

УДК 004.4:621.376.54:621.316.722

В. П. Беляев, А. А. Лабушев

Белорусский государственный технологический университет

**ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ
РЕГУЛЯТОРА НАПРЯЖЕНИЯ С ШИМ-УПРАВЛЕНИЕМ**

Статья посвящена разработке программного обеспечения функционирования полупроводникового регулятора напряжения, реализующего ШИМ-управление напряжением переменного тока на статорных обмотках асинхронного короткозамкнутого электродвигателя. Широтно-импульсное регулирование напряжением осуществляется запатентованным способом. Способ позволяет уничтожить некоторые высшие гармонические составляющие в питающем двигателе напряжении, что создает условия получения качественных пусковых, регулировочных и тормозных свойств регулируемого электропривода переменного тока, в основу которого положен регулятор напряжения. Высокие эксплуатационные характеристики полупроводниковых регуляторов достигаются в том числе использованием интеллектуальных силовых модулей и применением управления ими микроконтроллерами или микропроцессорами, на которые возлагаются функции регулирования, контроля, технологического управления всем электрооборудованием. Обозначена задача разработки программного обеспечения для них. Основу этого обеспечения составляет программа изменения угла управления ШИМ-напряжением и создания соответствующей последовательности включения или отключения силовых полупроводниковых ключей регулятора напряжения. Для разработки программы выбран объектно-ориентированный язык программирования *Java* с интерфейсами, работающий на платформе *Windows*. Приведен листинг программы, проведены наладочно-проверочные расчеты, показаны их формализованные результаты, подтверждающие адекватность разработанной программы, выполняющей генерацию логических команд формирования управления ШИМ-напряжением в соответствии с заданным технологическим циклом функционирования регулятора напряжения.

Ключевые слова: электропривод переменного тока, широтно-импульсная модуляция напряжения, программирование.

V. P. Belyayev, A. A. Labushev

Belarusian State Technological University

SOFTWARE VOLTAGE REGULATOR WITH PWM CONTROL

The article is devoted to the development of software for the functioning of a semiconductor voltage regulator that implements PWM control of the AC voltage on the stationary windings of an asynchronous short-circuited electric motor. Pulse width regulation voltage is carried out in a patented way. The method allows you to destroy some of the higher harmonic components in the motor supply voltage, which creates the conditions for obtaining high-quality starting, adjusting and braking properties of an adjustable AC drive, which is based on a voltage regulator. High performance characteristics of semiconductor controllers are achieved, including the Use of intelligent power modules and the Use of their control by microcontrollers or microprocessors, which are assigned the functions of regulation, control, and technological control of all electrical equipment. Denotes the task of developing software for them. The basis of this provision is the program for changing the angle of control of the PWM voltage and creating the corresponding sequence of switching on or off the power semiconductor switches of the voltage regulator. To develop a program, an object-oriented programming language *Java* with an interface has been chosen. running on the *Windows* platform. The program listing is given. Listed program listing. Conducted adjustment and verification calculations of the program. The formalized results of the program are shown, confirming the adequacy of the developed program that performs the generation of logical commands for generating the PWM voltage control in accordance with the specified technological cycle of the voltage regulator.

Key words: AC electric drive, pulse width voltage modulation, programming.

Введение. В силу выгодных технико-экономических причин современные электроприводы в большинстве своем создаются на базе асинхронного короткозамкнутого электродвигателя, который проще по конструкции, дешевле, имеет меньшие капитальные и эксплуатационные затраты. Успехи технологии полу-

проводниковой силовой техники позволили создать модульные полупроводниковые преобразователи на значительные мощности, реализующие разнообразные теоретические разработки регулируемого электропривода переменного тока. Основным способом получения качественных пусковых, регулировочных и

тормозных свойств регулируемого электропривода переменного тока, сопоставимых с аналогичными свойствами регулируемого электропривода постоянного тока, является частотный способ управления. Однако он реализуется достаточно сложными системами управления с использованием интеллектуальных полупроводниковых преобразователей. Параметрический способ управления асинхронным электроприводом имеет определенные ограничения по диапазону получения регулировочных характеристик, расширение которого приводит к увеличению габаритной мощности электродвигателя и, как правило, к недоиспользованию этой мощности. Но в вопросах организации процессов пуска он является альтернативным частотному управлению, что подтверждается промышленным выпуском регуляторов напряжения (устройств плавного пуска) [1]. В построении регулируемого электропривода переменного тока имеется особенность, состоящая в том, что его управление выполняется по одному каналу – по обмоткам статора двигателя, а также существует сложность в измерении ряда координат электропривода.

