCloseOpen(ai_0 + 1) < GetLowCloseOpen(ai_0 + 2) &&
GetLowCloseOpen(ai_0 + 1) < GetHighCloseOpen(ai_0 +
3) && GetBodyHeight(ai_0 + 2) <= 10.0 * Point &&
GetBodyHeight(ai_0 + 3) > 8.0 * Point && GetBodyHeight(ai_0 + 1) >
8.0 * Point) AddintoCFArray(gi_200, gi_156);
}
if (GetLowerShadowHeight(ai_0 + 1) < GetAllHeight(ai_0 + 1) /
4.0) {
    if (GetUpperShadowHeight(ai_0 + 1) > 2.0 *
GetBodyHeight(ai_0 + 1))
        if (IsHigher(ai_0 + 1) || IsBodyHigher(ai_0 + 1) ||
IsBodyHigher(ai_0 + 2)) AddintoCFArray(gi_204, gi_156);
}
if (GetAllHeight(ai_0 + 1) > 10.0 * Point && IsYang(ai_0 + 2)
&& IsHigher(ai_0 + 2) && High[ai_0 + 2] == High[ai_0 + 1] ||
High[ai_0 + 3] == High[ai_0 + 1] || High[ai_0 +
4] == High[ai_0 + 1]) AddintoCFArray(gi_224, gi_156);
if (GetAllHeight(ai_0 + 1) > 10.0 * Point && IsYing(ai_0 + 2)
&& IsLower(ai_0 + 2) && Low[ai_0 + 2] == Low[ai_0 + 1] || Low[ai_0
+ 3] == Low[ai_0 + 1] || Low[ai_0 +

```

```

4] == Low[ai_0 + 1]) AddintoCFArray(gi_228, gi_152);
if (GetBodyHeight(ai_0 + 1) >= 20.0 * Point && IsYang(ai_0 + 1)
&& GetLowerShadowHeight(ai_0) == 0.0 && IsLower(ai_0 + 1) &&
GetBodyHeight(ai_0 + 1) > GetAllHeight(ai_0 +
1) / 2.0) AddintoCFArray(gi_232, gi_152);
if (GetBodyHeight(ai_0 + 1) >= 20.0 * Point && IsYing(ai_0 + 1)
&& GetUpperShadowHeight(ai_0) == 0.0 && IsHigher(ai_0 + 1) &&
GetBodyHeight(ai_0 + 1) > GetAllHeight(ai_0 +
1) / 2.0) AddintoCFArray(gi_232, gi_156);
if (IsHigher(ai_0 + 3) && IsYang(ai_0 + 3) && IsYing(ai_0 + 2)
&& IsYing(ai_0 + 1) && Open[ai_0 + 2] > Open[ai_0 + 3] &&
Open[ai_0 + 1] > Open[ai_0 + 2] && Close[ai_0 +
1] < Close[ai_0 + 2] && GetLowCloseOpen(ai_0 + 1) >
GetHighCloseOpen(ai_0 + 3) && GetLowCloseOpen(ai_0 + 2) >
GetHighCloseOpen(ai_0 + 3)) AddintoCFArray(gi_236, gi_156);
if (IsYang(ai_0 + 5) && IsYing(ai_0 + 4) && IsYing(ai_0 + 3) &&
IsYing(ai_0 + 2) && IsYang(ai_0 + 1) && Close[ai_0 + 4] >
Close[ai_0 + 5] + 2.0 * Point && Open[ai_0 +
3] > Open[ai_0 + 4] && Close[ai_0 + 3] < Close[ai_0 + 4] &&
Open[ai_0 + 2] > Open[ai_0 + 3] && Close[ai_0 + 2] < Close[ai_0 +
3] && Open[ai_0 + 1] > Close[ai_0 + 2] &&
GetBodyHeight(ai_0 + 5) >= 20.0 * Point &&
GetBodyHeight(ai_0 + 1) >= 20.0 * Point) AddintoCFArray(gi_240,
gi_152);
if (IsThree_Crows(ai_0 + 1)) AddintoCFArray(gi_244, gi_156);
if (GetBodyHeight(ai_0 + 1) > 5.0 * Point && GetBodyHeight(ai_0
+ 2) > 5.0 * Point) {
if (IsHigher(ai_0 + 1) && IsYang(ai_0 + 2) && IsYing(ai_0 +
1) && MathAbs(GetLowCloseOpen(ai_0 + 2) - GetHighCloseOpen(ai_0 +
1)) <= 2.0 * Point) AddintoCFArray(gi_248, gi_156);
if (IsLower(ai_0 + 1) && IsYing(ai_0 + 2) && IsYang(ai_0 +
1) && MathAbs(GetLowCloseOpen(ai_0 + 1) - GetHighCloseOpen(ai_0 +
2)) <= 2.0 * Point) AddintoCFArray(gi_248, gi_152);
}
if (GetBodyHeight(ai_0 + 1) > 5.0 * Point && GetBodyHeight(ai_0
+ 2) > 5.0 * Point) {
if (IsYang(ai_0 + 2) && IsYing(ai_0 + 1) &&
MathAbs(Open[ai_0 + 1] - (Open[ai_0 + 2])) <= 2.0 * Point)
AddintoCFArray(gi_252, gi_156);
if (IsYing(ai_0 + 2) && IsYang(ai_0 + 1) &&
MathAbs(Open[ai_0 + 1] - (Open[ai_0 + 2])) <= 2.0 * Point)
AddintoCFArray(gi_252, gi_152);
}
if (Close[ai_0 + 1] == Open[ai_0 + 1] && Close[ai_0 + 1] ==
Low[ai_0 + 1]) AddintoCFArray(gi_256, gi_156);
if (MathAbs(Close[ai_0 + 1] - (Open[ai_0 + 1])) < 3.0 * Point
&& GetAllHeight(ai_0 + 1) > 20.0 * Point && High[ai_0 + 1] >
High[ai_0 + 2]) AddintoCFArray(gi_260, gi_156);
if (GetBodyHeight(ai_0 + 2) > 20.0 * Point && IsHigher(ai_0 +
2) || IsHigher(ai_0 + 1) && IsYang(ai_0 + 2) && MathAbs(Close[ai_0
+ 1] - (Open[ai_0 + 1])) < 2.0 * Point &&
MathAbs(High[ai_0 + 1] - MathMax(Close[ai_0 + 1], Open[ai_0
+ 1])) > 2.0 * Point && MathAbs(Low[ai_0 + 1] - MathMin(Close[ai_0

```

```

+ 1], Open[ai_0 + 1])) > 2.0 * Point) AddintoCFArray(gi_264,
gi_156);
    if (GetBodyHeight(ai_0 + 2) > 20.0 * Point && IsLower(ai_0 + 2)
|| IsLower(ai_0 + 1) && IsYing(ai_0 + 2) && MathAbs(Close[ai_0 +
1] - (Open[ai_0 + 1])) < 2.0 * Point &&
    MathAbs(High[ai_0 + 1] - MathMax(Close[ai_0 + 1], Open[ai_0
+ 1])) > 2.0 * Point && MathAbs(Low[ai_0 + 1] - MathMin(Close[ai_0
+ 1], Open[ai_0 + 1])) > 2.0 * Point) AddintoCFArray(gi_324,
gi_152);
    if (IsHigher(ai_0 + 3) || IsHigher(ai_0 + 2) && (IsYang(ai_0 +
3) && IsYang(ai_0 + 2) && IsYing(ai_0 + 1)) && Low[ai_0 + 2] -
(High[ai_0 + 3]) >= 2.0 * Point && Low[ai_0 +
    1] - (High[ai_0 + 3]) >= 2.0 * Point && (Open[ai_0 + 1] >
Open[ai_0 + 2] && Open[ai_0 + 1] < Close[ai_0 + 2]) && Close[ai_0
+ 1] < Close[ai_0 + 2] && MathAbs(GetBodyHeight(ai_0 +
    1) - GetBodyHeight(ai_0 + 2)) < 5.0 * Point)
AddintoCFArray(gi_268, gi_152);
    if (IsLower(ai_0 + 3) || IsLower(ai_0 + 2) && (IsYing(ai_0 + 3)
&& IsYing(ai_0 + 2) && IsYang(ai_0 + 1)) && Low[ai_0 + 3] -
(High[ai_0 + 2]) >= 2.0 * Point && Low[ai_0 +
    3] - (High[ai_0 + 1]) >= 2.0 * Point && (Open[ai_0 + 1] >
Close[ai_0 + 2] && Open[ai_0 + 1] < Open[ai_0 + 2]) && Close[ai_0
+ 1] > Open[ai_0 + 2] && MathAbs(GetBodyHeight(ai_0 +
    1) - GetBodyHeight(ai_0 + 2)) < 5.0 * Point)
AddintoCFArray(gi_268, gi_156);
    if (IsHigher(ai_0 + 3) && (IsYang(ai_0 + 3) && IsYang(ai_0 + 2)
&& IsYang(ai_0 + 1)) && Low[ai_0 + 2] - (High[ai_0 + 3]) >= 2.0 *
Point && MathAbs(Open[ai_0 + 1] > Open[ai_0 +
    2]) < 2.0 * Point && MathAbs(GetBodyHeight(ai_0 + 1) >
GetBodyHeight(ai_0 + 2)) < 4.0 * Point) AddintoCFArray(gi_272,
gi_152);
    if (IsLower(ai_0 + 3) && (IsYing(ai_0 + 3) && IsYang(ai_0 + 2)
&& IsYang(ai_0 + 1)) && Low[ai_0 + 2] - (High[ai_0 + 3]) >= 2.0 *
Point && MathAbs(Open[ai_0 + 1] > Open[ai_0 +
    2]) < 2.0 * Point && MathAbs(GetBodyHeight(ai_0 + 1) >
GetBodyHeight(ai_0 + 2)) < 4.0 * Point) AddintoCFArray(gi_272,
gi_156);
    if (IsHigher(ai_0 + 5) && IsYang(ai_0 + 5) && IsYang(ai_0 + 1)
&& IsYing(ai_0 + 2) && IsYing(ai_0 + 3) && IsYing(ai_0 + 4) &&
GetBodyHeight(ai_0 + 5) > 20.0 * Point &&
    GetBodyHeight(ai_0 + 1) > 20.0 * Point && Open[ai_0 + 1] >
Open[ai_0 + 5] && Close[ai_0 + 1] > Close[ai_0 + 5] && Open[ai_0 +
2] > Open[ai_0 + 5] && Close[ai_0 + 2] < Close[ai_0 + 5] &&
Open[ai_0 + 3] > Open[ai_0 + 5] && Close[ai_0 + 3] < Close[ai_0 +
5] && Open[ai_0 + 4] > Open[ai_0 + 5] && Close[ai_0 + 4] <
Close[ai_0 + 5]) AddintoCFArray(gi_276, gi_152);
    if (IsLower(ai_0 + 5) && IsYing(ai_0 + 5) && IsYing(ai_0 + 1)
&& IsYang(ai_0 + 2) && IsYang(ai_0 + 3) && IsYang(ai_0 + 4) &&
GetBodyHeight(ai_0 + 5) > 20.0 * Point &&
    GetBodyHeight(ai_0 + 1) > 20.0 * Point && Open[ai_0 + 1] <
Open[ai_0 + 5] && Close[ai_0 + 1] < Close[ai_0 + 5] && Open[ai_0 +
2] < Open[ai_0 + 5] && Close[ai_0 + 2] > Close[ai_0 + 5] &&
Open[ai_0 + 3] < Open[ai_0 + 5] && Close[ai_0 + 3] > Close[ai_0 +

```

