

УДК 681.2-5

**РАЗРАБОТКА СИСТЕМЫ УПРАВЛЕНИЯ ШАГОВЫМ  
ЭЛЕКТРОПРИВОДОМ НА БАЗЕ ПЛАТФОРМ  
ARDUINO И MATLAB**

**В.А. Куевда, П.И. Розкаряка**

ГОУВПО «Донецкий национальный технический университет»

*Работа посвящена разработке алгоритма, обеспечивающего максимальное быстродействие, и реализации системы управления шаговым электроприводом с помощью платы Arduino Uno и пакета прикладных программ MATLAB – Simulink.*

Шаговый двигатель – бесколлекторный электродвигатель постоянного тока, преобразующий электрические импульсы в дискретные механические перемещения с возможностью его дальнейшей фиксации [1].

Особенностью шаговых двигателей является то, что без датчика обратной связи возможно осуществлять позиционирование по перемещению, так как при подаче на обмотку двигателя импульса напряжения поворот его ротора осуществляется на некоторый угол (шаг). Этот шаг определяется конструктивными особенностями двигателя и схемой управления. В настоящее время используются двигатели с шагом в  $3.6^\circ$ ,  $1.8^\circ$ ,  $0.9^\circ$ . Скорость вращения двигателя определяется частотой подачи импульсов напряжения.

Электроприводы на основе шаговых двигателей получили широкое распространение в таких сферах, как компьютерная техника, манипуляторы, робототехника, различные координатные станки. Это обусловлено очевидными преимуществами такого типа электропривода – возможностью реализовать точное позиционирование и регулирование скорости с большой точностью без датчика обратной связи.

Достоинства шагового двигателя [1]:

- угол поворота ротора определяется числом импульсов, которые поданы на драйвер, нет необходимости в обратной связи, а скорость пропорциональна частоте входных импульсов;
- точное позиционирование и повторяемость, ошибка не накапливается от шага к шагу;
- быстрый старт/реверс;
- возможность получения низких скоростей и, следовательно, отсутствует необходимость в применении редуктора;
- отсутствие коллектора, следовательно, высокая надёжность;

- в двигателях с постоянными магнитами имеется фиксирующий момент (когда статор обесточен), равный примерно 10% от момента удержания.

Недостатки [1]:

- номинальное потребление энергии даже без нагрузки;
- возможен пропуск шага, следовательно, потеря контроля положения;
- явление резонанса;
- затруднена работа на высоких скоростях, вследствие не мгновенного нарастания тока.

Главным недостатком шагового двигателя является отсутствие обратной связи, что приводит к ограничению по ускорению, поэтому разгон двигателя необходимо осуществлять плавно. Кроме ограничения ускорения также необходимо не забывать и об ограничении скорости вращения.

Оптимальной по быстродействию диаграммой разгона являются тахограммы в виде треугольника и трапеции. Рассмотрим эти диаграммы подробнее.

Диаграмма 1.

Допустим необходимо отработать заданное перемещение  $\varphi_0$  по треугольной диаграмме (рис.1) за определенное время  $t_0$  без ограничения по ускорению  $\varepsilon_{\max}$  и по скорости  $\omega_{\max}$ .

Для нахождения необходимых параметров воспользуемся выражением для перемещения:

$$\varphi_0 = \frac{\varepsilon_0 t_1^2}{2} + \omega_0 t_y + \frac{\varepsilon_0 t_2^2}{2}, \quad (1)$$

где  $\varphi_0$  – перемещение;

$\omega_0$  – максимальная скорость электропривода;

$\varepsilon_0$  – ускорение электропривода при разгоне и торможении;