Основная часть. Одним из возможных способов регулирования координат асинхронного двигателя является изменение напряжения на обмотках статора (параметрическое управление). Для реализации этого способа регулирования между питающей сетью и обмотками статора включается, в частности, полупроводниковый регулятор напряжения, который осуществляет фазовый способ управления $U(t) = U_{\max} \sin \alpha$. Механическая характеристика $M = f(\delta)$ асинхронного двигателя на основе теории обобщенной машины описывается следующим выражением, из которого видна квадратичная зависимость момента двигателя от напряжения питания:

$$M_3 = \frac{3}{2} \cdot \frac{|\bar{u}_s|^2 a_{12} a_{21} \delta}{R_s \left[(a_{11}^2 + \omega_s^2) \cdot (\delta_k^2 + \delta^2) + 2a_{12} a_{21} \omega_s \delta \right]},$$

где $|\bar{u}_s|$ – модуль вектора фазного напряжения, питающего обмотки статора; R_s – активное сопротивление обмотки фазы статора; ω_s – частота вращения магнитного поля статора; $\delta = \omega_r - \omega_s$ – параметр абсолютного скольжения; ω_r – частота вращения ротора; δ_k – критическое значение параметра абсолютного скольжения; a_{11} , a_{12} , a_{21} – коэффициенты, составленные из параметров эквивалентной схемы электродвигателя переменного тока [2].

Использование классического параметрического регулирования приводит к тому, что в результате преобразования входного в регулятор напряжения на обмотки статора подается

выходное несинусоидальное, а иногда и несимметричное напряжение, качественный гармонический состав которого ухудшается с увеличением угла управления α . Такая ситуация ухудшает качество электромагнитного момента и скорости в динамических процессах [3]. Исправляется это применением широтно-импульсной модуляции напряжения, в задачу которой входит уменьшение высших гармонических составляющих в питающем обмотки двигателя напряжении. Вариант, позволяющий получить качественную ШИМ-напряжения, предложен в [4], основа которого изображена на рис. 1.

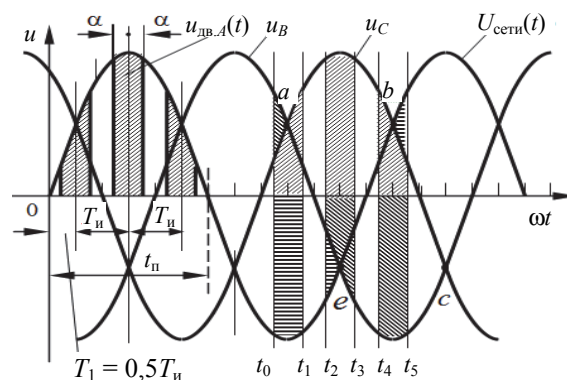


Рис. 1. Эпюры формирования ШИМ-напряжения

На рис. 1 обозначены U_A , U_B , U_C – система трехфазного напряжения питающей сети; a , b , c , e – точки естественного пересечения полупериодов напряжений; $U_{дв.А}(t)$ – импульс ШИМ-напряжения фазы A , подающийся на обмотку статора двигателя; $U_{сети}(t)$ – напряжение питающей сети; α – угол регулирования; T_1 – расстояние до центра первого пультса; $T_и$ – расстояние между пультсами; $t_п$ – время полупериода напряжения питающей сети.

Проведенный математический анализ подтвердил подавление определенных гармоник, чем улучшил гармонический состав питающего напряжения и энергетические показатели, получение линейной выходной характеристики по первой энергетической составляющей напряжения $U_{вых1} = f(\alpha)$, а также позволил сформировать симметричную трехфазную систему как для фазных, так и для линейных напряжений по сравнению с другими способами [5]. Укажем, что угол регулирования α может изменяться по любому закону: линейному, пропорциональному, прямоугольному, синусоидальному и т. п.

Высокие эксплуатационные характеристики полупроводниковых регуляторов достигаются в том числе использованием интеллектуальных силовых модулей и применением управления ими микроконтроллерами или микропроцессорами

ми, на которые возлагаются функции регулирования, контроля, технологического управления всем электрооборудованием. Таким образом, возникает задача разработки программного обеспечения для них. Естественно, основу этого обеспечения составляет программа изменения угла управления ШИМ-напряжением.

Технология программирования. Для наглядности и облегчения понимания сути изменения угла управления ШИМ-напряжения была составлена графическая интерпретации этого процесса

(рис. 2). Начальными положениями для разработки программы определены: численное задание идеи способа с последующей технологией изменения угла управления напряжением; формирование условий синхронизации программы с состоянием трехфазного напряжения питания на входе регулятора напряжения, определяющим порядок включения или отключения его транзисторов; входные логические сигналы технологии управления электроприводом (пуск, торможение, аварийное состояние и т. п.).