```

5] && Open[ai_0 + 4] < Open[ai_0 + 5] && Close[ai_0 + 4] >
Close[ai_0 + 5]) AddintoCFArray(gi_276, gi_156);
    if (IsYing(ai_0 + 3) && IsYing(ai_0 + 2) && IsLower(ai_0 + 3)
&& IsLower(ai_0 + 2) && GetBodyHeight(ai_0 + 3) > 20.0 * Point &&
GetAllHeight(ai_0 + 3) - GetBodyHeight(ai_0 +
    3) < 10.0 * Point && GetBodyHeight(ai_0 + 2) < 10.0 * Point
&& GetLowerShadowHeight(ai_0 + 2) > GetBodyHeight(ai_0 + 2) &&
GetUpperShadowHeight(ai_0 + 2) <= 2.0 * Point &&
    GetBodyHeight(ai_0 + 1) < 10.0 * Point &&
GetLowerShadowHeight(ai_0 + 1) <= 2.0 * Point &&
GetUpperShadowHeight(ai_0 + 1) <= 2.0 * Point)
AddintoCFArray(gi_280, gi_152);
    if (IsYang(ai_0 + 1) && Low[ai_0 + 1] > Low[ai_0 + 2] + 2.0 *
Point) {
        for (int li_8 = 4; li_8 <= 11; li_8++) {
            if (GetBodyHeight(ai_0 + 2 + li_8) > 40.0 * Point) {
                l_count_4 = 0;
                for (int li_12 = li_8 + 1; li_12 > 0; li_12--) {
                    if (High[li_12 + ai_0 + 2] < Close[ai_0 + 2 + li_8]
&& Low[li_12 + ai_0 + 2] > Open[ai_0 + 2 + li_8]) continue;
                    l_count_4++;
                }
                if (l_count_4 == 0) AddintoCFArray(gi_284, gi_152);
            }
        }
    }
    if (IsThree_White_Soldiers(ai_0 + 1)) AddintoCFArray(gi_292,
gi_152);
    if (IsYang(ai_0 + 3) && IsYang(ai_0 + 2) && IsYang(ai_0 + 1) &&
Open[ai_0 + 2] > Open[ai_0 + 3] + GetBodyHeight(ai_0 + 3) / 2.0 &&
Open[ai_0 + 1] > Open[ai_0 + 2] +
        GetBodyHeight(ai_0 + 2) / 2.0 && Close[ai_0 + 2] >
Close[ai_0 + 3] && Close[ai_0 + 1] > Close[ai_0 + 2] && High[ai_0
+ 2] > High[ai_0 + 3] && High[ai_0 + 1] > High[ai_0 + 2] &&
testBodyHeight(ai_0 + 3, 20) && GetBodyHeight(ai_0 + 3) >
GetBodyHeight(ai_0 + 2) && GetBodyHeight(ai_0 + 2) >
GetBodyHeight(ai_0 + 1) && Open[0] < Open[1])
AddintoCFArray(gi_296, gi_156);
    if (IsYang(ai_0 + 3) && IsYang(ai_0 + 2) && IsYang(ai_0 + 1) &&
Open[ai_0 + 2] > Open[ai_0 + 3] + GetBodyHeight(ai_0 + 3) / 2.0 &&
Open[ai_0 + 1] > Open[ai_0 + 2] +
        GetBodyHeight(ai_0 + 2) / 2.0 && Close[ai_0 + 2] >
Close[ai_0 + 3] && Close[ai_0 + 1] > Close[ai_0 + 2] && High[ai_0
+ 2] > High[ai_0 + 3] && High[ai_0 + 1] > High[ai_0 + 2] &&
testBodyHeight(ai_0 + 2, 20) && GetBodyHeight(ai_0 + 3) < 10.0 *
Point && GetBodyHeight(ai_0 + 1) < 10.0 * Point && Open[0] <
Open[1]) AddintoCFArray(gi_300, gi_156);
    if (IsYang(ai_0 + 3) && IsYang(ai_0 + 2) && IsYing(ai_0 + 1) &&
IsHigher(ai_0 + 3) && Low[ai_0 + 2] > High[ai_0 + 3] + 2.0 * Point
&& Open[ai_0 + 1] > Open[ai_0 +
    2] && Open[ai_0 + 1] < Close[ai_0 + 2] && Close[ai_0 + 1] >
Open[ai_0 + 3] + (Close[ai_0 + 1]) < Close[ai_0 + 3])
AddintoCFArray(gi_328, gi_152);

```



```

    if (IsYing(ai_0 + 3) && IsYing(ai_0 + 2) && IsYang(ai_0 + 1) &&
    IsLower(ai_0 + 3) && High[ai_0 + 2] < Low[ai_0 + 3] - 2.0 * Point
    && Open[ai_0 + 1] > Close[ai_0 +
        2] && Open[ai_0 + 1] < Open[ai_0 + 2] && Close[ai_0 + 1] >
    Close[ai_0 + 3] + (Close[ai_0 + 1]) < Open[ai_0 + 3])
    AddintoCFArray(gi_332, gi_156);
    if (IsYang(ai_0 + 1) && IsThree_Crows(ai_0 + 2) && High[ai_0 +
    1] > iHighest(NULL, 0, MODE_HIGH, 3, 2) && Low[ai_0 + 1] <
    iLowest(NULL, 0, MODE_LOW, 3, 2)) AddintoCFArray(gi_336, gi_156);
    if (IsYing(ai_0 + 1) && IsThree_White_Soldiers(ai_0 + 2) &&
    High[ai_0 + 1] > iHighest(NULL, 0, MODE_HIGH, 3, 2) && Low[ai_0 +
    1] < iLowest(NULL, 0, MODE_LOW, 3, 2)) AddintoCFArray(gi_336,
    gi_152);
    if (IsYing(ai_0 + 2) && IsYang(ai_0 + 1) && testBodyHeight(ai_0
    + 1, 20) && testBodyHeight(ai_0 + 2, 20) && Close[ai_0 + 1] <
    Open[ai_0 + 2] - (Open[ai_0 + 2] - (Close[ai_0 +
        2])) / 2.0 && Open[ai_0 + 1] < GetLowCloseOpen(ai_0 + 2))
    AddintoCFArray(gi_340, gi_156);
    return (gi_148);
}

```

```

int init() {
    gi_128 = TRUE;
    SetIndexStyle(0, DRAW_ARROW, STYLE_SOLID, 1);
    SetIndexArrow(0, 226);
    SetIndexBuffer(0, g_ibuf_120);
    SetIndexStyle(1, DRAW_ARROW, STYLE_SOLID, 1);
    SetIndexArrow(1, 225);
    SetIndexBuffer(1, g_ibuf_116);
    SetIndexEmptyValue(0, 0.0);
    SetIndexEmptyValue(1, 0.0);

    if(TimeFrame == "M1")        {tframe = 1;}
    if(TimeFrame == "M5")        {tframe = 5;}
    if(TimeFrame == "M15")       {tframe = 15;}
    if(TimeFrame == "M30")       {tframe = 30;}
    if(TimeFrame == "H1")        {tframe = 60;}
    if(TimeFrame == "H4")        {tframe = 240;}
    if(TimeFrame == "D1")        {tframe = 1440;}
    if(TimeFrame == "W1")        {tframe = 10080;}
    if(TimeFrame == "MN1")       {tframe = 43200;}

    SetIndexStyle(0, DRAW_LINE, STYLE_DOT, 1);
    SetIndexBuffer(0, Buffer1);
    SetIndexLabel(0, "Pivot Point");

    SetIndexStyle(1, DRAW_LINE, STYLE_DASH, 1);
    SetIndexBuffer(1, Buffer2);
    SetIndexLabel(1, "Pivot Range Top");

    SetIndexStyle(2, DRAW_LINE, STYLE_DASH, 1);
    SetIndexBuffer(2, Buffer3);
    SetIndexLabel(2, "Pivot Range Bottom");
}

```

```

SetIndexStyle(3,DRAW_LINE, STYLE_SOLID, 1);
SetIndexBuffer(3,Buffer4);
SetIndexLabel(3,"Previous Day High");

SetIndexStyle(4,DRAW_LINE, STYLE_SOLID, 1);
SetIndexBuffer(4,Buffer5);
SetIndexLabel(4,"Previous Day Low");

SetIndexStyle(5,DRAW_LINE);
SetIndexBuffer(5,Buffer6);
SetIndexLabel(5,"14 MA");

SetIndexStyle(6,DRAW_LINE);
SetIndexBuffer(6,Buffer7);
SetIndexLabel(6,"30 MA");

SetIndexStyle(7,DRAW_LINE);
SetIndexBuffer(7,Buffer8);
SetIndexLabel(7,"50 MA");

initDefine();
return (0);
}

int deinit() {
RemoveObjects();
return (0);
}

int start() {
RemoveObjects();
if (gi_128 == FALSE) return (0);
for (int l_count_0 = 0; l_count_0 <= ViewBars; l_count_0++)
getCFResult(l_count_0);
return (0);
}

int getCFResult(int ai_0) {
int li_4;
double ld_8;
string ls_16;
int li_ret_24 = gi_148;
int l_arr_size_28 = 0;
ArrayResize(gia_172, 0);
ArrayResize(gia_176, 0);
MMCFCondition(ai_0);
l_arr_size_28 = ArraySize(gia_172);
for (int l_index_32 = 0; l_index_32 < l_arr_size_28;
l_index_32++) {
li_ret_24 = gia_176[l_index_32];
}
}

```

```

    li_4 = gia_172[l_index_32];
    if (l_index_32 == 0) ld_8 = 15.0 * Point;
    else ld_8 = 15.0 * Point + l_index_32 * 8;
    if (li_ret_24 == gi_164 || li_ret_24 == gi_156 &&
        (ShowStrongReversalPatterns == TRUE && gia_352[li_4] == 0) ||
        (ShowWeakReversalPatterns == TRUE && gia_352[li_4] == 1) ||
        (ShowContinuationPatterns == TRUE && gia_352[li_4] == 2)
        || (ShowUnclassified == TRUE && gia_352[li_4] == 3)) {
        g_ibuf_120[ai_0 + 1] = High[ai_0 + 1] + 8.0 * Point;
        CreateTextObject(Time[ai_0 + 1], High[ai_0 + 1] + ld_8,
            TextColor, gsa_344[li_4]);
        ls_16 = gsa_344[li_4];
    }
    if (li_ret_24 == gi_160 || li_ret_24 == gi_152 &&
        (ShowStrongReversalPatterns == TRUE && gia_352[li_4] == 0) ||
        (ShowWeakReversalPatterns == TRUE && gia_352[li_4] == 1) ||
        (ShowContinuationPatterns == TRUE && gia_352[li_4] == 2)
        || (ShowUnclassified == TRUE && gia_352[li_4] == 3)) {
        g_ibuf_116[ai_0 + 1] = Low[ai_0 + 1] - 8.0 * Point;
        CreateTextObject(Time[ai_0 + 1], Low[ai_0 + 1] - ld_8,
            TextColor, gsa_344[li_4]);
        ls_16 = gsa_344[li_4];
    }
}
if (li_ret_24 != gi_148 && ai_0 < 1) {
    if (g_time_124 != Time[0]) {
        Print(TimeToStr(Time[0]), Period(), ls_16, " in ",
            Symbol());
        if (EmailAlert) SendMail(Period() + ls_16 + " in " +
            Symbol(), "");
        if (SoundAlert) PlaySound("alert2.wav");
        g_time_124 = Time[0];
    }
}

```

```

string barTime="", lastBarTime="";
string barDay="", lastBarDay="";
int closeBar;

if(typePS == "fractal")
{
    for(int i=Bars; i>=0; i--)
    {
        barTime = TimeToStr(Time[i], TIME_MINUTES);
        lastBarTime = TimeToStr(Time[i+1], TIME_MINUTES);
        barDay = TimeToStr(Time[i], TIME_DATE);
        lastBarDay = TimeToStr(Time[i+1], TIME_DATE);

        //need to handle if pivotrangestart/end is 00:00
        if ((PivotRangeEnd == "00:00" && barTime>=PivotRangeEnd
            && barDay>lastBarDay) || (barTime>=PivotRangeEnd &&
            lastBarTime<PivotRangeEnd))
    }
}

```

```

        {
            closeBar = i + 1;

            if (openBar>0)
            {
                calculatePivotRangeValues(openBar, closeBar);
            }
        }

        if      ((PivotRangeStart      ==      "00:00"      &&
barTime>=PivotRangeStart      &&      barDay>lastBarDay)      ||
(barTime>=PivotRangeStart && lastBarTime<PivotRangeStart))
        {
            openBar = i;
        }

        if (openBar>0)
        {
            drawIndicators(i);
        }
    }
}

return (li_ret_24);
}

```

```

void calculatePivotRangeValues(int openBar, int closeBar)
{
    pivotRangeHigh = High[Highest(NULL, 0, MODE_HIGH, (openBar -
closeBar + 1), closeBar)];
    pivotRangeLow  = Low[Lowest(NULL, 0, MODE_LOW, (openBar -
closeBar + 1), closeBar)];
    pivotRangeClose = Close[closeBar];
    pivotPoint      =      (pivotRangeHigh      +      pivotRangeLow      +
pivotRangeClose)/3;
    pivotDiff       =      MathAbs(((pivotRangeHigh + pivotRangeLow)/2) -
pivotPoint);
    pivotTop        = pivotPoint + pivotDiff;
    pivotBottom     = pivotPoint - pivotDiff;

    if (DisplayMAs) calcPivotMA();
}

```

```

void calcPivotMA()
{
    //create temp array
    double pivs[50];

    //load new pivot
    ArrayCopy(pivs,pivots,1,0,49);
}

```

```

pivs[0] = pivotPoint;
ArrayCopy(pivots, pivs, 0, 0, WHOLE_ARRAY);

//calcMA
double pivSum = 0;
int count = ArraySize(pivots);
if (count>=14)
{
    for (int p=0; p<count; p++)
    {
        pivSum+=pivots[p];

        if (p==13)
        {
            pivot14MA = pivSum/14;
        }

        if (p==29)
        {pivot30MA = pivSum/30;}

        if (p==49)
        {pivot50MA = pivSum/50;}
    }
}

void drawIndicators(int curBar)
{
    if (DisplayPivotPoint) Buffer1[curBar]=pivotPoint;

    Buffer2[curBar]=pivotTop;
    Buffer3[curBar]=pivotBottom;

    if (DisplayPreviousHighLow)
    {
        Buffer4[curBar]=pivotRangeHigh;
        Buffer5[curBar]=pivotRangeLow;
    }

    if (DisplayMAs)
    {
        Buffer6[curBar]=pivot14MA;
        Buffer7[curBar]=pivot30MA;
        Buffer8[curBar]=pivot50MA;
    }
}

void OnTick(void)
{
    double PACurrent,PAPrevious;
    double SignalCurrent,SignalPrevious;
    double MaCurrent,MaPrevious;
    int    cnt,ticket,total;

```

