

И. А. КАЗАК, канд. педаг. наук, ассистент, НТУУ «КПИ», Киев

ИСПОЛЬЗОВАНИЕ ВОЗМОЖНОСТЕЙ VBA ДЛЯ УМЕНЬШЕНИЯ ТРУДОЕМКОСТИ СОЗДАНИЯ ПРОГРАММ

В статье предлагается способ автоматического создания программы с помощью макрорекордера в VBA на примере построения графика функции. Программы, написанные таким способом, можно многократно использовать с возможностью их дальнейшего редактирования, что позволит уменьшить трудоемкость работы по созданию программ с типичными действиями для студентов, инженеров, исследователей. Показано, что возможности VBA упрощают также написание и редактирование программ начинающим пользователям.

Ключевые слова: программа, макрорекордер, макрос, VBA, график, аргумент, функция, создание, проект, трудоемкость

Введение. На сегодняшний день, одной из актуальных проблем является трудоёмкость создания программ. Одним из способов решения данной проблемы является возможность автоматического создания программ с помощью макрорекордера в VBA.

VBA (Visual Basic for Applications) - это диалект языка программирования Visual Basic, который расширяет его возможности и предназначен для работы с приложениями Microsoft Office и другими приложениями от Microsoft и других компаний [1–4].

В большинство программ Microsoft Office встроено замечательное средство, которое позволит вам создавать программы, вообще ничего не зная о программировании. Это средство называется макрорекордер.

Цель работы. Целью работы является ознакомление с возможностями макрорекордера в VBA для автоматического создания программы (на примере построения графика заданной функции по «Информатике» для учебных и исследовательских процессов), которые могут применяться для любого рода типичных многократно повторяющихся действий (расчетов) студентами, инженерами, исследователями для получения, анализа, обработки различной информации и т.п.

Методика экспериментов автоматического создания программ в VBA. Макрорекордер, как понятно из его названия, - это средство для записи макросов. Макрос - всего лишь еще одно название для VBA - программы, а макрорекордер - средство для его автоматического создания [5–7].

Приложения Microsoft Office по умолчанию настроены так, что не позволяют запускать макросы. Поэтому перед тем, как приступить к созданию макросов, в меню Сервис | Макрос | Безопасность переставьте переключатель «Уровень безопасности» в положение «Средняя» или «Низкая», закройте и снова откройте программу. Это нужно сделать только один раз в начале работы.

Принцип работы макрорекордера больше похож на принцип работы магнитофона: мы нажимаем на кнопку - начинается запись тех действий, которые

мы выполняем. Мы нажимаем на вторую кнопку - запись останавливается, и мы можем его проиграть.

Чтобы создать макрос в макрорекордере, необходимо:

1. В меню Сервис | Макрос выберите «Начать запись». В окне «Запись макроса» вам потребуется определить:

- имя макроса (правило такое: имя не должно начинаться с цифры, не должно содержать пробелы и символы пунктуации);

- где сохранить макрос (текущая книга, создать макрос одновременно с созданием новой книги и личная книга макросов PERSONAL.XLS, макросы из которой будут доступны во всех книгах).

2. После нажатия кнопки ОК или назначения кнопки или клавиатурной комбинации начнется запись макроса. Появится маленькая панель ■ «Остановить запись» [8-10].

В дальнейшем у пользователя может возникнуть желание внести в макрос любые дополнения или исправления. Самый простой, но неудобный способ - воспользоваться окном меню Сервис | Макрос | Макросы | Изменить. Если вы пользуетесь макросом постоянно, то можно использовать самый быстрый способ его вызова - клавиатурную комбинацию.

Полные возможности программирования в Office раскрываются при использовании редактора Visual Basic. Для входа в него в меню Сервис | Макрос выбрать «Редактор Visual Basic», или нажать клавиши <Alt> + <F11>.

В верхнем левом углу обычно открыто окно проводника проекта. Если оно случайно было закрыто, то вызвать его можно, нажав клавиши <Ctrl> + <R>.

Самый верхний уровень - это проект (Project), которому соответствует документ Word, рабочая книга Excel и т. д. Проекты VBA выполняются вместе с другими приложениями. Приложение, в котором разрабатывается и выполняется проект VBA, называется основным. Например, можно создать проект VBA, который работает вместе с Microsoft Excel. В этом случае Excel является основным приложением. Фактически, используя основное приложение, нельзя построить приложение VBA [1].