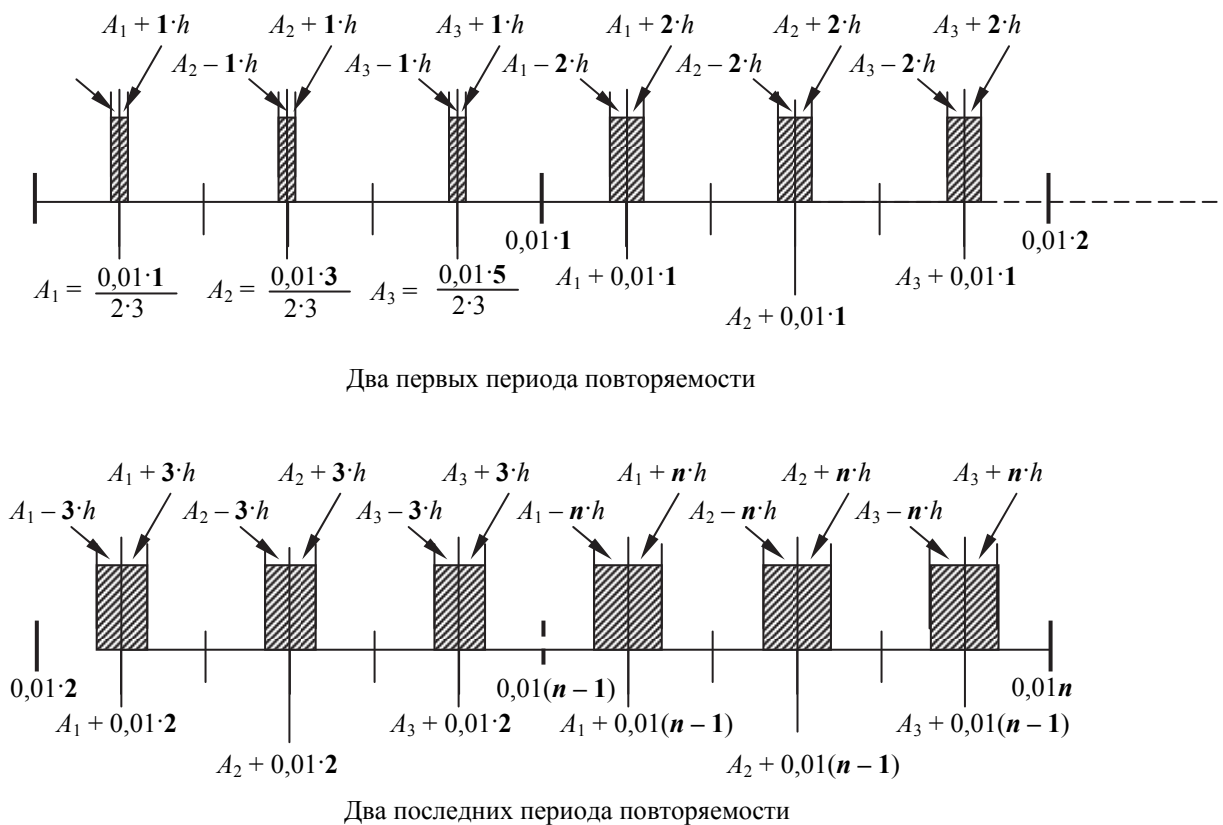


Рис. 2. Графическая интерпретация алгоритма программы

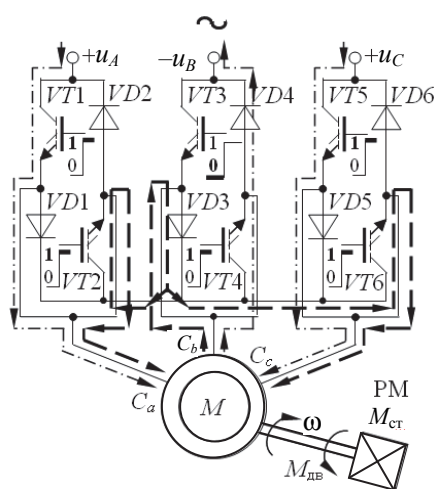


Рис. 3. Схема регулятора напряжения при U_A и $U_C > 0$, $U_B < 0$

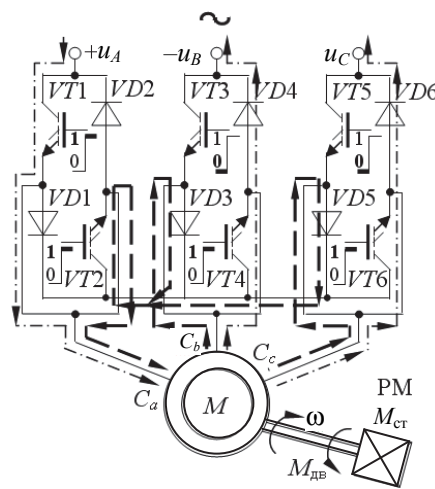


Рис. 4. Схема регулятора напряжения при $U_A > 0$, U_B и $U_C < 0$

Состояние трехфазного напряжения на входе регулятора напряжения (рис. 1) имеет шесть вариантов сочетаний положительных и отрицательных его значений на местах центров формирующихся импульсов напряжения. Однако принципиальны два варианта: U_A и $U_C > 0$, а $U_B < 0$ (рис. 3) или $U_A > 0$, а U_B и $U_C < 0$ (рис. 4). Другие аналогично повторяемые. Анализ работы силовых транзисторов на приведенных схемах дает ответ, на какие из них программа должна подавать логическую единицу для включения или отключения для формирования импульса напряжения на обмотках двигателя, а на какие – для формирования пауз, что является *основным конечным результатом работы программы*. Это входной параметр программы, определяемый технологическим циклом работы оборудования, например $tp = 2$ с. Поскольку на каждом полупериоде напряжения переменного тока, который назовем периодом повторяемости, может располагаться в принципе $m = 3, 6, 9 \dots$ импульсов, регулируемых по ширине, то в программе должна быть заложена эта возможность. Длительность периода повторяемости для напряжения с частотой $f = 50$ Гц равна $tr = 1 / f / 2 = 0,01$ с.