```

//---
// initial data checks
// it is important to make sure that the expert works with a
normal
// chart and the user did not make any mistakes setting external
// variables (Lots, StopLoss, TakeProfit,
// TrailingStop) in our case, we check TakeProfit
// on a chart of less than 100 bars
//---
    bool signalPA = getCFResult(l_count_0);

    if(Bars<100)
    {
        Print("bars less than 100");
        return;
    }
    if(TakeProfit<10)
    {
        Print("TakeProfit less than 10");
        return;
    }
//--- to simplify the coding and speed up access data are put into
internal variables
    PACurrent=iPA(NULL,0,12,26,9,PRICE_CLOSE,MODE_MAIN,0);
    PAPrevious=iPA(NULL,0,12,26,9,PRICE_CLOSE,MODE_MAIN,1);
    SignalCurrent=iPA(NULL,0,12,26,9,PRICE_CLOSE,MODE_SIGNAL,0);
    SignalPrevious=iPA(NULL,0,12,26,9,PRICE_CLOSE,MODE_SIGNAL,1);
    MaCurrent=iMA(NULL,0,MATrendPeriod,0,MODE_EMA,PRICE_CLOSE,0);
    MaPrevious=iMA(NULL,0,MATrendPeriod,0,MODE_EMA,PRICE_CLOSE,1);

//TODO make right !!!!
    if(checkPower == true)
    {
        powerSignalSet = SFUSignal;
    }
    else
    {
        powerSignalSet = 99999;
    }

    if(powerSignal >= powerSignalSet)
    {
        signalPA = false;
        Print("Not enough Power of signal");
    }

    total=OrdersTotal();
    if((total<1) && (signalPA==true))
    {
        //--- no opened orders identified
        if(AccountFreeMargin()<(1000*Lots))

```

```

        {
            Print("We have no money. Free Margin =
",AccountFreeMargin());
            return;
        }
        //--- check for long position (BUY) possibility
        if(PACurrent<0 && PACurrent>SignalCurrent &&
PAPrevious<SignalPrevious &&
            MathAbs(PACurrent)>(PAOpenLevel*Point) &&
MaCurrent>MaPrevious)
        {

ticket=OrderSend(Symbol(),OP_BUY,Lots,Ask,3,0,Ask+TakeProfit*Point
,"PA sample",16384,0,Green);
            if(ticket>0)
            {
                if(OrderSelect(ticket,SELECT_BY_TICKET,MODE_TRADES))
                    Print("BUY order opened : ",OrderOpenPrice());
            }
            else
                Print("Error opening BUY order : ",GetLastError());
            return;
        }
        //--- check for short position (SELL) possibility
        if(PACurrent>0 && PACurrent<SignalCurrent &&
PAPrevious>SignalPrevious &&
            PACurrent>(PAOpenLevel*Point) && MaCurrent<MaPrevious)
        {
            ticket=OrderSend(Symbol(),OP_SELL,Lots,Bid,3,0,Bid-
TakeProfit*Point,"PA sample",16384,0,Red);
            if(ticket>0)
            {
                if(OrderSelect(ticket,SELECT_BY_TICKET,MODE_TRADES))
                    Print("SELL order opened : ",OrderOpenPrice());
            }
            else
                Print("Error opening SELL order : ",GetLastError());
        }
        //--- exit from the "no opened orders" block
        return;
    }
    //--- it is important to enter the market correctly, but it is
more important to exit it correctly...
    for(cnt=0;cnt<total;cnt++)
    {
        if(!OrderSelect(cnt,SELECT_BY_POS,MODE_TRADES))
            continue;
        if(OrderType()<=OP_SELL && // check for opened position
            OrderSymbol()==Symbol()) // check for symbol
        {
            //--- long position is opened
            if(OrderType()==OP_BUY)
            {

```

```

        //--- should it be closed?
        if(PACurrent>0    &&    PACurrent<SignalCurrent    &&
PAPrevious>SignalPrevious &&
        PACurrent>(PACloseLevel*Point))
        {
            //--- close order and exit

if(!OrderClose(OrderTicket(),OrderLots(),Bid,3,Violet))
        Print("OrderClose error ",GetLastError());
        return;
    }
    //--- check for trailing stop
    if(TrailingStop>0)
    {
        if(Bid-OrderOpenPrice()>Point*TrailingStop)
        {
            if(OrderStopLoss()<Bid-Point*TrailingStop)
            {
                //--- modify order and exit

if(!OrderModify(OrderTicket(),OrderOpenPrice(),Bid-
Point*TrailingStop,OrderTakeProfit(),0,Green))
                Print("OrderModify    error
",GetLastError());
                return;
            }
        }
    }
}
else // go to short position
{
    //--- should it be closed?
    if(PACurrent<0 && PACurrent>SignalCurrent &&
        PAPrevious<SignalPrevious    &&
MathAbs(PACurrent)>(PACloseLevel*Point))
    {
        //--- close order and exit