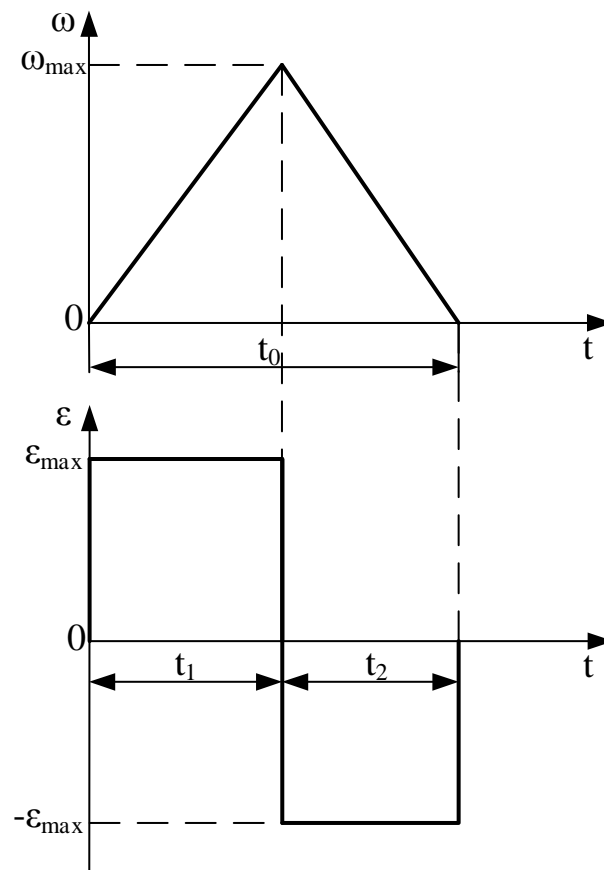
$t_1$  – продолжительность разгона двигателя до скорости  $\omega_0$ ;

$t_2$  – продолжительность торможения двигателя до нуля;

$t_y$  – продолжительность движения двигателя с установившейся скоростью.

Так как скорость должна изменяться по треугольной диаграмме, то движение с установившейся скоростью отсутствует ( $t_y = 0$ ). В таком случае уравнение (1) будет иметь вид:

$$\varphi_0 = \frac{\varepsilon_0 t_1^2}{2} + \frac{\varepsilon_0 t_2^2}{2}, \quad (2)$$



*Рис. 1 – Треугольная диаграмма*

Предположим, что продолжительность разгона и торможения равны между собой (ускорения при разгоне и торможении одинаковы). Тогда время находится по формуле:

$$t_1 = t_2 = \frac{t_0}{2}, \quad (3)$$

Подставим уравнение (3) в (2) и выразим максимальное ускорение:

$$\varepsilon_{\max} = \frac{4\varphi_0}{t_0^2}, \quad (4)$$

Максимальная скорость находится по формуле:

$$\omega_{\max} = \frac{\varepsilon_0 t_0}{2}, \quad (5)$$

Диаграмма 2.

Теперь допустим, что необходимо отработать заданное перемещение  $\varphi_0$  по треугольной диаграмме (рис.1) с ограничением по ускорению  $\varepsilon_0$  и без ограничения по времени  $t_{0\text{рас}}$  и по скорости  $\omega_{\max}$ .

Для нахождения времени отработки перемещения  $t_{0pac}$  воспользуемся формулой (2):

$$t_{0pac} = \sqrt{\frac{4\varphi_0}{\varepsilon_0}}, \quad (6)$$

Продолжительность разгона и торможения находится по формуле:

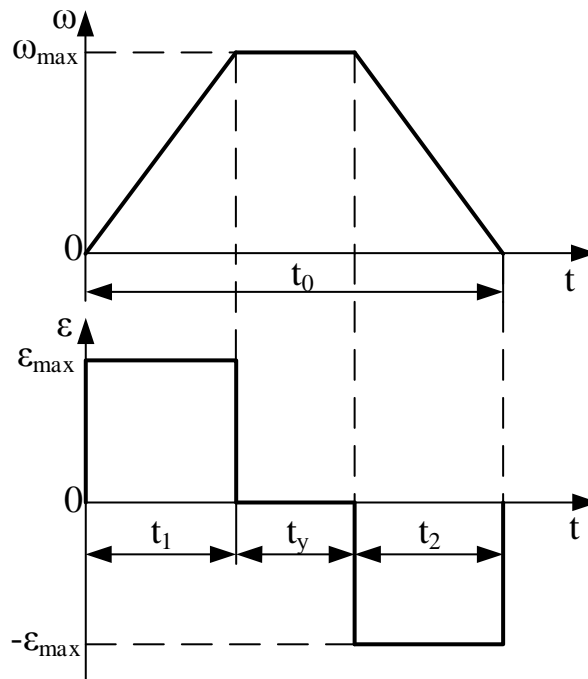
$$t_1 = t_2 = \frac{t_{0pac}}{2}, \quad (7)$$

Максимальная скорость находится по формуле:

$$\omega_{max} = \frac{\varepsilon_0 t_{0pac}}{2}. \quad (8)$$

Диаграмма 3.