Составление самой программы начато с разработки блок-схемы ее алгоритма. Продемонстрируем разработку программы для $m = 3, 6$, что обоснуется впоследствии. С учетом данного основного условия сформируем остальные.

Максимальный угол управления составит $\alpha_{\max} = tr / 2m = 0,001667$ с. Время увеличения напряжения от минимального до максимального значения обозначим tp – время пуска.

Таким же входным параметром программы является время уменьшения напряжения, что обозначится как время процесса торможения $tt = 2$ с. Центры расположения импульсов в общем виде при пуске определяются так:

$$A_{k,q} = \left[\left(j \frac{tr}{2m} \right)_k + tr(Zp_q) \right]_q,$$

где j – показатель, определяющий место центров импульсов на первом периоде повторяемости; $j = 1, 3, 5$; q – номер периода повторяемости (каждый полупериод питающего напряжения), на котором выполняется увеличение угла управления напряжением (ширины импульса), $k = 1, 2, 3$, $q = 0, 1, 2, 3, 4, 5, 6, 7 \dots$ $Zp = tp / tr$.

Центры расположения импульсов в общем виде при торможении определяются так:

$$A_{k,q} = \left[\left(j \frac{tr}{2m} \right)_k - tr(Zp_q) \right]_q,$$

где j – показатель, определяющий место центров импульсов на первом периоде повторяемости при торможении; $j = 1, 3, 5$; q – номер периода повторяемости (каждый полупериод питающего напряжения), на котором выполняется уменьшение угла управления напряжением (ширины импульса), $k = 1, 2, 3$; $q = 0, 1, 2, 3, 4, 5, 6, 7 \dots$ $Zt = tt / tr$.

Для $m = 3$ имеем $k = 1, 2, 3$ и координаты центров расположения импульсов на всех полупериодах вычисляются следующим образом:

$$A_{1,q} = 1A_{1,q} = \frac{1 \cdot 0,01}{2 \cdot 3} = 0,001667 \text{ с};$$

$$A_{1,q} = 3A_{1,q} = \frac{3 \cdot 0,01}{2 \cdot 3} = 0,005 \text{ с};$$

$$A_{1,q} = 5A_{1,q} = \frac{5 \cdot 0,01}{2 \cdot 3} = 0,008333 \text{ с}.$$

Поскольку технологический цикл работы любого оборудования состоит из приведения его в движение, работы на определенных заданных уровнях скорости, остановов оборудования, то и напряжение на электромеханическом преобразователе-двигателе, приводящем его в движение, должно увеличиваться, сохранять постоянное значение, а затем уменьшаться.

Поэтому разделим программу на три интервала: время пуска, работы и торможения, на которых для определенного упрощения сохраним принятый шаг программирования [6]. Это условие определило задание времени работы программы и количество шагов программирования, например, $tu = 5$ с. Таким образом, количество периодов повторяемости при расчетах пуска или торможения в программе составит $Zp = Zt = tp / tr = tt / tr = 2 / 0,01 = 200$. Для рассматриваемого варианта программы определим пропорциональный закон изменения угла от периода к периоду повторяемости. Следовательно, на каждом полупериоде исходного напряжения ширина импульсов одинаковая, соответствующая изменяющемуся углу управления. Изменение ширины импульсов от периода повторяемости к периоду (шаг программирования времени изменения ширины импульсов) вытекает из следующих рассуждений, состоящих в том, что время пуска задано в размере, например $tp = 2$ с, и за это время рассчитывается $Zp = 200$ периодов, а ширина импульсов достигает максимального значения $\alpha_{\max} = 0,001667$ с, следовательно, шаг программирования составит

$$hp = \alpha_{\max} / Zp = 0,001667 / 200 = 8,33 \cdot 10^{-6} \text{ с}.$$

Этот параметр, в принципе, определяет дискрету точности изменения напряжения в динамических процессах. Анализ шага про-

граммирования указывает на очень маленькое его значение, что приводит к загрузке программы многочисленными промежуточными вычислениями, занимающими большой объем памяти. Для увеличения шага можно принять, что изменение ширины импульсов будет выполняться на двух периодах повторяемости (на одном периоде входного в регулятор напряжения). В данном случае при прочих равных условиях шаг будет равен $hp = \alpha_{\max} / Zp / 2 = 0,001667 / 100 = 1,67 \cdot 10^{-5}$ с. В силу такой ситуации с получением малого значения шага еще практически имеет смысл расположение на полупериоде шести импульсов, что и сделано в предлагаемой программе.

Оценка динамики увеличения действующего напряжения на выходе регулятора напряжения при его регулировании подсказала бы оптимальное значение шага изменения угла.