VBA имеет собственную среду разработки, называется интегрированной средой разработки или IDE. VBA IDE - это окно, содержащее меню, другие окна и элементы, применяемые при создании проектов VBA [2]. Ниже меню VBA расположены программные модули. При работе макрорекордера в Excel – это Module1, куда и записываются все создаваемые макрорекордером макросы.

Вывести данные программы в VBA можно прямо в ячейку Excel.

После того, как нужная ячейка найдена, для записи в нее используется свойство Value, например: x.Value = значение.

Если ячейка уже выбрана, можно записать: ActiveCell.Value = значение.

Или сразу указать ячейку вывода: Cells (6, 1) = значение.

Выбрать ячейку, смещенную относительно текущей на i вниз и на j вправо:

ActiveCell.Offset (i, j) .Select

Range - очень важная функция. Возвращает объект Range, который представляет собой диапазон ячеек и используется в Excel практически для любых операций с ячейками.

Выбрать какую-либо ячейку, например, A6: Range ("A6"). Select.

Или диапазон ячеек: Range ("A1: A6"). Select

Прежде чем что-то выводить в таблицу, полезно очистить письмо, для чего служит команда Cells.Clear.

Formula (или FormulaR1C1) = "=" позволяет записать формулу в ячейку.

Пример:

Range ("A3"). Formula = "= 2 * \$ A \$ 1 ^ 2 + cos (\$ B \$ 1)" (абсолютные ссылки) или "= 2 * A1 ^ 2 + cos (B1)" (относительно ссылки).

ActiveCell.FormulaR1C1 = "= 2 * RC [-1] ^ 2 + cos (RC [-1])"

В формулу могут быть включены и переменные. Например, вместо 2 - используем переменную a. Тогда предварительную формулу следует переписать:

ActiveCell.FormulaR1C1 = "=" & a & "* RC [-1] ^ 2 + cos (RC [-1])" [7].

Рассмотрим автоматическое создание программы макрорекордером в VBA на примере задания по «Информатике» для бакалавров машиностроительных и других специальностей - построение графика заданной функции, например: $y(x) = -48 \cdot x^3 - 20 \cdot x^2 + 7 \cdot x - 13$. Научиться выполнять это задание при изучении курса «Информатика» очень актуально, так как построение графиков приходится выполнять очень часто и многократно студентам, будущим инженерам и ученым, необходимых при проведении научных исследований по техническим наукам.

Чтобы создать автоматически любую программу (макрос) макрорекордером в VBA, необходимо в меню Microsoft Office Excel 2010 выбрать Сервис | Макрос и нажать «Начать запись». В окне «Запись макроса» написать: имя макроса (не должно начинаться с цифры, содержать пробелы и символы пунктуации) и где сохранить макрос (текущая книга). После этого нажать кнопку ОК, чтобы началась запись макроса и в момент начала записи появится маленькая панель ■ - «Остановить запись», которой мы воспользуемся по завершении записи макроса [1].

Включив запись макроса, выполнить следующие действия в текущей книге Excel по построению графика заданной функции $y(x) = -48 \cdot x^3 - 20 \cdot x^2 + 7 \cdot x - 13$:

- ввести в ячейки A1 и A2 первое и второе численные значения аргумента x;
- выделить ячейки A1 и A2 с численными значениями x и за правый нижний угол ячейки A2 растягиваем вниз на диапазон значений аргумента x, при котором хотим посчитать заданную функцию y;

- в ячейке B1 ввести формулу функции $y(x) = -48 \cdot x^3 - 20 \cdot x^2 + 7 \cdot x - 13$, затем растянуть за правый нижний угол ячейки B1 вниз на диапазон заданных значений аргумента x и таким образом получаем в столбце B автоматически посчитанные численные значения функции y для заданного диапазона значений x;

- выделить столбцы A и B с заполненными ячейками, для этого нажать на левую кнопку мыши, не отпуская ее, протянуть сверху вниз до последних численных значений x и y. По выделенным значениям у строим график в Excel, для этого выбираем в меню Вставка | Диаграммы | Точечная диаграмма. Результат построения графика заданной функции представлен на рис. 1.

- останавливаем запись макроса макрорекордером по построению графика заданной функции, нажимая на ■ - «Остановить запись».