Допустим необходимо обработать заданное перемещение  $\varphi_0$  по трапецидальной диаграмме (рис.2) с ограничением по скорости  $\omega_0$  и по времени  $t_0$ , за которое необходимо обеспечить заданное перемещение, но без ограничения по ускорению  $\varepsilon_{max}$ .



*Рис. 2 – Трапецидальная диаграмма*

Максимальное ускорение находится по формуле:

$$\varepsilon_{max} = \frac{\omega_0^2}{\omega_0 t_0 - \varphi}, \quad (9)$$

Продолжительность разгона и торможения находится по формуле:

$$t_1 = t_2 = \frac{\omega_0}{\varepsilon_0}, \quad (10)$$

Продолжительность вращения с установившейся скорости находится по формуле:

$$t_y = t_0 - 2t_1, \quad (11)$$

Диаграмма 4.

Теперь допустим, что необходимо отработать заданное перемещение  $\varphi_0$ , по трапецеидальной диаграмме (рис.2) с ограничением по ускорению  $\varepsilon_0$  и скорости  $\omega_0$ , но без ограничения по времени  $t_{0pac}$  за которое необходимо обеспечить заданное перемещение.

Продолжительность разгона и торможения находится по формуле:

$$t_1 = t_2 = \frac{\omega_0}{\varepsilon_0}, \quad (12)$$

Для нахождения продолжительности работы с установившейся скоростью воспользуемся формулой (1) и выразим  $t_y$ :

$$t_y = \frac{\varphi_0 - \varepsilon_0 t_1^2}{\omega_0}, \quad (13)$$

Продолжительность отработки перемещения определяется по формуле:

$$t_{0pac} = 2t_1 + t_y, \quad (14)$$

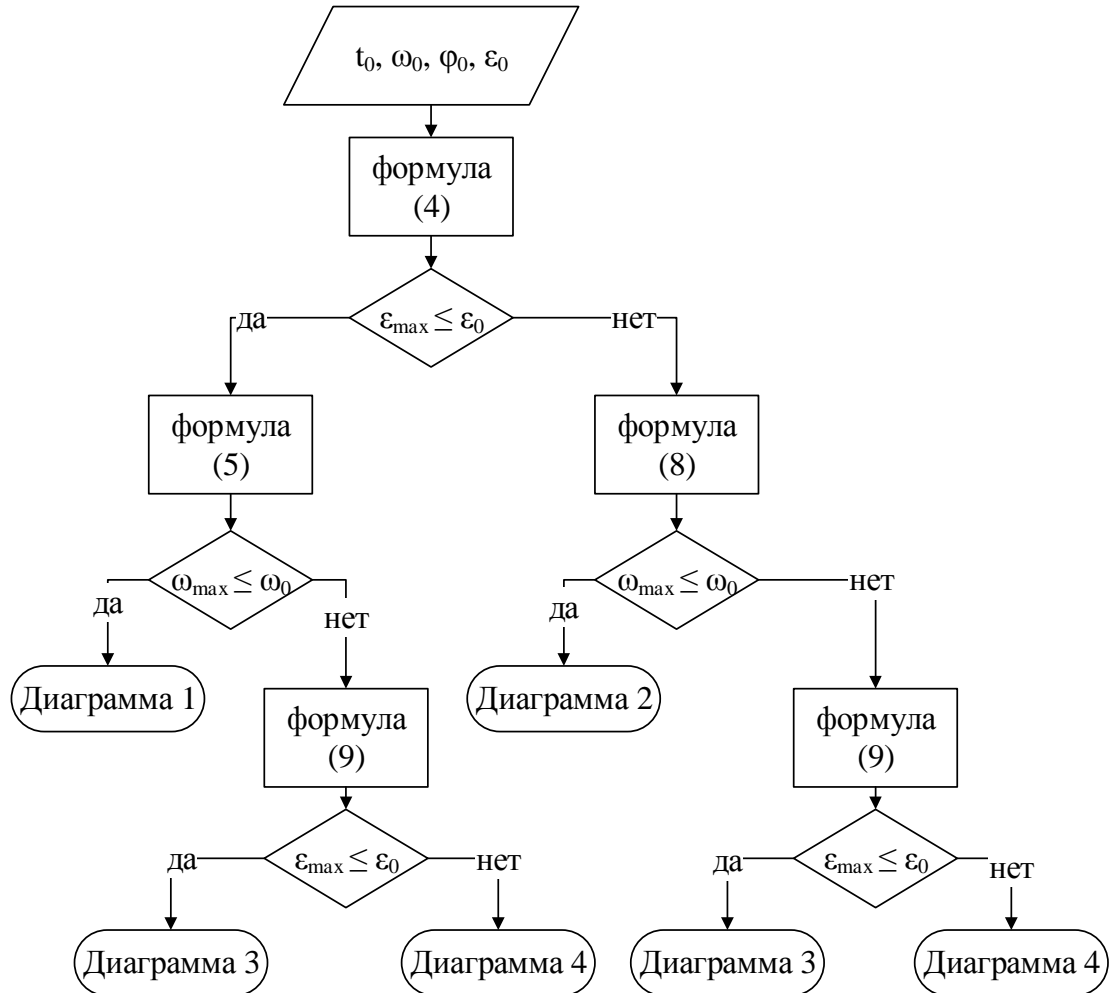
Для выбора диаграммы был разработан алгоритм (рис.3), который на основе входных параметров выбирает оптимальную диаграмму отработки заданного перемещения.