Для формирования силового импульса напряжения на любой обмотке асинхронного двигателя включением или отключением транзисторов силовой схемы регулятора напряжения необходимо сформировать логическую единицу в момент t_0 для включения транзистора и логический ноль в момент t_1 для его отключения. При программировании это будет выглядеть так. На каждом шаге обозначается значение некоторой выходной функция F следующим алгоритмом:

если $t_n h \geq 0,01n$, где t_n – время перехода к следующему полупериоду напряжения, $t_n = t_{n-1} + 1$, а $t_1 = 1$, $n = 2, 3, 4, \dots$ tp / Zp , то $A1 = A1 + 0,01n$;

если $t \leq A_k - t_n h$, то $F = 0$ и, если $t \geq A_k - t_n h$, то $F = 1$;

если $t \leq A_k + t_n h$, то $F = 1$ и, если $t \geq A_k + t_n h$, то $F = 0$.

В программе заложены некоторые технологические команды, связанные с ее прекращением по внешним воздействиям, заранее задающимися или приходящими в течение выполнения технологического цикла. Примером может служить следующая ситуация. По мере увеличения напряжения питания асинхронного двигателя ускорение ротора постепенно увеличивается и достигает установившегося значения после того, как исчезнут колебания электромагнитного момента. При выходе напряжения питания на номинальное значение скорость ротора приобретает установившееся значение. Анализ изменения электромагнитного момента за весь процесс разгона показывает, что он достигает своего максимального значения в момент времени t_s в зоне критического скольжения $s_{кр}$, когда двигатель переходит от «жесткого» режима работы к «мягкому». Несмотря на увеличивающееся напряжение питания двигателя,

значение электромагнитного момента уменьшается в соответствии с формой его механической характеристики. Скорость продолжает возрастать и достигает того значения, которое достаточно близко соответствует установившемуся режиму работы двигателя с M_{xx} , причем к этому времени значение электромагнитного момента становится равным M_{xx} . С этого момента времени увеличение напряжения питания не оказывает существенного влияния на изменение электромагнитного момента и скорости двигателя. При разном времени управляемого пуска $t_{упр.п}$ (различной интенсивности увеличения напряжения питания) эта зона составляет 10–40% $t_{упр.п}$. Чем больше время управляемого пуска, тем больше эта зона. Поэтому рационально не продолжать увеличение угла управления ШИМ-напряжением, а выполнить скачкообразный переход на значение угла, равного α_{\max} , т. е., что равносильно подаче напряжения питающей сети непосредственно на обмотки статора двигателя. Это достигается выводом регулятора напряжения из работы путем его шунтирования [7].

Для разработки программы был выбран объектно-ориентированный язык программирования *Java* с интерфейсами [8]. Листинг программы, составленный по оговоренным условиям, приведен на рис. 5.

Листинг программы

```
public class Main {
public static void main(String[] args) {
double A1=0, A2=0, A3=0, A4=0, A5=0, A6=0,
N1=0, N2=0, N0=0; //задание значения, чтобы
потом в циклах их можно было изменить
double f=50; //задание значения частоты
double P=1/f; //задание периода напряжения
(0.02)
double T=P/2; //задание полупериода напряже-
ния(0.01)
double m=3; //задание количества импульсов
//задание значения скачков по времени
double A11=1*T/(2*m); //0.000833
double A22=3*T/(2*m); //0.0025
double A33=5*T/(2*m); //0.00416
double A44=7*T/(2*m); //0.00583
double A55=9*T/(2*m); //0.0075
double A66=11*T/(2*m); //0.00916
double N00=0;
double N11=T-2*(T/3); //0.003333
double N22=T-(T/3); //0.006666
//задание переменных VT1-VT6 для того, чтобы
с каждым новым шагом программирования
включались или отключались необходимые
транзисторы
int vt1=0; int vt2=0; int vt3=0;
```

```

int vt4=0; int vt5=0; int vt6=0;
//задание напряжений Ua, Ub, Uc и начальное
их положение
int Ua=-1; int Ub=-1; int Uc=1;
double t_puska=0.1; //задание времени пуска
double t_RP=1; //задание времени работы программы
double t_zatuh=2; //задание времени затухания
торможения
double t_sum=t_puska+t_RP+t_zatuh; // расчет
суммарного времени работы программы
double procent=1; //задание процента ширины
импульса, от которого идет скачок
double KP=t_puska/T; //расчет количества по-
вторений(200)
double h=A11/KP; //нахождение шага по времени()
//затем задаются параметры для их дальней-
шего изменения
//если инициализировать переменную в момент
изменения – программа работает неправильно
double n=1; double n1=1;
// переменные для счета периодов повторяемости
и моментов, когда напряжение меняет свой знак
double i=0; double ia=0; double ib=0;
double ic=0; int F=0;
boolean S=true; //задание параметра работы
программы, если true – работает, если
S=false – программа не работает
boolean Q=true; //задание параметра аварийного
режима, если true – аварийный режим, остано-
вка программы, если false – никаких действий
if (S==true) { //проверка значения S, в данном
случае на значение true
//1 цикл – создание первого цикла для времени пуска
for (double t=0; t<=t_puska; t=t+h) {
if (t>=(i*0.01)) { //проверка по времени измене-
ния положения точек импульса
A1=A11+(i*0.01); A2=A22+(i*0.01);
A3=A33+(i*0.01); A4=A44+(i*0.01);
A5=A55+(i*0.01); A6=A66+(i*0.01);
n1=i+1; //счет ведется до конца(200), нужно
для 3 цикла(t_zatuh)
if (t<t_puska*procent) { //проверка по заданному
проценту для скачка
n=i+1; //счет ведется до определенного значе-
ния, чтобы в нужный момент шли одинаковые
импульсы}}
//проверка по времени положения точки изме-
нения знака напряжения
if (t>=N11+(ic*0.01)){ N1=N11+(ic*0.01);}
if (t>=N00+(ia*0.01)){ N0=N00+(ia*0.01);}
if (t>=N22+(ib*0.01)){ N2=N22+(ib*0.01);}
//проверка и изменение знака напряжения Ua
if (N0==N00+(ia*0.01)){ia=ia+1;
if (Ua===-1) {Ua=1;} else if (Ua==1){Ua=-1;}
// проверка и изменение знака напряжения Ub
if (N2==N22+(ib*0.01)){ib=ib+1;
if (Ub==1) {Ub=-1;} else if (Ub===-1){Ub=1;}}
// проверка и изменение знака напряжения Uc