if(!OrderClose(OrderTicket(),OrderLots(),Ask,3,Violet))
        Print("OrderClose error ",GetLastError());
        return;
    }
    //--- check for trailing stop
    if(TrailingStop>0)
    {
        if((OrderOpenPrice()-Ask)>(Point*TrailingStop))
        {
            if((OrderStopLoss()>(Ask+Point*TrailingStop)) ||
(OrderStopLoss()==0))
            {
                //--- modify order and exit

```


ВІДГУК

на дипломний проект студента групи 121м-16-1 Клімова Д. Г.
на тему: «Удосконалення методики побудови автоматизованої торгової системи
з використанням патернів «Price Action»

Тема дипломного проекту – удосконалення методики побудови автоматизованої торгової системи з використанням патернів «Price Action».

Актуальність поставленого завдання обумовлюється широким попитом на автоматизовані торгові системи в сфері ритейлової ринкової діяльності.

Тема дипломного проекту прямо пов'язана з об'єктом діяльності 8.05010301. Програмна інженерія магістр з інформаційних технологій.

Зміст роботи включає аналіз існуючих недоліків в технології побудови автоматизованих торгових систем, які орієнтовані на широкого споживача. Були сформульовані та розв'язані проблеми недостатньої ефективності конкурентних пропозицій в сфері автоматизації торгівельної діяльності. При рішенні поставленої задачі використовувалися наукові досягнення в сфері технічного і фундаментального аналізу ринкової динаміки та в області розробки програмного забезпечення. Наукова новизна отриманих результатів полягає у доказі ефективного використання комбінованої методики аналізу ринкового ціноутворення.