Программа для формирования управляющих импульсов была написана в среде программирования MATLAB – Simulink (рис.4). В параметрах модели был выбран метод ode3 (Bogacki–Shampine) с фиксированным шагом 0.001.

Программа состоит из:

- блока Verification\_data, который входные параметры проверяет в соответствии с алгоритмом (рис.3), выбирает оптимальную диаграмму отработки и выдает пересчитанные параметры в соответствии с формулами (3 –14);
- блока Driver\_figure, который формирует диаграмму на основе пересчитанных параметров;

Данный блок был построен на основе задатчика интенсивности и формирует линейно изменяющийся сигнал до заданного уровня  $w_{max}$  за заданное время  $t_1$ .

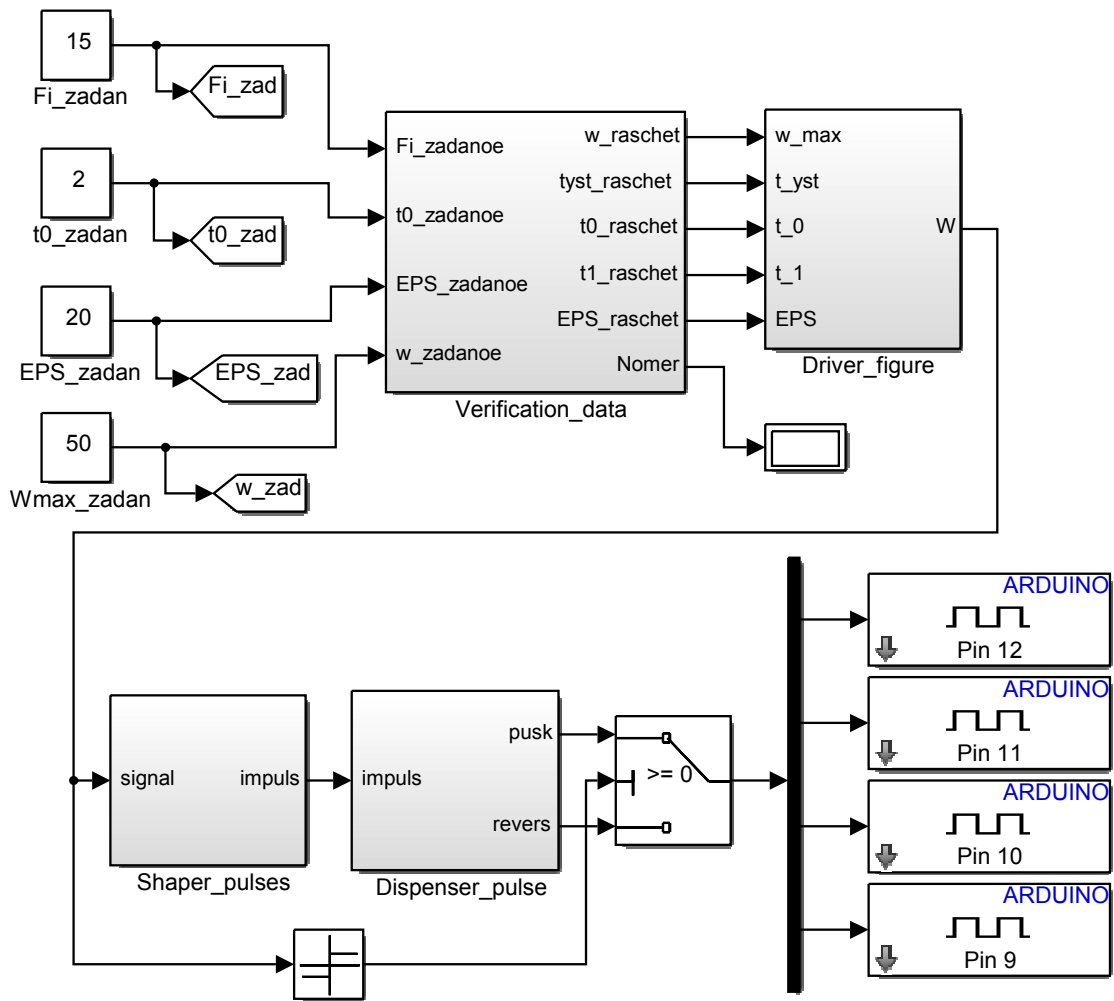


*Рис. 3 – Блок схема алгоритма выбора диаграммы*

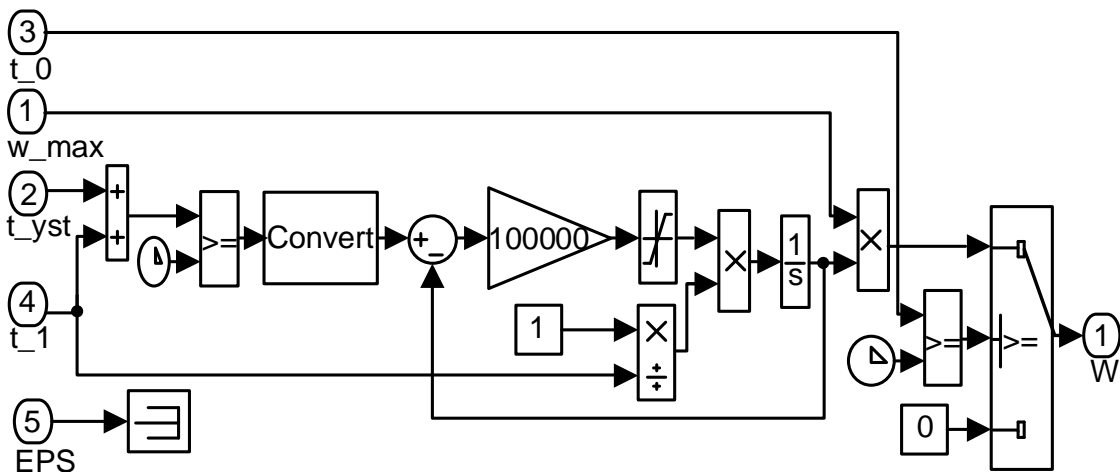
• блока `Sharer_pulses`, который преобразует диаграмму и импульсы;

На рис.6 представлен частотный (FM) модулятор, который в зависимости от амплитуды входного сигнала  $S_m(t)$  изменяет частоту колебательного сигнала  $S_{FM}(t)$ . Работа модулятора описывается уравнением [4]:

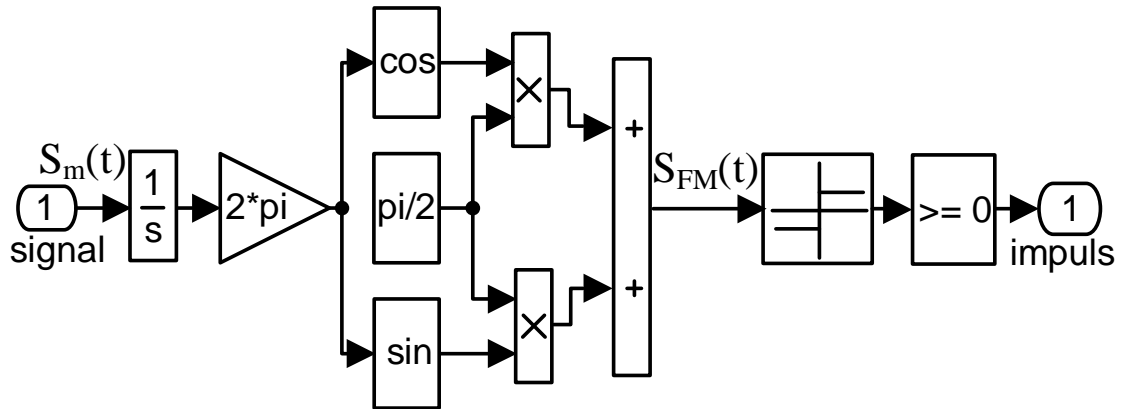
$$S_{FM}(t) = \frac{\pi}{2} (\cos(2\pi \int S_m(t) dt) + \sin(2\pi \int S_m(t) dt)) \quad (15)$$



*Рис. 4 – Программа, для работы шаговым электроприводом*



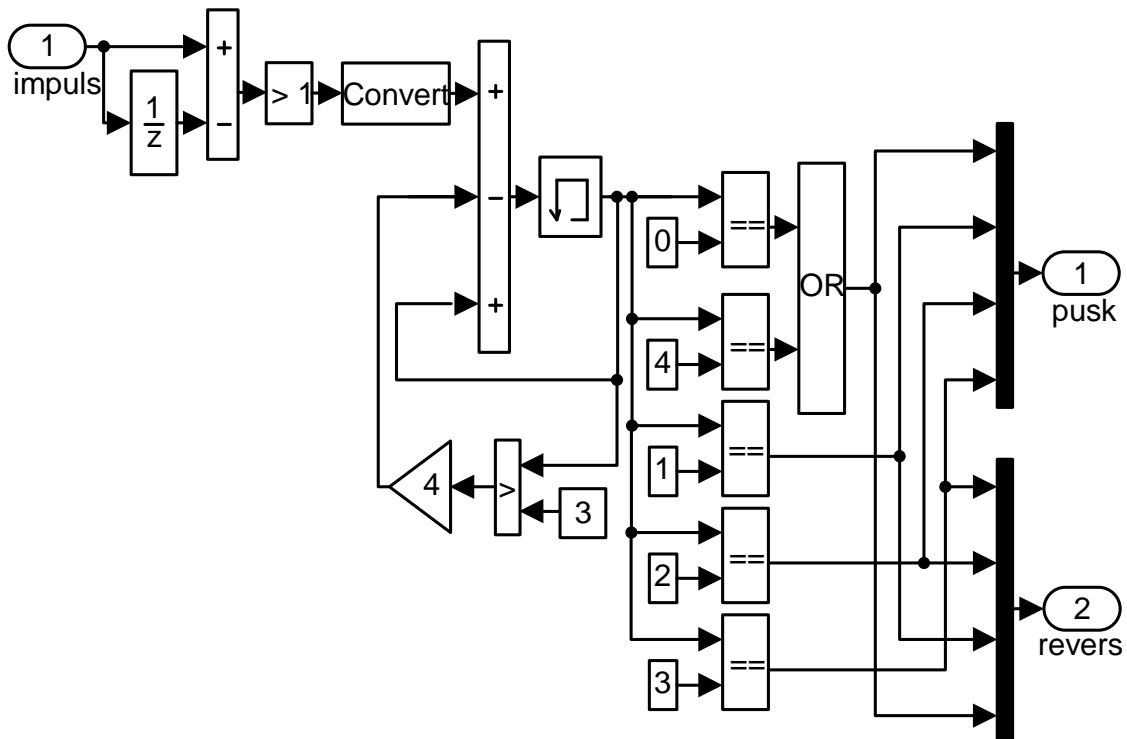
*Рис. 5 – Блок Driver\_figure*



*Рис. 6 – Блок Shaper\_pulses*

- блока Dispenser\_pulse, который распределяет поток импульсов на четыре канала.