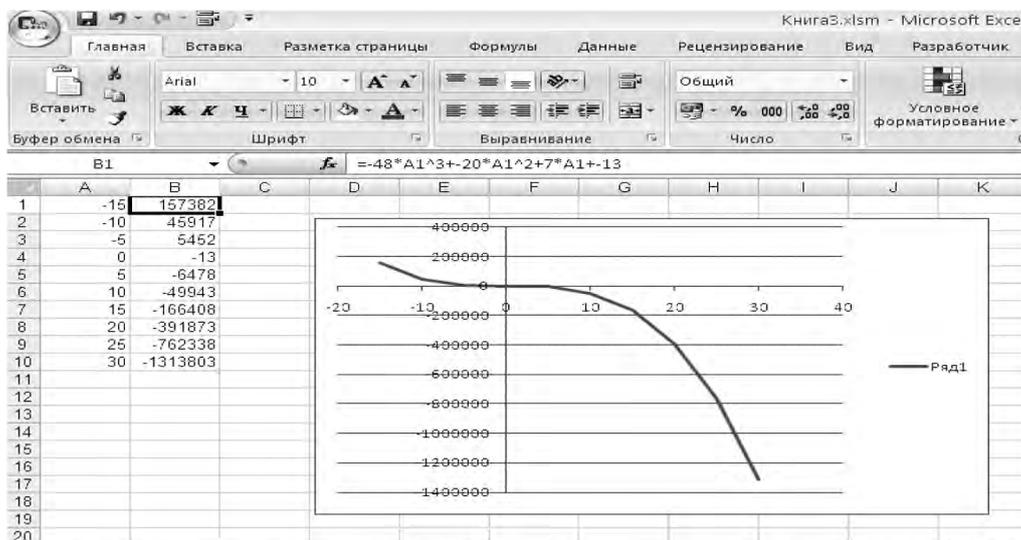


Рис. 1 - Численные значения аргумента x и функции $y(x) = -48 \cdot x^3 - 20 \cdot x^2 + 7 \cdot x - 13$ на заданном диапазоне и построенный точечный график для автоматической записи программы

Обсуждение результатов записанного макроса макрорекордером по построению графика функции в VBA. Проверяем работу автоматически созданной программы в VBA по построению графика рассмотренной функции $y(x) = -48 \cdot x^3 - 20 \cdot x^2 + 7 \cdot x - 13$:

- удаляем в текущей книге Excel все, что было нами записано в столбцах А и В для построения графика заданной функции и построенный точечный график, для этого выделяем всю область численных значений x и y и полученный график функции, нажимаем клавишу Delete;

- запускаем автоматически написанную программу (макрос) по построению графика функции, для этого в меню Microsoft Office Excel 2010 выбрать Сервис | Макрос и выбрать «Выполнить».

Автоматическая и записанная программа по построению графика функции представлена в окне VBA в подпрограмме Sub Mac () на рис. 2;

```

Microsoft Visual Basic - Книга1 - [Module1 (Code)]
Project - VBAProject
  VBAProject (Книга1)
    Microsoft Excel Objects
      Лист1 (Лист1)
      Лист2 (Лист2)
      Лист3 (Лист3)
    ЭтаКнига
    Modules
      Module1

Properties - Module1
Module1 Module
  Alphabetic | Categorized
  (Name) Module1

(General)
Sub Mac ()
  Mac Макрос

  Range("A1").Select
  ActiveCell.FormulaR1C1 = "-15"
  Range("A2").Select
  ActiveCell.FormulaR1C1 = "-10"
  Range("A1:A2").Select
  Selection.AutoFill Destination:=Range("A1:A10"), Type:=x1FillSeries
  Range("A1:A10").Select
  Range("B1").Select
  ActiveCell.FormulaR1C1 = "--48*RC[-1]^3-20*RC[-1]^2+7*RC[-1]-13"
  Range("B1").Select
  Selection.AutoFill Destination:=Range("B1:B10"), Type:=x1FillValues
  Range("B1:B10").Select
  ActiveSheet.Shapes.AddChart.Select
  ActiveChart.SetSourceData Source:=Range("'Лист1'!$B$1:$B$10")
  ActiveChart.ChartType = x1XYScatterLines
End Sub
  
```

Рис. 2 - Автоматически написанная программа макрорекордером по построению графика функции $y(x) = -48 \cdot x^3 - 20 \cdot x^2 + 7 \cdot x - 13$

- в текущей книге Excel видим записанные значения x и y в столбцах А и В и построенный по ним точечный график заданной функции $y(x) = -48 \cdot x^3 - 20 \cdot x^2 + 7 \cdot x - 13$, которые выполнила автоматически созданная макрорекордером программа (макрос) после его написания для построения графика этой функции (рис. 3).