Практична цінність полягає у високій ефективності прогнозування ринкової динаміки, завдяки використанню комбінації різноманітних засобів технічного аналізу, зокрема, патернів «Price Action». Це дає можливість вести стабільну та прибуткову торгову діяльність.

Ступінь опрацювання компонентів даного проекту, дозволяє оцінити роботу на «відмінно» і рекомендувати присвоїти студенту Д. Г. Клімову кваліфікацію «інженер-програміст».

Керівник дипломного проекту,
д.т.н., проф. кафедри ПЗКС

/Мещеряков Л.І./

РЕЦЕНЗИЯ

на дипломний проект студента групи 121м-16-1 Клімова Д. Г.
на тему: «Удосконалення методики побудови автоматизованої торгової системи з використанням патернів «Price Action»

У сучасному світі фінансові ринки відіграють велику роль в економічному розвитку окремих країн та світу. У другій половині 20 століття, після відміни «золотого стандарту», почало розвиватися вільне ціноутворення, яке залежить від попиту та пропозиції. Разом з цим з'явилась необхідність аналізу широкого спектру різноманітних факторів, які впливають на психологію ринку, щоб була можливість отримати прибуток від фінансових інвестицій. Розвиваються дві основні школи аналізу ринку: технічний та фундаментальний.

З розвитком обчислюваної техніки стає очевидним, що автоматизація торгової діяльності дозволяє уникнути різноманітних недоліків, що зв'язані з людськими факторами. Проте, в цей час, існують проблеми у сфері автоматизації торгових систем, які орієнтовані на широкого споживача. Доступні рішення від існуючих конкурентів або слабо ефективні, або поширюються шахраями.

В дипломному проекті в результаті проведеної роботи була розроблена, автоматизована торгова система, яка спроектована під найбільш поширену у світі інформаційно-торгівельну платформу MetaTrader 4.

При розробці торгового алгоритму студент Клімов Д. Г. використовував відомості про різні види аналізу ринкового ціноутворення, серед яких, за допомогою тестування, вибрав найбільш ефективні. Основною методикою пошуку торгових сигналів обрана школа «Price Action». У поєднанні з іншими методами технічного була досягнута висока ефективність торгових сигналів, які генеруються розробленою торговою системою.

В підсумку, можна зробити висновок, що даний проект цілком відповідає вимогам, що пред'являються до кваліфікаційних робіт рівня магістра.

В цілому проведений аналіз використаних компонентів даного проекту, дозволяє оцінити роботу на «відмінно» і рекомендувати присвоїти студенту Д. Г. Клімову кваліфікацію «Інженер-програміст».

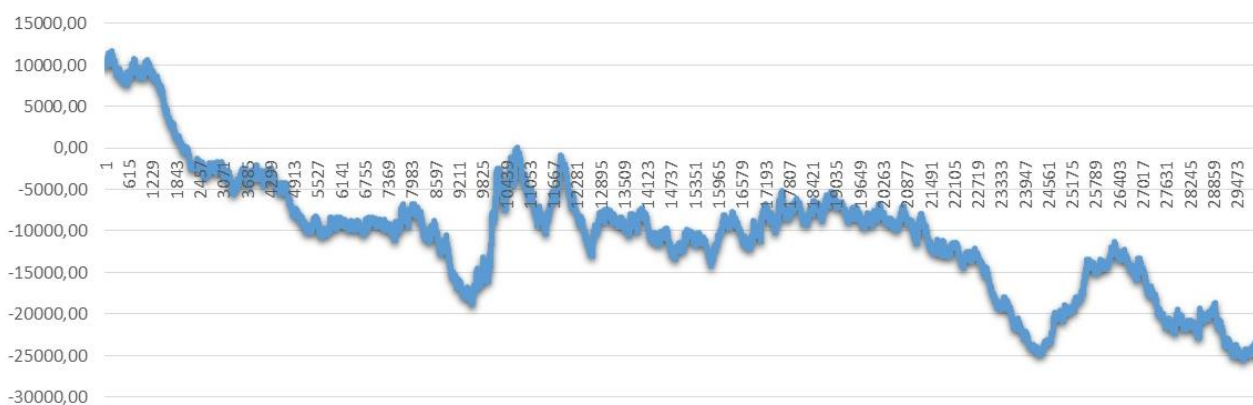
Рецензент