Данный блок уменьшает длительность входных импульсов до одной дискреты, далее с помощью логических и математических операций распределяет эти импульсы по четырем каналам. В данном случае для управления шаговым двигателем используется полношаговый режим без перекрытия фаз.



*Рис. 7 – Блок Dispenser\_pulse*

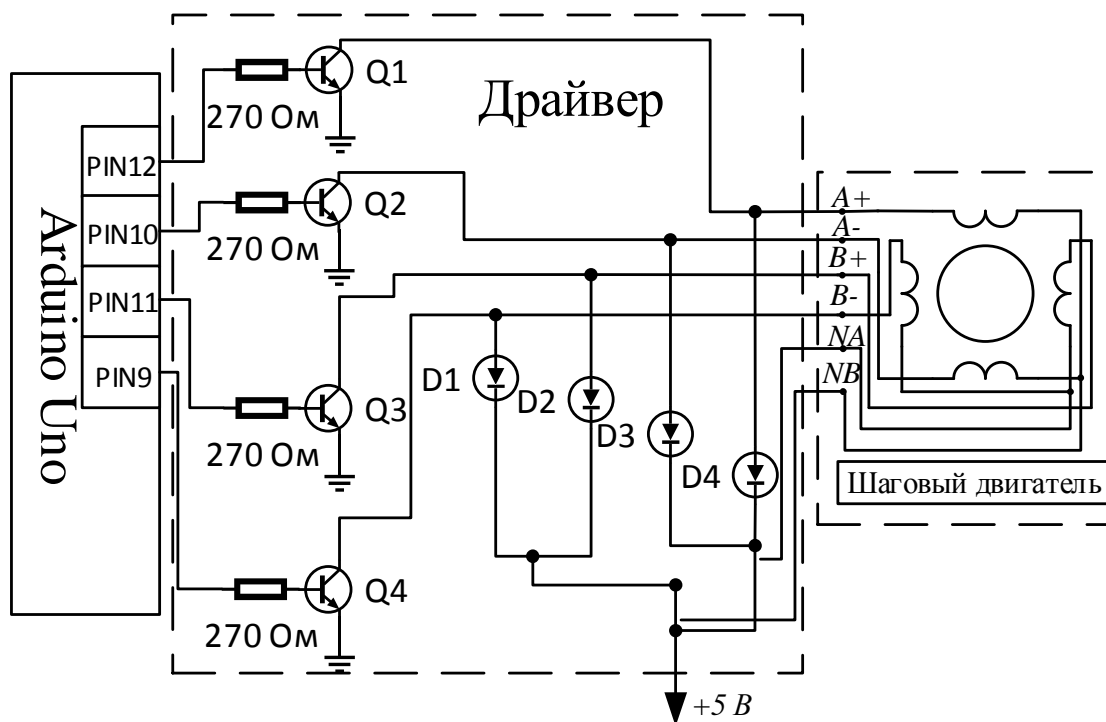
В качестве электропривода был выбран гибридный униполярный шаговый двигатель фирмы Sanyo Denki с углом поворота  $1.875^\circ$  на



один шаг, а в качестве формирователя импульсов используется плата Arduino Uno.

Arduino Uno построена на базе микроконтроллера ATmega328, имеет 14 цифровых входов/выходов (из них 6 могут использоваться в качестве ШИМ-выходов), 6 аналоговых входов, кварцевый резонатор, разъем USB, разъем питания, разъем для внутрисхемного программирования (ICSP) и кнопка сброса. Рабочее напряжение составляет 5В, а максимальный выходной ток одного выхода составляет 40 мА [5].

Arduino в соответствии с программой формирует управляющие импульсы и через соответствующие пины подает их на драйвер шагового двигателя. Драйвер состоит из 4 транзисторов, 4 резисторов и 4 встречно включенных диодов. Драйвер подключается к плате, источнику постоянного напряжения 5 В и непосредственно шаговому двигателю [3].