При желании можно внести в автоматически написанную программу (макрос) любые дополнения или исправления. Самый простой способ - воспользоваться окном меню Сервис | Макрос | Макросы | Изменить.

Теперь с помощью автоматически полученной программы для построения графика

функции, изменив, например, коэффициенты при аргументе x (или полностью написав новую функцию), можно запустить на выполнение эту программу, и она сама построит график функции с измененными коэффициентами (или функцией). При этом не нужно заново выполнять все действия по построению графика измененной функции, а эти действия многократно будет выполнять автоматически написанная программа по построению графика с внесенными изменениями в программу.

Например, в ранее автоматически написанной программе Sub Mac() макрорекордером по построению графика заданной функции $y(x) = -48 \cdot x^3 - 20 \cdot x^2 + 7 \cdot x - 13$ поменяем коэффициенты в уравнении функции при аргументе x в полиноме третьей степени на 48, в полиноме второй степени - на 20 и получим уже другую функцию $y(x) = 48 \cdot x^3 + 20 \cdot x^2 + 7 \cdot x - 13$ (рис. 4).

Уделите несколько минут внимательному изучению программы, которую сгенерировал Excel. Если вы новичок, вклад нескольких минут в изучение автоматически написанной программы макрорекордером дадут потрясающий результат в знакомстве с объектами VBA в дальнейшем.

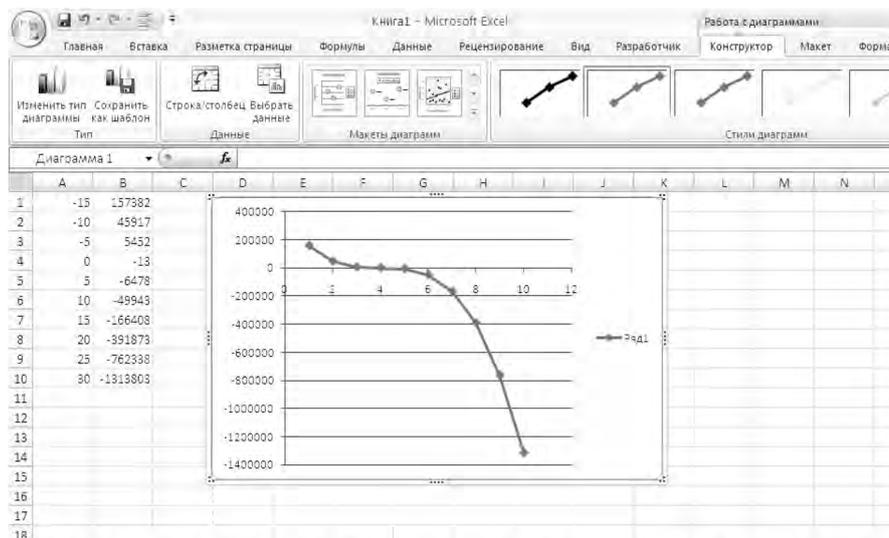


Рис. 3 - Численные значения аргумента x и функции $y(x) = -48 \cdot x^3 - 20 \cdot x^2 + 7 \cdot x - 13$ на заданном диапазоне и построенный точечный график, выполненные макросом

```

Sub Mac ()
    Mac Макрос

    Range("A1").Select
    ActiveCell.FormulaR1C1 = "-15"
    Range("A2").Select
    ActiveCell.FormulaR1C1 = "-10"
    Range("A1:A2").Select
    Selection.AutoFill Destination:=Range("A1:A10"), Type:=xlFillSeries
    Range("A1:A10").Select
    Range("B1").Select
    ActiveCell.FormulaR1C1 = "=48*RC[-1]^3+20*RC[-1]^2+7*RC[-1]-13"
    Range("B1").Select
    Selection.AutoFill Destination:=Range("B1:B10"), Type:=xlFillValues
    Range("B1:B10").Select
    ActiveSheet.Shapes.AddChart.Select
    ActiveChart.SetSourceData Source:=Range("=Лист1!$B$1:$B$10")
    ActiveChart.ChartType = xlXYScatterLines
End Sub
    
```