РЕЗУЛЬТАТИ АНАЛІЗУ ЕФЕКТИВНОСТІ ПАТЕРНІВ «PRICE ACTION»
на таймфреймах Н1 та Н4

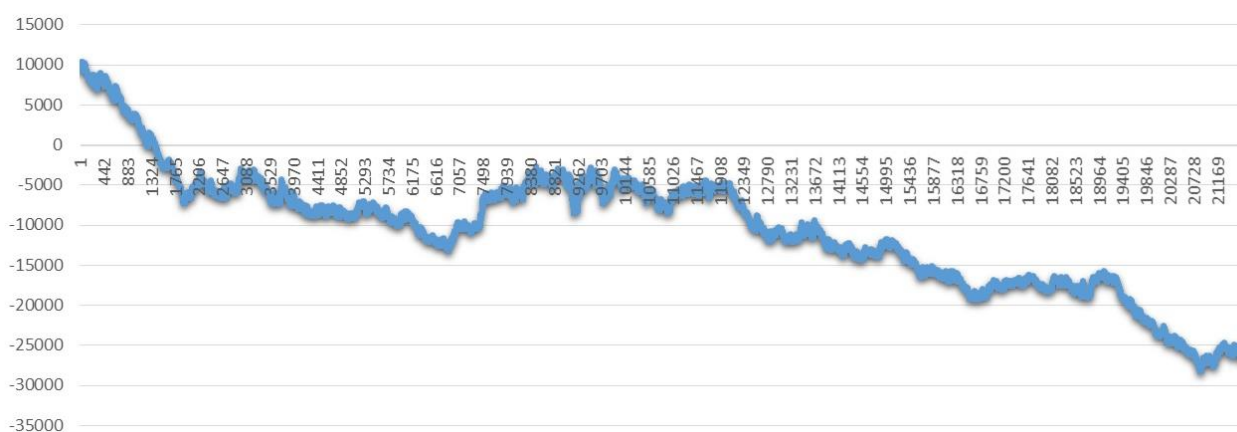
Період Н1

Графики прибутковості:

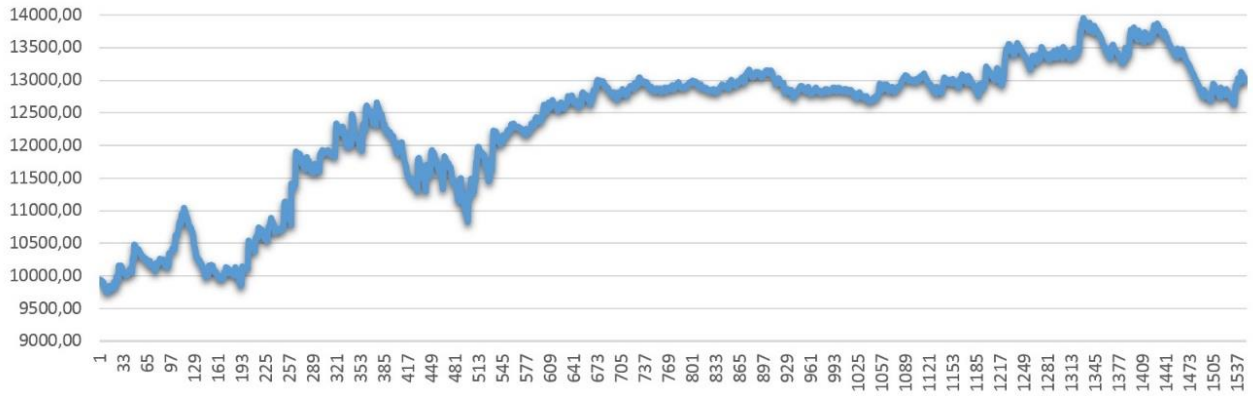
Патерн «Внутрішній бар»



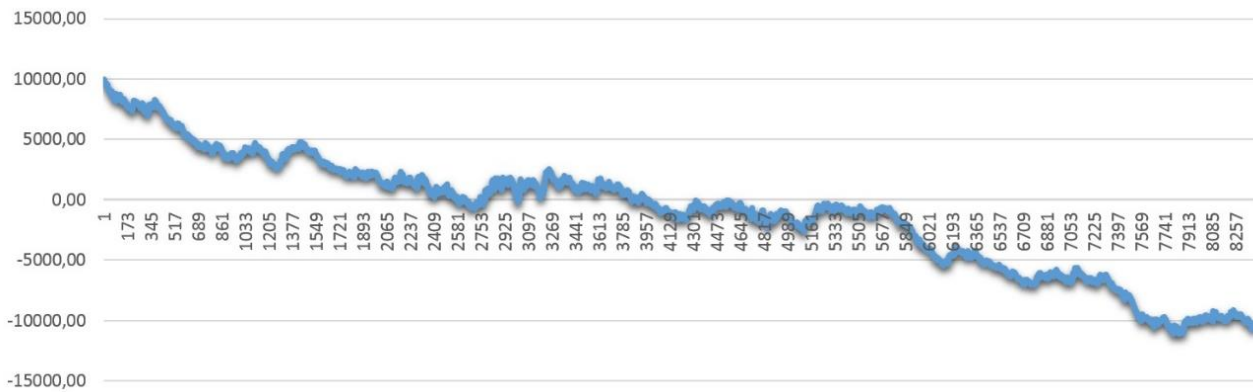
Патерн «Доджи»



Патерн «Пінбар»

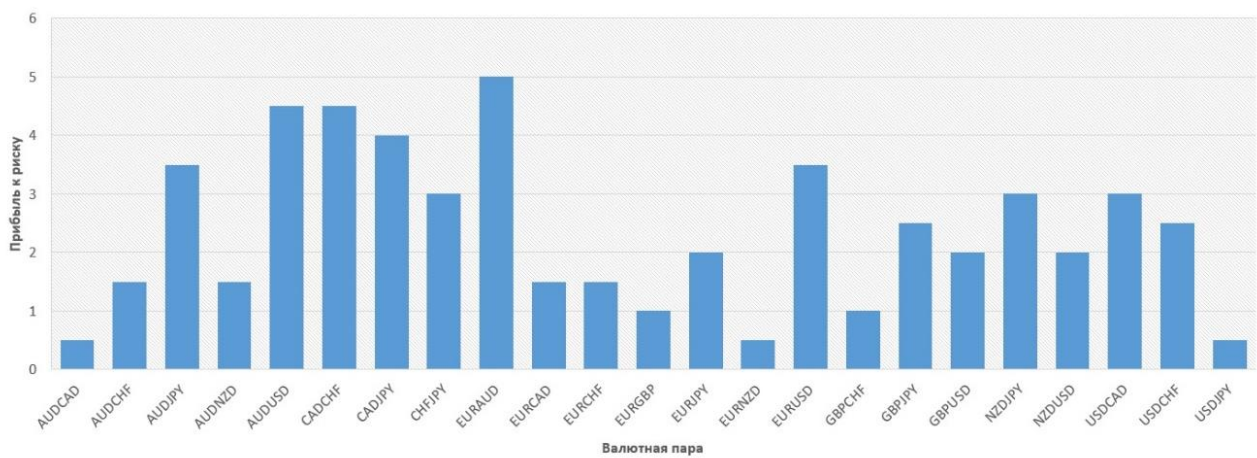


Патерн «Поглинання»

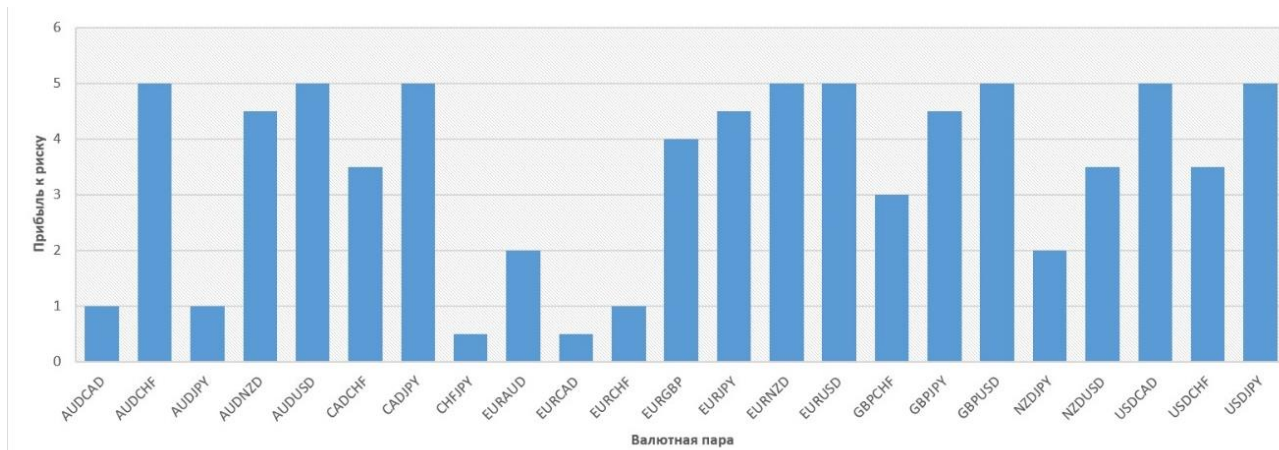


Оптимальные отношения стопов к тейк профитам:

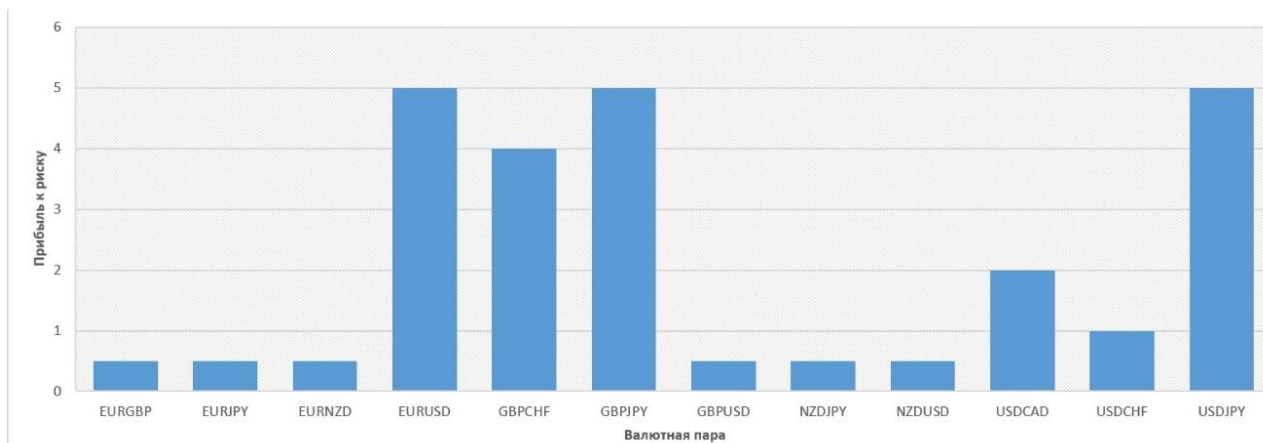
Патерн «Внутрішній бар»



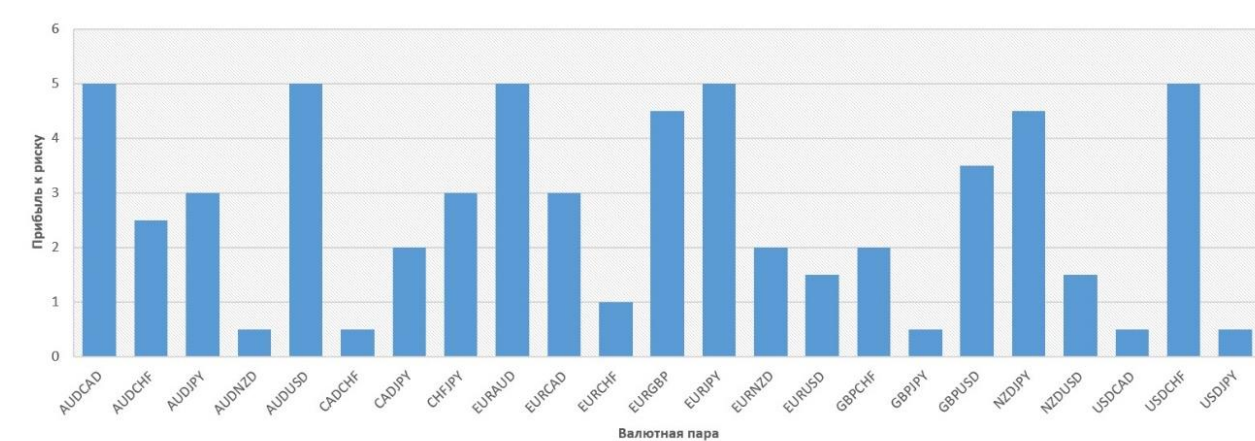
Патерн «Доджи»



Патерн «Пінбар»

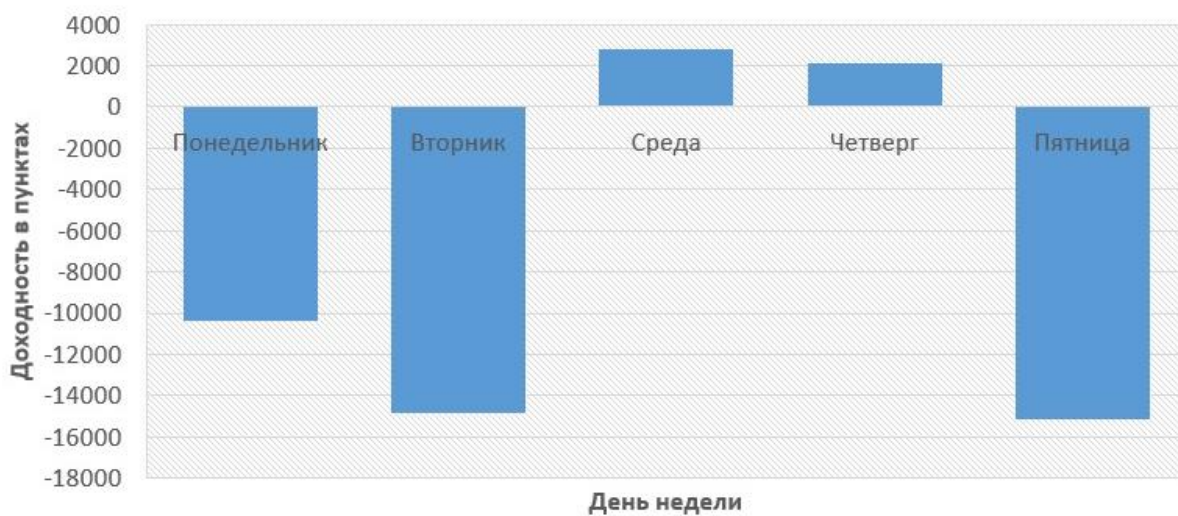


Патерн «Поглинання»

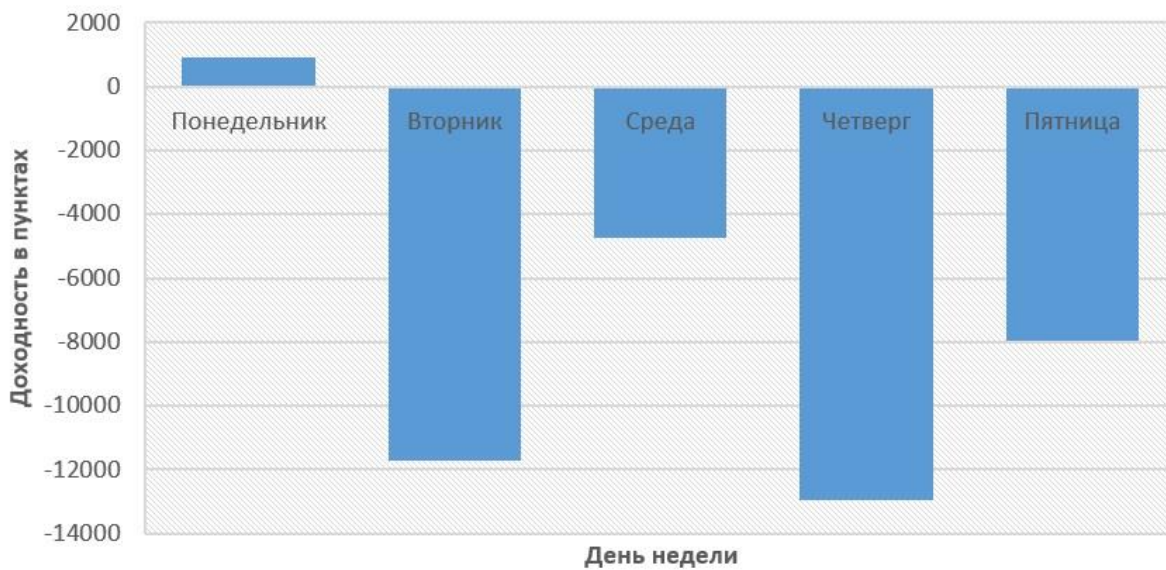


Доходность по дням недели:

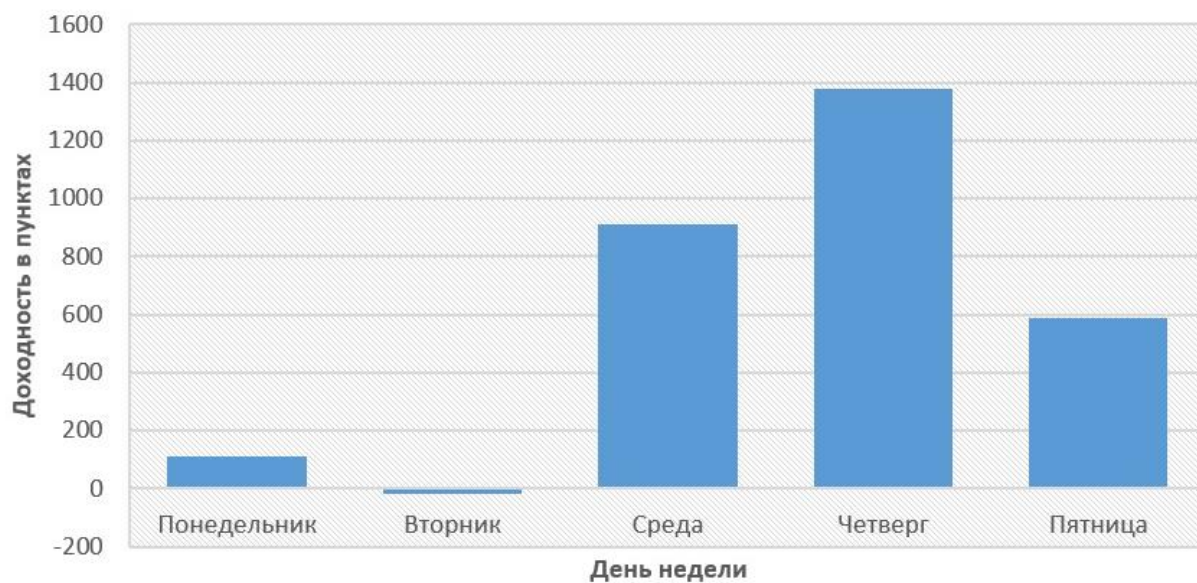
Патерн «Внутрішній бар»



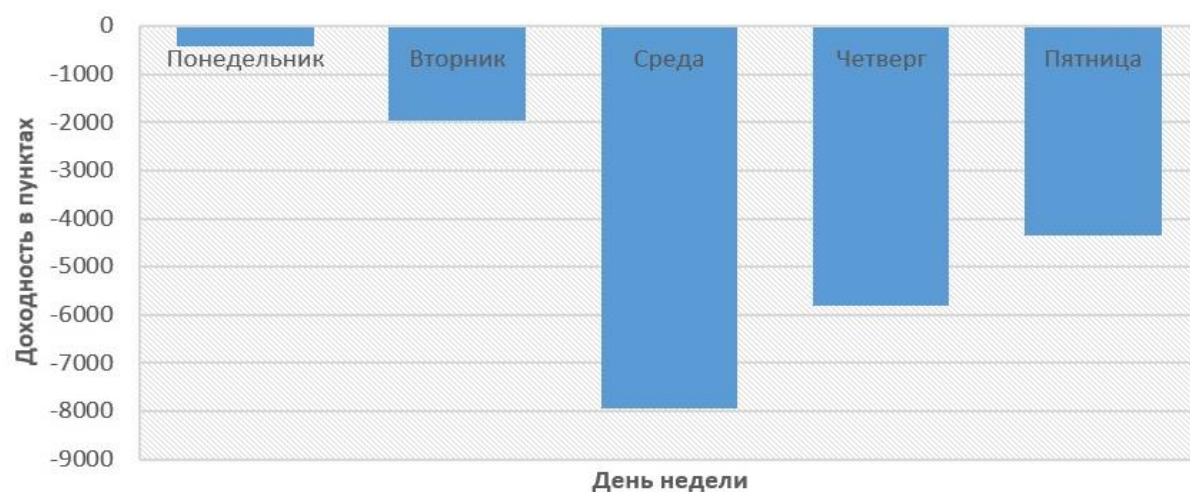
Патерн «Доджи»



Патерн «Пінбар»

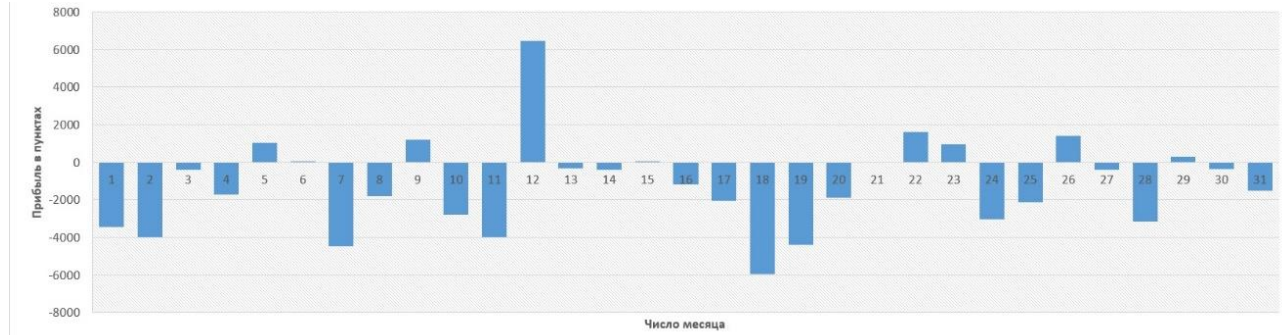


Патерн «Поглинання»

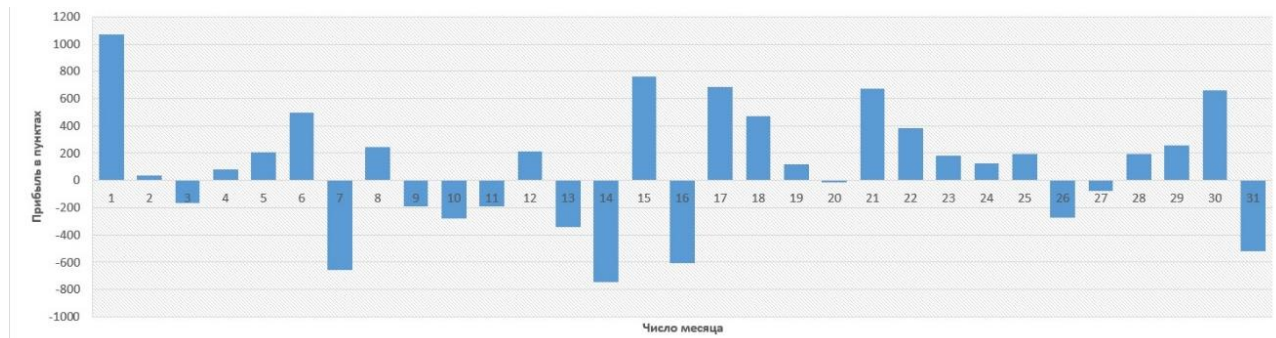


Доходность по числам месяца

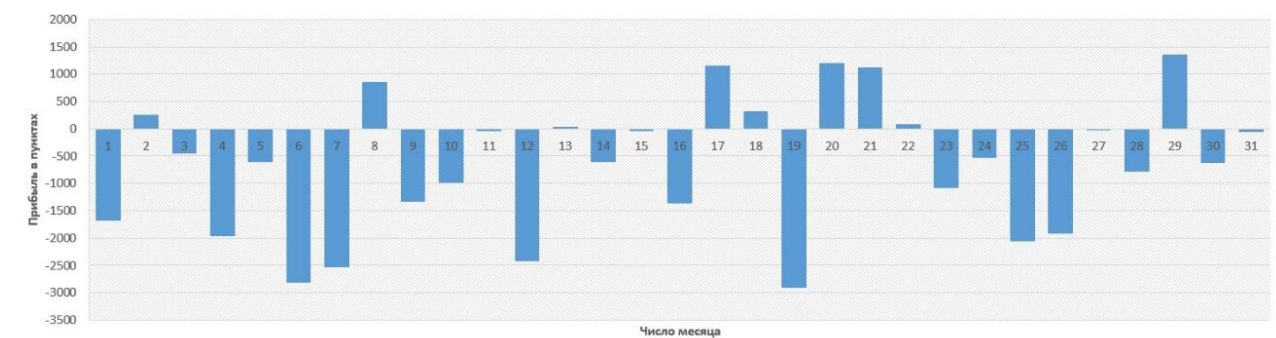
Патерн «Внутрішній бар»



Патерн «Доджи»

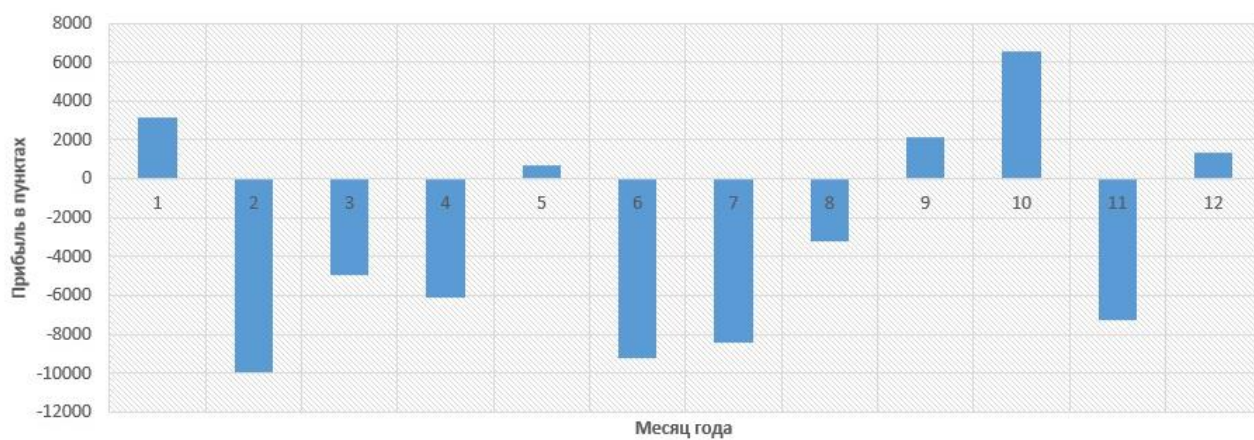


Патерн «Пінбар»

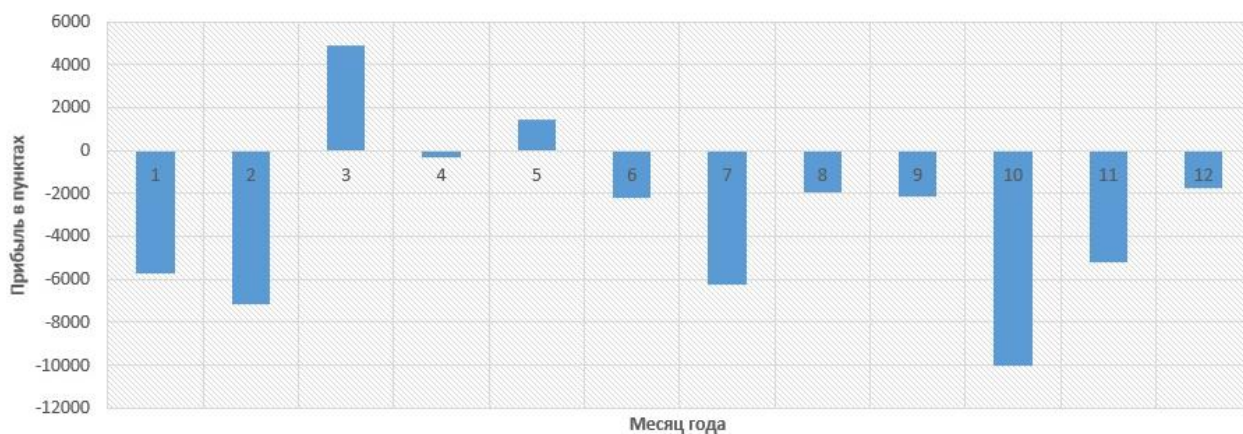


Доходность по месяцам года:

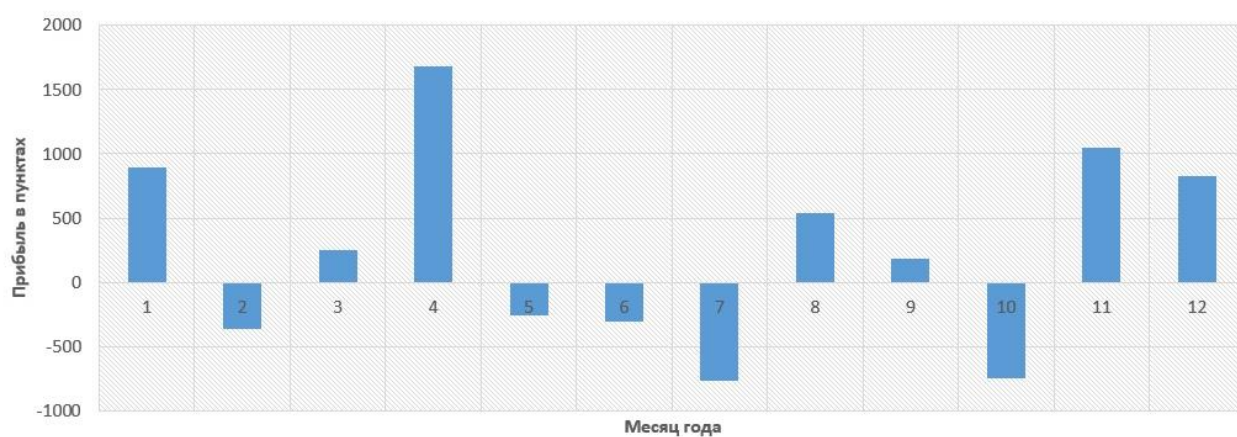
Патерн «Внутрішній бар»



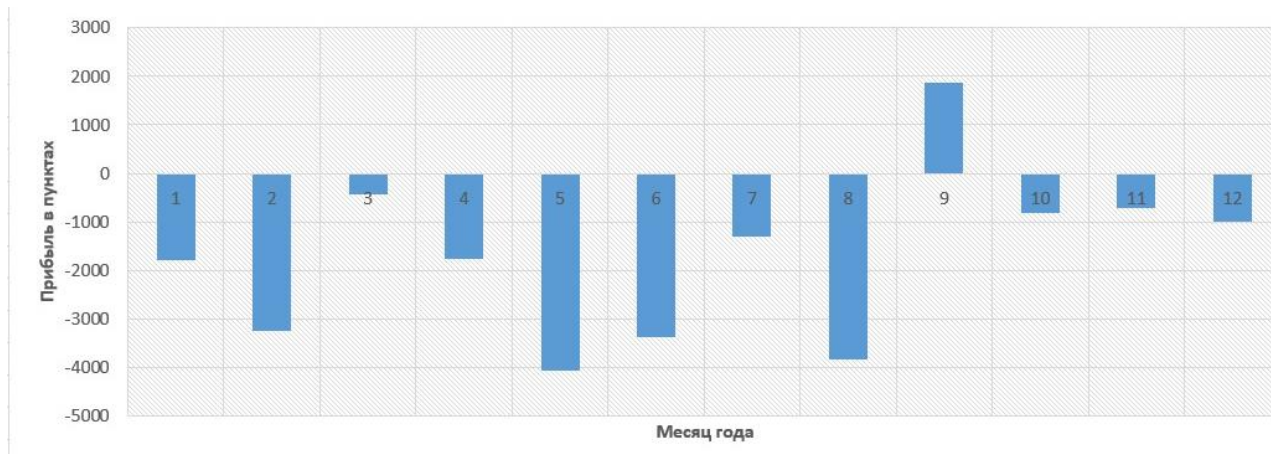
Патерн «Доджи»



Патерн «Пінбар»