```

```

if (N1==N11+(ic*0.01)){ic=ic+1;
if (Uc===-1){Uc=1;} else if (Uc==1){Uc=-1;}}
if (A1==A11+(i*0.01)) { //проверка на равенство
для изменения числа i, необходимое для опреде-
ления новых точек центра импульса
i=i+1;}
//Проверка по времени процесса пуска на 3 им-
пульсах A1, A2, A3 для вывода значений F
//проверка: какие напряжения положительны на
одном из периодов повторяемости, и в зависимости
от этого включаются необходимые транзисторы
if (t<A1-(h*n)) {
if (Uc==1 && Ua==1) {
vt1=0; vt2=0; vt3=0; vt4=0; vt5=0; vt6=1;
} else if (Ub==1){
vt1=0; vt2=1; vt3=0; vt4=1; vt5=0; vt6=0;}
F=0;
} else if (t>=A1+(h*n)){
if (Ua==1) {
vt1=0; vt2=1; vt3=0; vt4=0; vt5=0; vt6=1;
} else if (Ub==1){
vt1=0; vt2=0; vt3=0; vt4=1; vt5=0; vt6=0;}
F=0;
} else if (t>=A1-(h*n)) {
if (Ua==1 && Uc==1) {
vt1=1; vt2=0; vt3=0; vt4=0; vt5=1; vt6=0;
} else if (Ub==1){
vt1=0; vt2=0; vt3=1; vt4=0; vt5=0; vt6=0;}
F=1;}
if (t>=A2+(h*n)) {
if (Ua==1) {
vt1=0; vt2=1; vt3=0; vt4=0; vt5=0; vt6=0;
} else if (Uc==1){
vt1=0; vt2=0; vt3=0; vt4=1; vt5=0; vt6=1;}
F=0;
} else if (t>=A2 - (h*n)) {
if (Ua==1) {
vt1=1; vt2=0; vt3=0; vt4=0; vt5=0; vt6=0;
} else if (Ub==1 && Uc==1){
vt1=0; vt2=0; vt3=1; vt4=0; vt5=1; vt6=0;}
F=1;}
if (t>=A3+(h*n)) {
if (Ub==1 && Ua==1) {
vt1=0; vt2=1; vt3=0; vt4=1; vt5=0; vt6=0;
} else if (Uc==1){
vt1=0; vt2=0; vt3=0; vt4=0; vt5=0; vt6=1;}
F=0;
} else if (t>=A3 - (h*n)) {
if (Ua==1 && Ub==1) {
vt1=1; vt2=0; vt3=1; vt4=0; vt5=0; vt6=0;
} else if (Uc==1){
vt1=0; vt2=0; vt3=0; vt4=0; vt5=1; vt6=0;}
F=1;}
if (t>=A4+(h*n)) {F=0;
} else if (t>=A4 - (h*n)) {F=1;}
if (t>=A5+(h*n)) {F=0;
} else if (t>=A5 - (h*n)) {F=1;}
if (t>=A6+(h*n)) {F=0;

```

```

} else if (t >= A6 - (h*n)) {F=1;}
System.out.print(vt6);
// System.out.print(F); // вывод значений F
//2 цикл – создание второго цикла для времени
работы программы
for (double t1=t_puska; t1<=t_puska+t_RP;
t1=t1+h) {F=1;
// System.out.print(F); // вывод значений F
// дальнейшие расчеты необходимы для рабо-
ты третьего цикла
i=0;
A11=A11+t_puska+t_RP; //3.000833
A22=A22+t_puska+t_RP; //3.0025
A33=A33+t_puska+t_RP; //3.00416
A44=A44+t_puska+t_RP; //3.00583
A55=A55+t_puska+t_RP; //3.0075
A66=A66+t_puska+t_RP; //3.00916
//3 цикл – создание третьего цикла для времени
затухания
for (double t2=t_puska+t_RP; t2<=t_sum;
t2=t2+h) {
if (t2 >= ((t_puska+t_RP)+(i*0.01))) { // проверка по
времени изменения положения центра импульса
A1=A11+(i*0.01); A2=A22+(i*0.01);
A3=A33+(i*0.01); A4=A44+(i*0.01);
A5=A55+(i*0.01); A6=A66+(i*0.01);
n=n1-i;
if (A1==A11+(i*0.01)) { // проверка на равенство
для изменения числа i, необходимого для опре-
деления новых точек центра импульса
i=i+1;
}
//Проверка по времени процесса затухания на 3