Рис. 4 - Измененная автоматически написанная программа макрорекордером по построению графика функции $y(x) = 48 \cdot x^3 + 20 \cdot x^2 + 7 \cdot x - 13$

Далее в книге 1 удалим всю предыдущую информацию выделением этой информации и нажатием клавиши Delete. Потом воспользуемся окном меню Excel в открытой книге 1 - Сервис | Макрос | Макросы | Выполнить и выбираем макрос Мас, который уже содержит изменения в программе в коэффициентах функции, в книге 1 автоматически построится новый график функции с измененными коэффициентами:

$$y(x) = 48 \cdot x^3 + 20 \cdot x^2 + 7 \cdot x - 13$$

(рис. 5).

Обратите внимание, что рассмотренный

нами пример всего лишь иллюстрация. Следовательно, овладев VBA для Excel, вы сможете сразу перейти к созданию макросов для других программных продуктов Microsoft. Более того, вы сможете создавать полноценные программные продукты, одновременно использующие функции самых разных приложений [5].

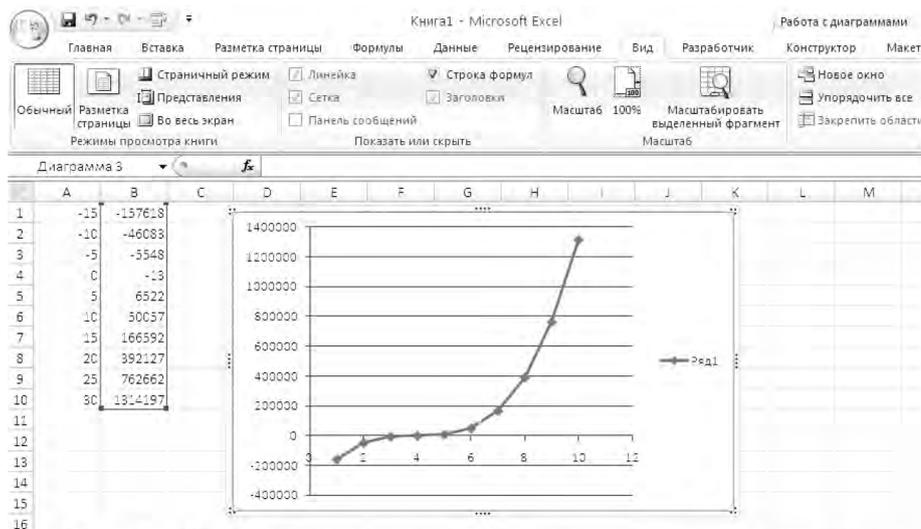


Рис. 5 - Численные значения аргумента x и функции $y(x) = 48 \cdot x^3 + 20 \cdot x^2 + 7 \cdot x - 13$ на заданном диапазоне и построенный точечный график, выполненный измененным макросом

Выводы. Таким образом, использование автоматически созданных маркорекордером программ в VBA позволит уменьшить затраты времени студентов, ученых, инженеров на выполнение многократных, типично повторяющихся действий, например, построение графиков различных функций для последующего их анализа, обобщения информации и т.п. А также, возможности VBA упрощают написание и редактирование программ начинающим пользователям.

Список литературы: 1. Гарнаев, А. Ю. Excel, VBA, Internet в экономике и финансах / А. Ю. Гарнаев. – СПб.: БХВ-Петербург, 2001. – 816 с. 2. Згуровський, М. З. Вступ до комп'ютерних інформаційних технологій: навч. посіб. / М. З. Згуровський, І. І. Коваленко, В. М. Михайленко. – К.: Вид-во Європ. ун-ту (фінанси, інформ. системи, менеджм. і бізнес), 2000. – 265 с. 3. Геворкян, Г. Х. Бейсик – это просто / Г. Х. Геворкян, В. Н. Семенов. – М.: Радио и связь, 1989. – 144 с. 4. Якушева, Н. М. Visual Basic для студентов / Н. М. Якушева. – М.: Радио и связь, 2001. – 232 с. 5. Слетцова, Л. Д. Программирование на VBA. Самоучитель / Л. Д. Слетцова. – М.: Издательский дом "Вильямс", 2004. – 384 с. 6. Назаров, С. В. Программирование в пакетах MS Office: навч. посіб. / С. В. Назаров, П. П. Мельников, Л. П. Смольников и др. – М.: Финансы и статистика, 2007. – 656 с. 7. Хорев, В. Д. Самоучитель программирования на VBA в Microsoft Office / В. Д. Хорев. – К.: Юниор, 2001. – 320 с. 8. Кузьменко В. Г. Программирование на VBA 2002 / В. Г. Кузьменко. – М.: ООО "Бином-Пресс", 2003 г. – 880 с. 9. Камминг, Стив. VBA для "чайников" / Стив Камминг; пер. с англ. – [3-е изд.]. – М.: Издательский дом "Вильямс", 2001 – 448 с. 10. Глушаков, С. В. Microsoft Excel 2007. Краткий курс / С. В. Глушаков, А. С. Сурядный. – М.: Харвест, 2008. – 352 с.