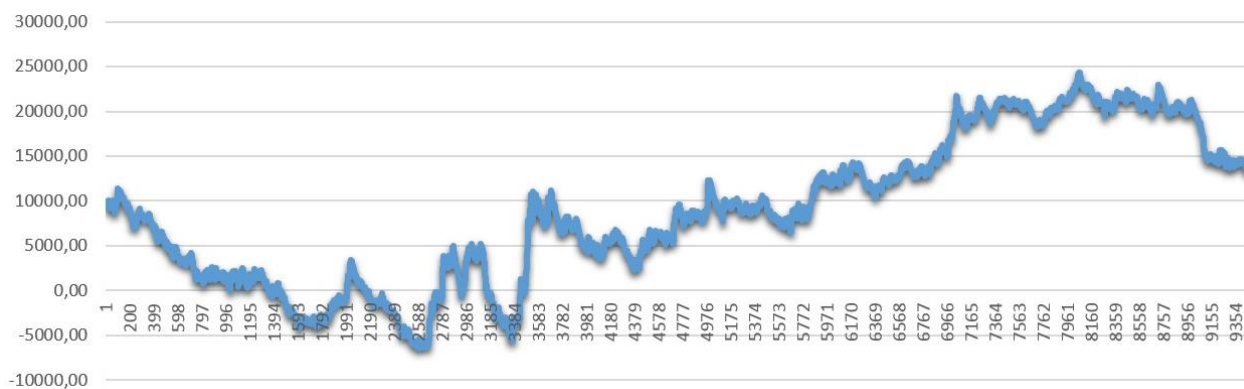
Патерн «Поглинання»



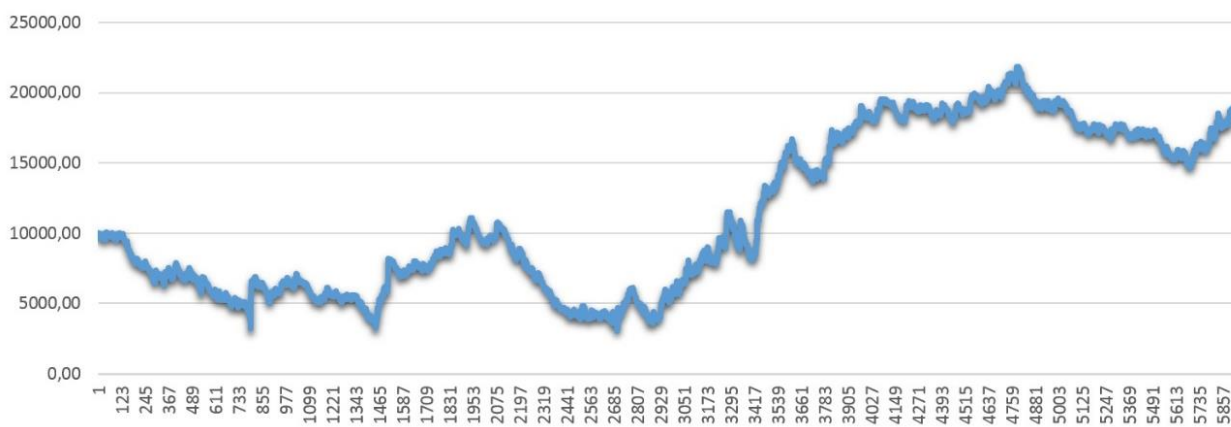
Період Н4

Графики доходностей:

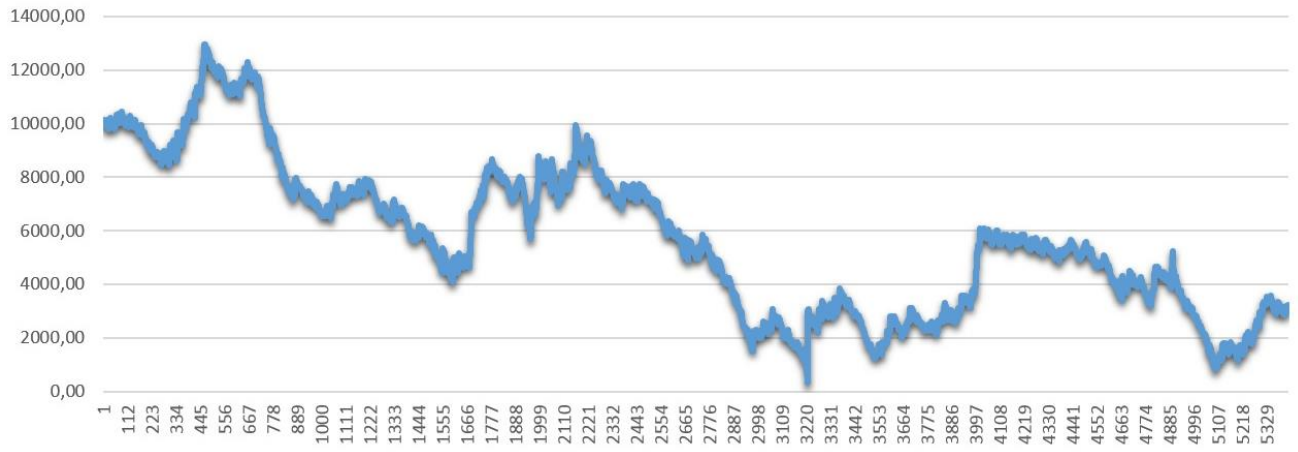
Патерн «Внутрішній бар»



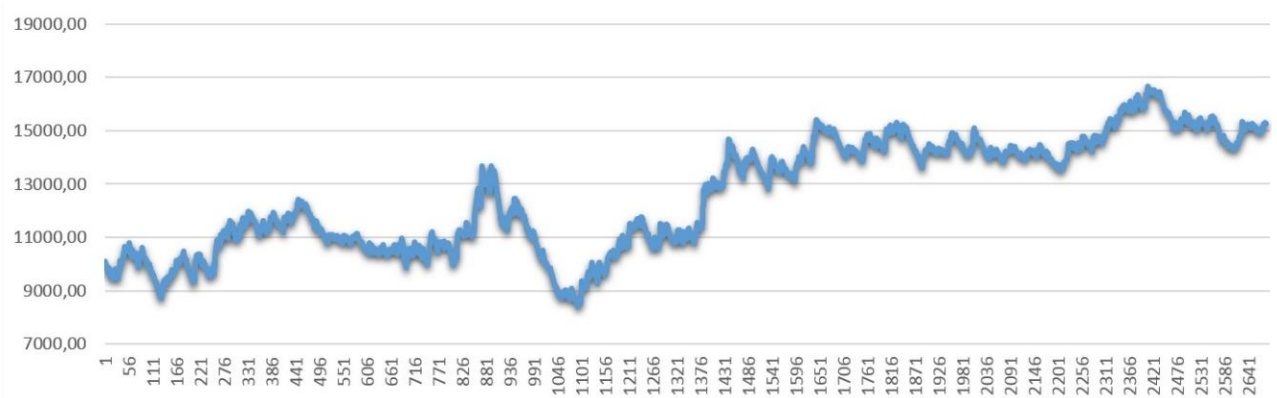
Патерн «Доджи»



Патерн «Пінбар»

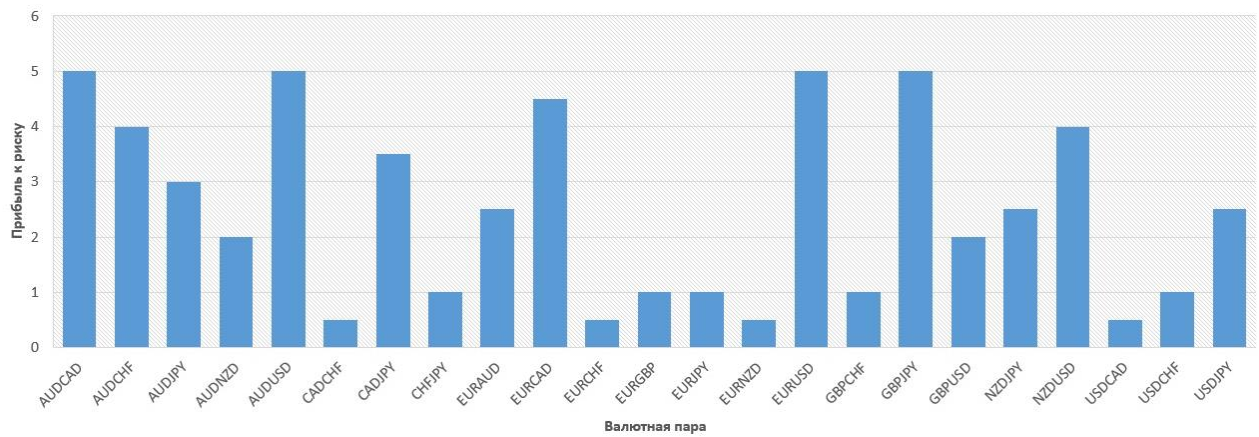


Патерн «Поглинання»

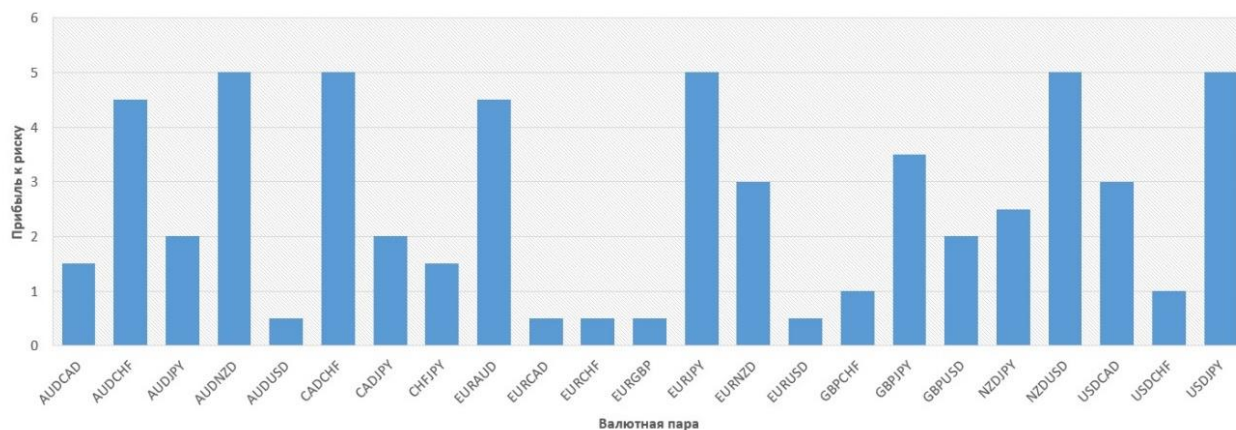


Оптимальные отношения стопов к тейк профитам:

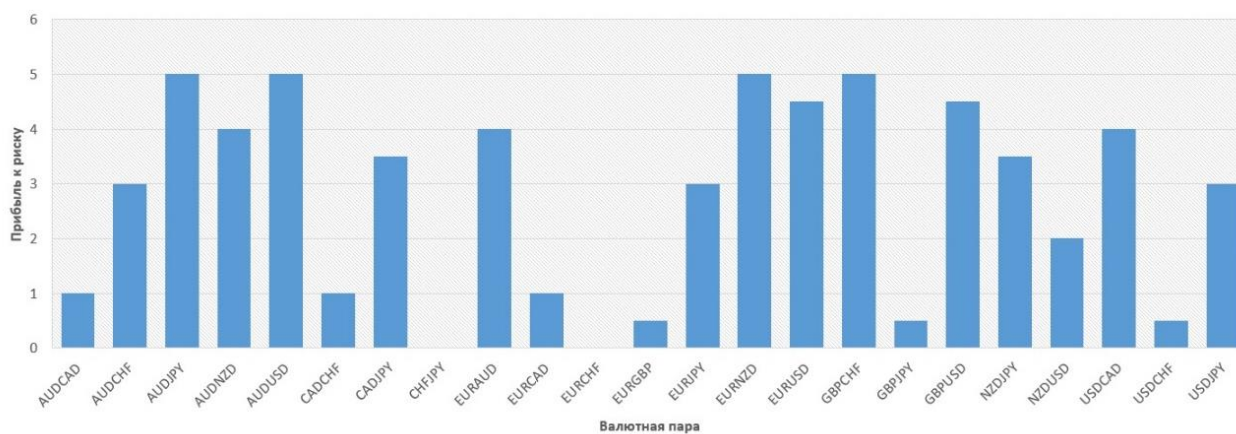
Патерн «Внутрішній бар»



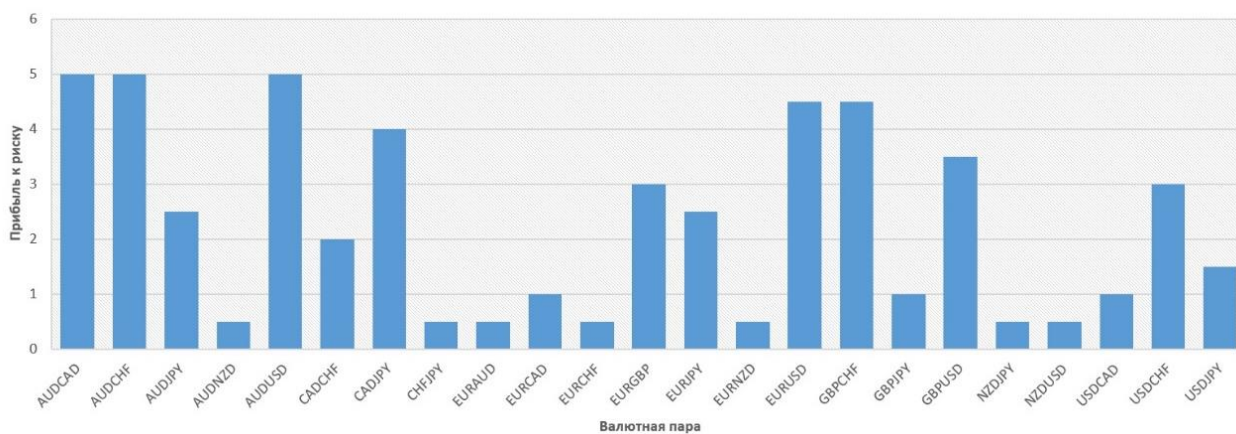
Патерн «Доджи»



Патерн «Пінбар»

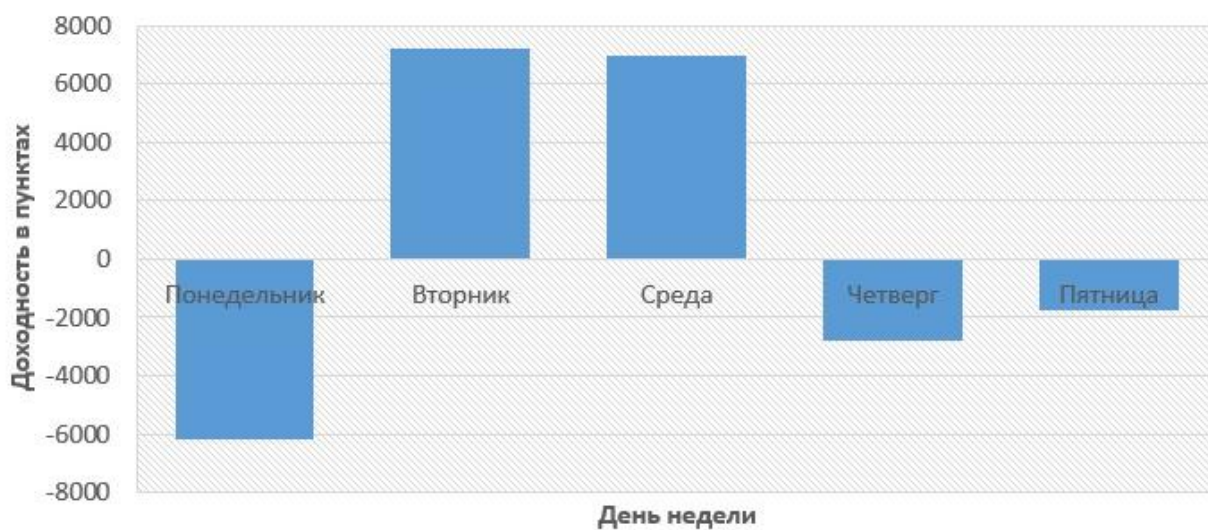


Патерн «Поглинання»

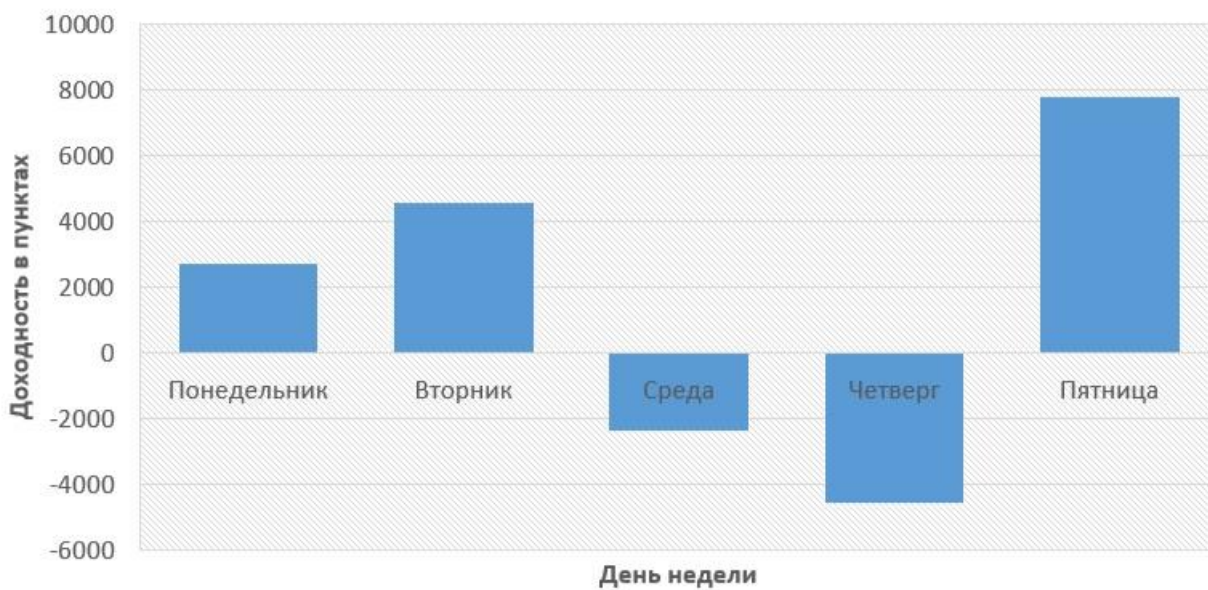


Доходность по дням недели:

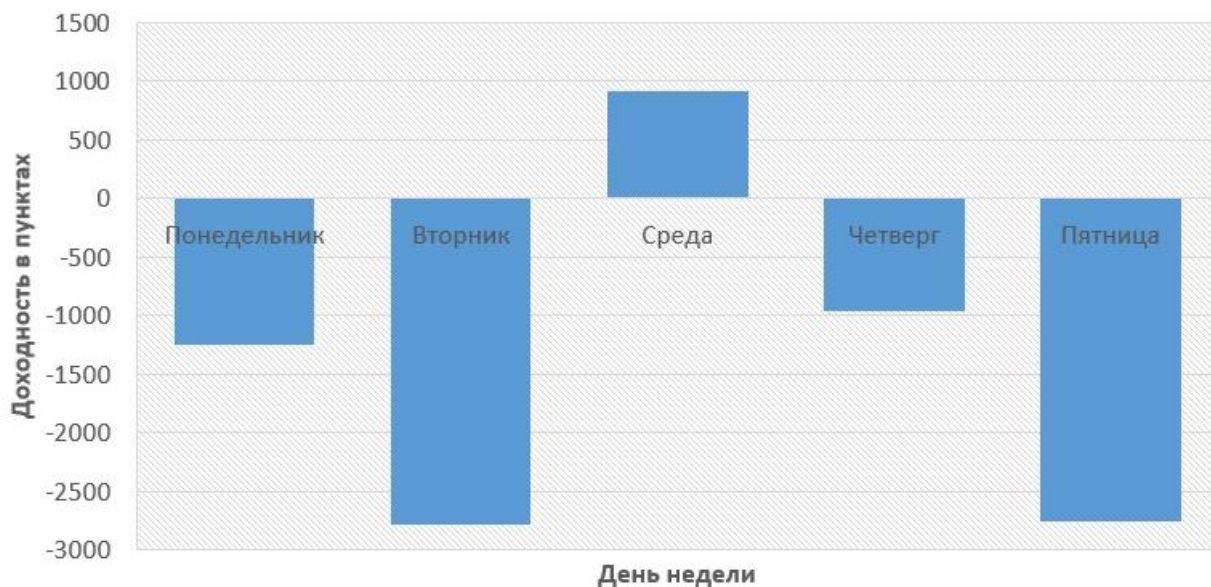
Патерн «Внутрішній бар»



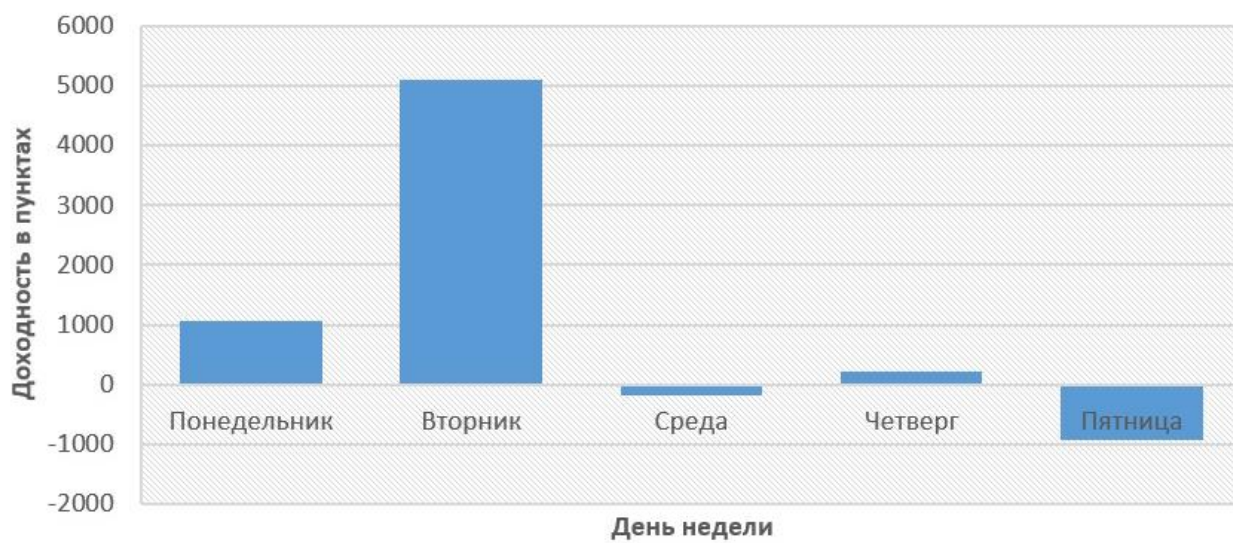
Патерн «Доджи»



Патерн «Пінбар»

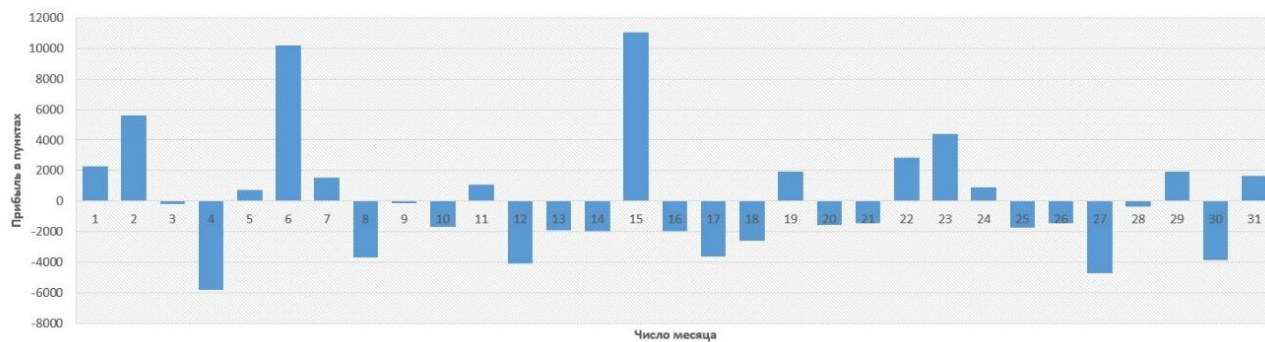


Патерн «Поглинання»

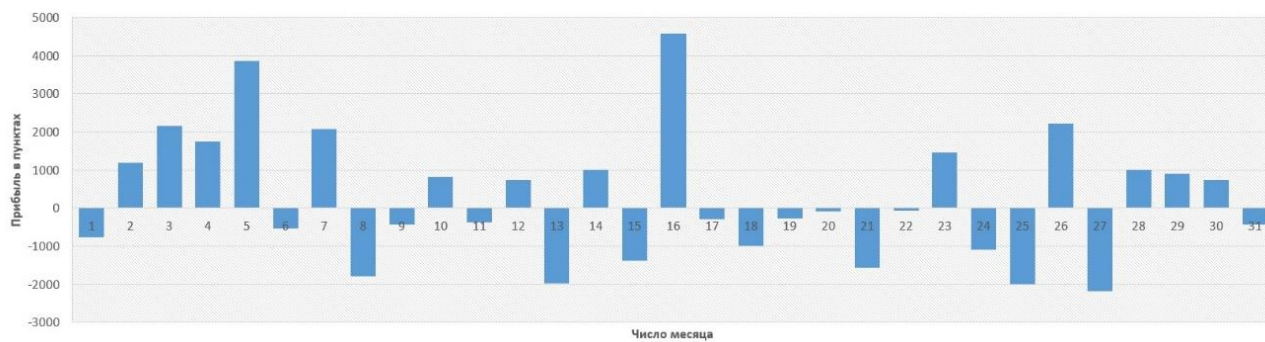


Доходность по числам месяца:

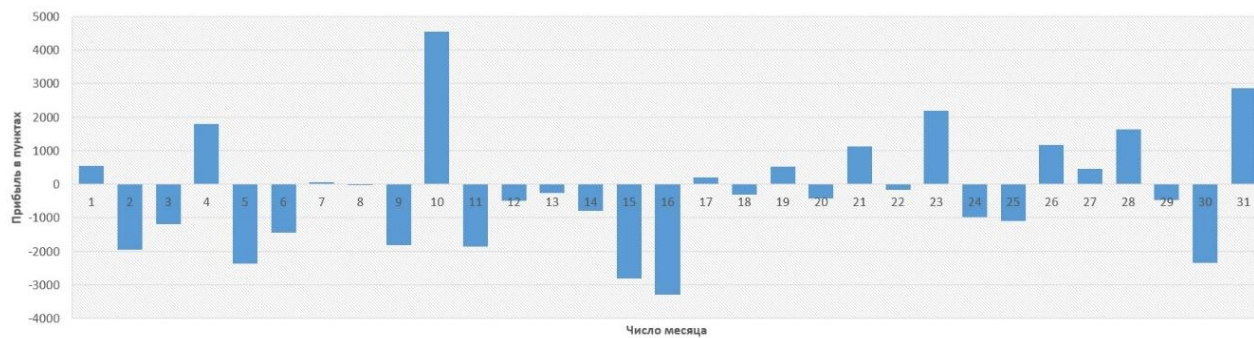
Патерн «Внутрішній бар»



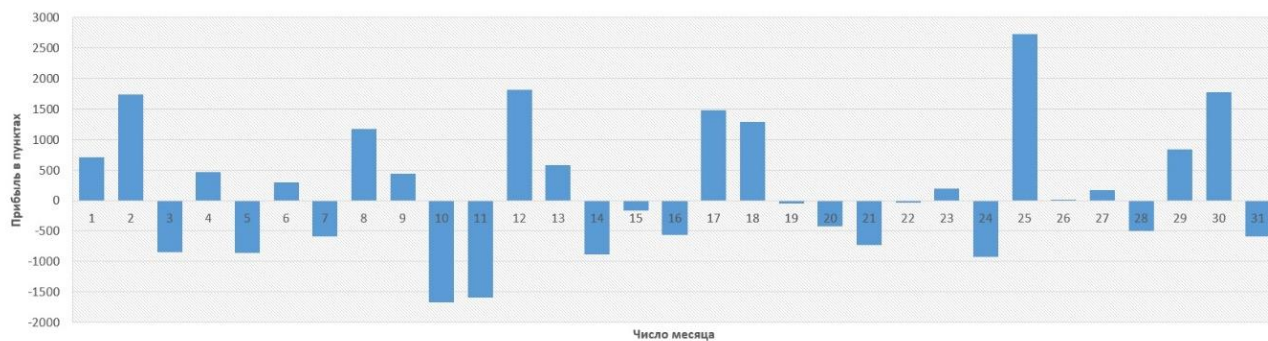
Патерн «Доджи»



Патерн «Пінбар»

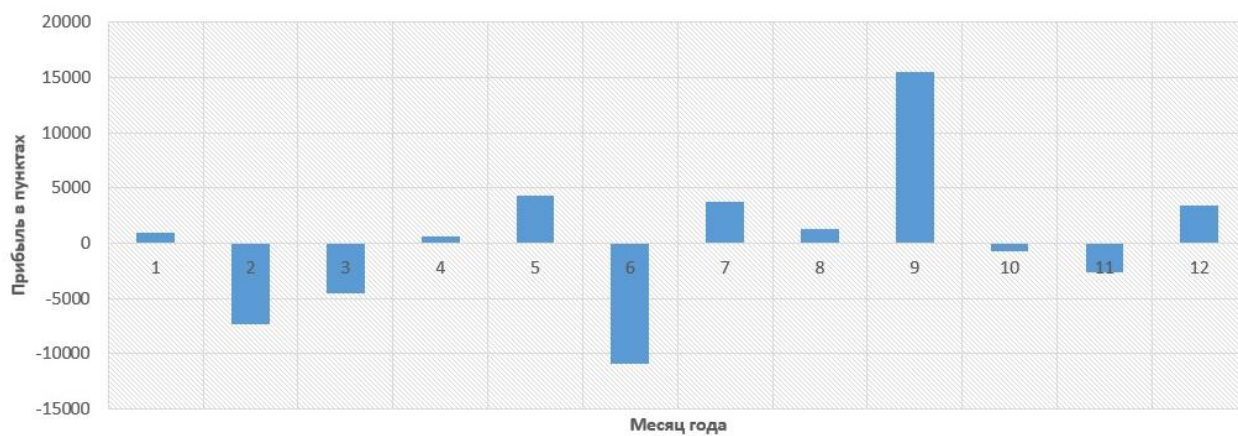


Патерн «Поглинання»

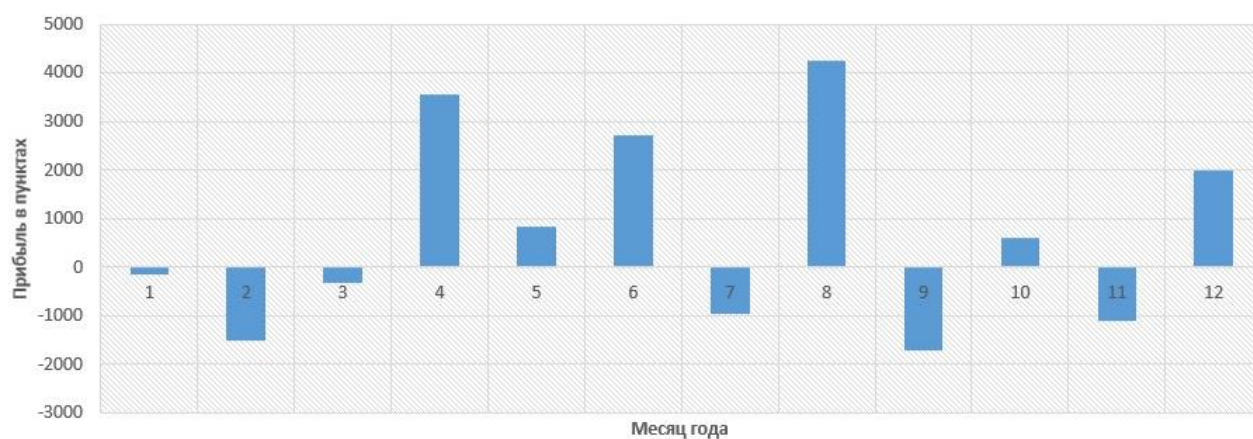


Доходність по місяцям года:

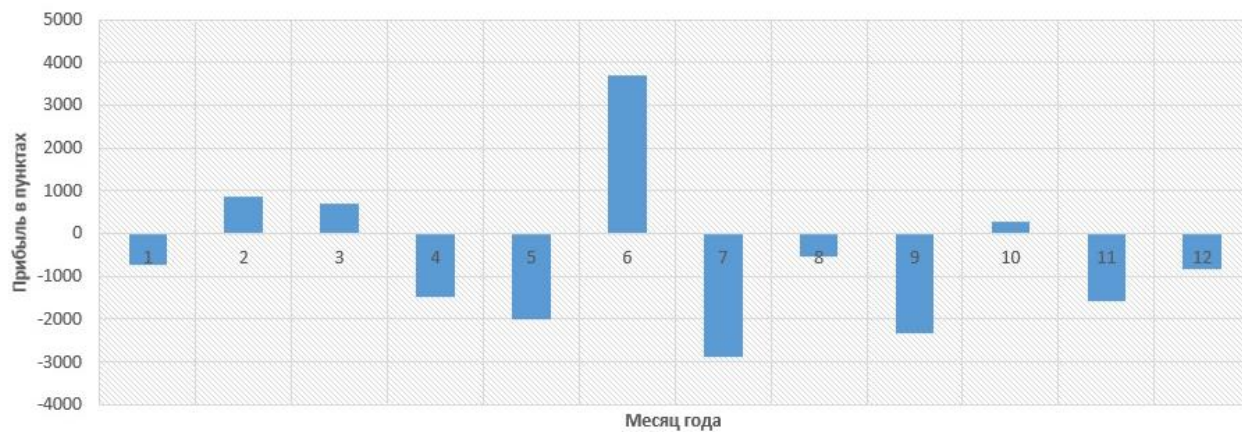
Патерн «Внутрішній бар»



Патерн «Доджи»



Патерн «Пінбар»



Патерн «Поглинання»