```

```

импульсах A1, A2, A3 для вывода значений F
if (t2 < A1 - (h*n)) {F=0;
} else if (t2 >= A1 + (h*n)) {F=0;
} else if (t2 >= A1 - (h*n)) {F=1;
} if (t2 >= A2 + (h*n)) {F=0;
} else if (t2 >= A2 - (h*n)) {F=1;
} if (t2 >= A3 + (h*n)) {F=0;
} else if (t2 >= A3 - (h*n)) {F=1;
} if (t2 >= A4 + (h*n)) {F=0;
} else if (t2 >= A4 - (h*n)) {F=1;
} if (t2 >= A5 + (h*n)) {F=0;
} else if (t2 >= A5 - (h*n)) {F=1;
} if (t2 >= A6 + (h*n)) {F=0;
} else if (t2 >= A6 - (h*n)) {F=1;
}
// System.out.print(F); // вывод значений F
} else if (S==false) { // проверка значения S, в дан-
ном случае по значению false
System.exit(0); // не запускает программу } }

```

Рис. 5. Листинг программы

Программа была отлажена и проверена рас-
 четами при принятых выше условиях. Для на-
 глядности и контроля результатов они выведе-
 ны на консоль. Фрагмент (скриншот) резуль-
 татов работы программы по формированию логи-
 ческих единиц для управления транзисторами
 VT1–VT6 регулятора, на котором центры им-
 пульсов обозначены черточками, а начало и
 конец периода повторяемости обозначены
 жирными символами (цифры 0 или 1) приведен
 на рис. 6.

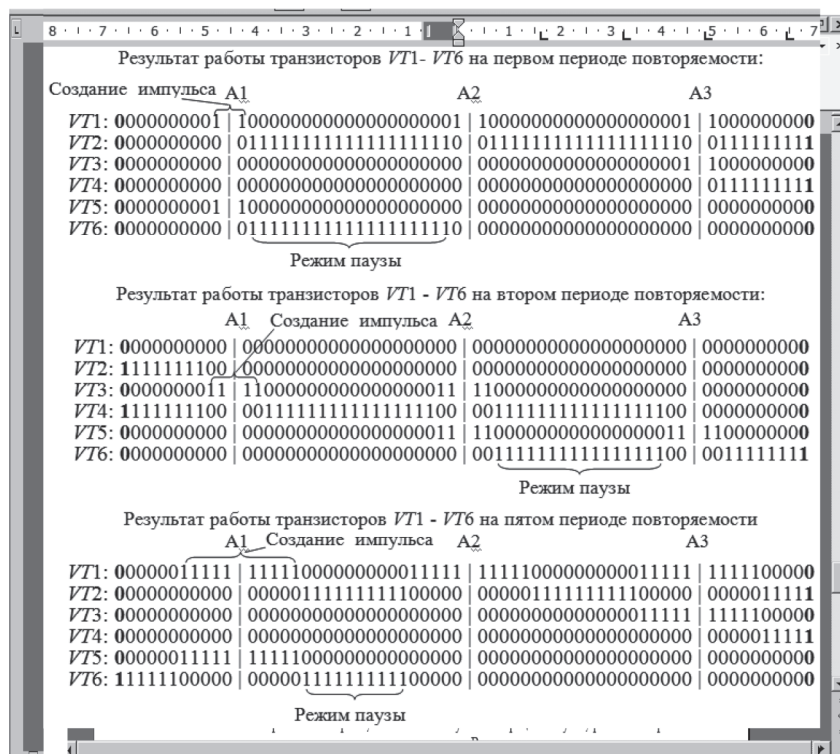


Рис. 6. Скриншот результатов работы программы

На скриншоте обозначены моменты создания ширины импульсов – это совокупность последовательно стоящих логических единиц, выставленных программой на консоль для включения силовых транзисторов регулятора, номера которых указаны слева в соответствии с рис. 3 и 4. Аналогичным приемом обозначен режим формирования паузы, также в соответствии с рассмотренным включением транзисторов. Количество единиц характеризует последовательное увеличение ширины импульса и уменьшения продолжительности паузы.

Заключение. Разработан алгоритм, по которому в процессе пуска с каждым новым периодом повторяемости ширина его трех импульсов равномерно увеличивается, а паузы постепенно уменьшаются до тех пор, пока угол управления

не станет максимальным. Для четных и нечетных периодов повторяемости включение силовых транзисторов выполняется по определенному алгоритму в зависимости от того, положительным или отрицательным будет питающее регулятор напряжение. Программа обслуживает процессы пуска, работы и торможения. В процессе торможения развитие ширины импульсов имеет инверсный характер.

Анализ результатов проведенных наладочно-проверочных расчетов позволяет сделать вывод об адекватности разработанной программы, выполняющей генерацию логических команд формирования управления ШИМ-управлением в соответствии с заданным технологическим циклом функционирования регулятора напряжения.