Bibliography (transliterated): 1. *Garnaev, A. Y.* (2001). Excel, VBA, Internet in economics and finance, 816. 2. *Zgurovsky, M. Z.* (2000). Introduction to computer information technology: Navch.posib. / M. Z. Zgurovsky, I. I. Kovalenko, V.M. Mihaylenko, 265. 3. *Gevorkian, G. H.* (1989). Basic is easy, 144. 4. *Yakushev, N. M.* (2001). Visual Basic for students, 232. 5. *Sleptsova, L. D.* (2004). VBA Programming. The tutorial, 384. 6. *Nazarov, C. V.* (2007). Programming in MS Office packages: Navch.posib. / C. V. Nazarov, P. P. Mel'nikov, L. P. Smolnikov, etc., 656. 7. *Horev, C. D.* (2001). Manual of VBA programming in Microsoft Office, 320. 8. *Kuzmenko, C.* (2003) VBA Programming 2002, 880. 9. *Cumming, Steve.* (2001). VBA for dummies.: TRANS. from English, 448. 10. *Glushakov, S. C.* (2008). Microsoft Excel 2007. Short course, 352.

Надійшла (received) 10.12.2014

УДК 621.182.2.001.57

Є. Є. ЧАЙКОВСЬКА, канд. техн. наук, с. н. с., доц., ОНПУ, Одеса;
Б. І. МОЛОДКОВЕЦЬ, магістр, ОНПУ, Одеса;

ПІДТРИМКА ФУНКЦІОНУВАННЯ БІОГАЗОВОЇ УСТАНОВКИ У СКЛАДІ КОГЕНЕРАЦІЙНОЇ СИСТЕМИ

Розроблено метод підтримки процесу зброджування, відвантаження збродженої сировини та завантаження свіжого матеріалу на рівні прийняття рішень. Прогнозування зміни температури зброджування дозволяє використовувати зброжене сушло в якості низькопотенційного джерела енергії для теплового насоса щодо встановлення температури теплоносія, що гріє, на вході в теплообмінник, вбудований в метантенк, при вимірюванні температури теплоносія, що гріє, на виході з теплообмінника.

Ключові слова: когенерація, електрична енергія, теплота, біогаз, зброджування, метантенк, тепловий насос, прийняття рішень

Вступ. Однією з переваг когенераційних технологій в умовах ресурса та енергозбереження є можливість використання біогазу як альтернативного джерела енергії [1]. Когенераційні технології, що мають у своєму складі первинний двигун, електрогенератор, систему утилізації теплоти, систему контролю й управління дозволяють включати в технологічні схеми різні енергетичні установки: поршневий двигун, газову турбін, парову турбін або комбінацію парової та газової турбін. Теплові насоси, що використовують у складі когенераційних систем, не мають зв'язку з виробництвом біогазу, не використовують зброжений матеріал у якості низькопотенційного джерела енергії, а виконують підтримку співвідношення виробництва електричної енергії та теплоти з використанням інших джерел енергії [2 – 4]. При виробництві ж біогазу зброжене сушло має енергетичний потенціал щодо можливості використання у якості низькопотенційного джерела енергії для теплового насоса, а на підтримку процесу зброджування витрачають до 20% –30% виробленої енергії.

Умови когенерації при використанні біогазу ускладнені непостійним виходом біогазу, що вимагає додаткових баків для зброженої сировини, додаткових ємностей газгольдерів, значних витрат теплоти на підтримку процесу зброджування за рахунок виробленої енергії [5 – 7]. При безперервному

© Є. Є. ЧАЙКОВСЬКА, Б. І. МОЛОДКОВЕЦЬ, 2014