Литература

1. Katalog Simens [Электронный ресурс]. URL: <https://publikacije.siemens-info.com/pdf/217/Katalog%20SIRIUS%20IC10A%202018%20-%20softstartery%203RW.pdf> (дата обращения: 22.04.2018).
2. Беляев В. П., Давидович Л. М. Электромеханика. Электромеханическое преобразование при частотном управлении электрическими машинами. Минск: БГТУ, 2004. 82 с.
3. Браславский И. Я. Асинхронный полупроводниковый электропривод с параметрическим управлением. М.: Энергоатомиздат, 1988. 224 с.
4. Способ широтно-импульсной модуляции выходного напряжения переменного тока: а.с. 764096, СССР, МКИ³ Н 02 Р 13/18 / В. П. Беляев, Г. И. Гульков, В. Г. Сидоров; заявл. 10.11.78, опубл. 15.09.80 // Открытия. Изобретения. 1980. № 34. С. 2.
5. Беляев В. П. Электропривод переменного тока с ШИМ-управлением // Энергетика – Изв. высш. учеб. заведений и энерг. объединений СНГ. 2014. № 1. С. 11–20.
6. Беляев В. П. Электрооборудование полиграфических машин. Минск: БГТУ, 2012. 199 с.
7. Способ пуска короткозамкнутого асинхронного электродвигателя с параметрическим управлением в статоре: пат. 10546 Респ. Беларусь, МПК Н 02 К 16/17 / В. П. Беляев, заявитель Белорус. гос. технол. ун-т, № а 20060721; заявл. 13.07.2006; опубл. 2008.04.30 // Афіцыйны бюл. / Нац. цэнтр інтэлектуал. уласнасці. 2008. № 4. С.17.
8. Шилдт Герберт. Java 8: руководство для начинающих, 6-е изд.; пер. с англ. М.: ООО «И.Д. Вильямс», 2015. 720 с.

References

1. Katalog Simens [Electronic resource]. Available at: <https://publikacije.siemens-info.com/pdf/217/Katalog%20SIRIUS%20IC10A%202018%20-%20softstartery%203RW.pdf> (accessed 22.04.2018).
2. Belyayev V. P., Davidovich L. M. *Elektromekhanika. Elektromekhanicheskoye preobrazobaniye pri chastotnom upravlenii elektricheskimi mashinami* [Electromechanics. Electromechanical conversion in the frequency control of electric machines]. Minsk, BGTU Publ., 2004. 82 p.
3. Braslavskiy I. Ya. *Asinkhronnyy poluprovodnikovyy elektroprivod s parametricheskim upravleniyem* [Parametric Asynchronous Semiconductor Electric Drive]. Moscow, Energoatomizdat Publ., 1988. 224 p.
4. Belyayev V. P., Gul'kov G. I., Sidorov V. G. *Sposob shirotno-impul'snoy modulyatsii vykhodnogo napryazheniya peremennogo toka* [The method of pulse width modulation of the output voltage of an alternating current]. A.S. no. 764096, 1980. P. 2.
5. Belyayev V. P. PWM AC drive. *Energetika – Izv. vyssh. ucheb. zavedeniy i energ. ob'yedineniy SNG*. [Power engineering news of higher educational institutions and energy associations of the CIS], 2014, no. 1, pp. 11–20 (In Russian).
6. Belyayev V. P. *Elektrooborudovaniye poligraficheskikh mashin* [Electrical equipment for printing machines]. Minsk, BGTU Publ., 2012. 199 p.
7. Belyayev V. P. *Sposob puska korotkozamknutogo asinkhronnogo elektrodvigatelya s parametricheskim upravleniyem v statore* [Method for starting a short-circuited asynchronous motor with parametric control in the stator]. Patent Republic of Belarus, no. 10546. 2008.

8. Shildt Gerbert. *Java 8: rukovodstvo dlya nachinayushchikh* [Java 8: beginner's guide]. Moscow, ООО "I.D. Vil'yams" Publ., 2015. 720 p.

Информация об авторах

Беляев Валерий Павлович – кандидат технических наук, доцент, доцент кафедры полиграфического оборудования и систем обработки информации. Белорусский государственный технологический университет (220006, г. Минск, ул. Свердлова, 13а, Республика Беларусь). Email: beliaev@belstu.by

Лабушев Александр Александрович – студент кафедры полиграфического оборудования и систем обработки информации. Белорусский государственный технологический университет (220006, г. Минск, ул. Свердлова, 13а, Республика Беларусь). Email: 3snowik2903@gmail.com

Information about the authors

Belyayev Valeriy Pavlovich – PhD (Engineering), Assistant Professor, Associate Professor, the Department of Printing Equipment and Information Processing Systems. Belarusian State Technological University (13a, Sverdlova str., 220006, Minsk, Republic of Belarus). Email: beliaev@belstu.by

Labushev Aleksandr Aleksandrovich – student, the Department of Printing Equipment and Information Processing Systems. Belarusian State Technological University (13a, Sverdlova str., 220006, Minsk, Republic of Belarus). Email: 3snowik2903@gmail.com

Поступила 04.06.2019