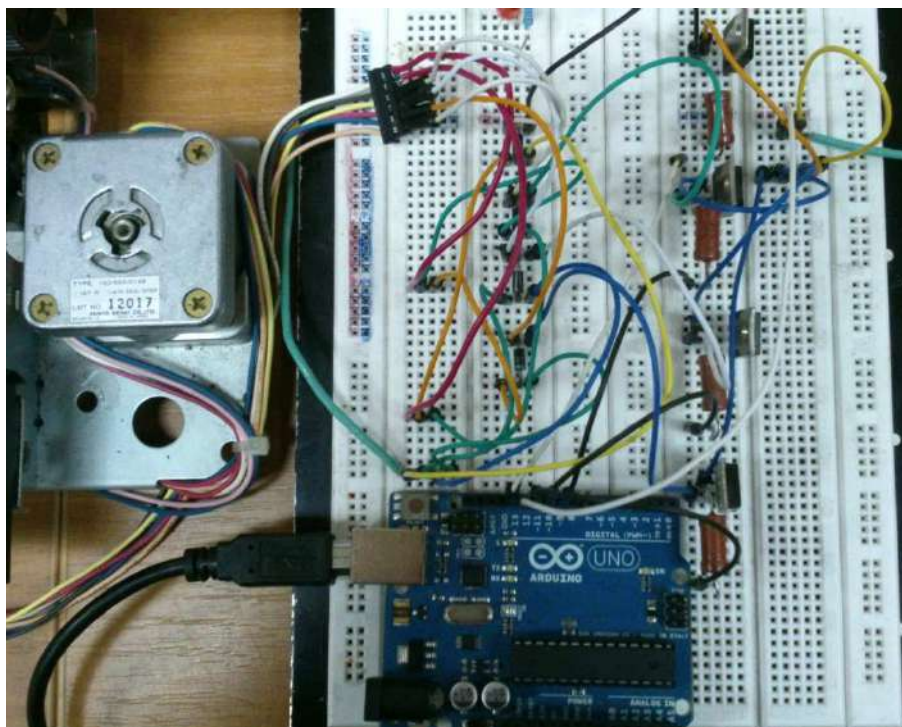


*Рис. 8 – Функциональная схема системы управления*

В драйвере используются диоды, так как обмотки двигателя представляют собой индуктивность, а это значит, что ток идущий через индуктивность не может изменяться мгновенно. При подключении обмоток к источнику питания ток будет с некоторой скоростью нарастать, а при отключении обмотки – произойдет выброс напряжения (напряжение самоиндукции), который способен

повредить транзисторы (ключи), поэтому для ограничения этого выброса используются диоды (D1...D4) [3].

На рис. 9 представлена реальная система, где драйвер был собран на наборной плате с использованием контактных проводов.



*Рис. 9 – Внешний вид системы управления и шагового двигателя*

Работоспособность системы, представленной на рис. 9, была проверена на практике и показала отработку заданного перемещения с максимальным быстродействием. Система работоспособна, если требуется частота выходных импульсов на драйвер не более 1000 Гц. Для увеличения частоты или точности требуется уменьшить величину шага моделирования в настройках модели-программы.

Для решения более сложных задач плата Arduino Uno может не обеспечить приемлемый результат, так как, при программировании Arduino с помощью MATLAB – Simulink, в плату загружается дополнительный системный код, который перегружает процессор. Поэтому для решения более сложных задач следует использовать плату с большей вычислительной способностью. На данном этапе Arduino Uno обеспечивает необходимый результат и выполняет поставленную задачу.

### **Выводы**

1. Разработан алгоритм управления шаговым электроприводом, обеспечивающий отработку заданного перемещения с максимальным быстродействием.
2. Реализована и проверена работоспособность системы управления на реальном шаговом двигателе на платформе ARDUINO и пакета MATLAB.

### **Перечень ссылок**

1. Черенков Н.С. Возможности применения шагового двигателя в промышленности [Электронный ресурс] / Н.С. Черенков, Г.А. Матул – М: «Студенческий научный форум» – 2014 – Режим доступа к статье: <http://www.scienceforum.ru/2014/422/4145>.
2. Семейство драйверов шагового двигателя AMIS-30xxx от ON Semiconductor, [Электронный ресурс]. Режим доступа: <http://www.russianelectronics.ru/leader-r/review/40498/doc/47912>.
3. Контроллер шагового двигателя, [Электронный ресурс]. Режим доступа: <http://www.kazus.ru/shemes/showpage/0/843>.
4. Сигналы с угловой модуляцией. Частотная (FM) и фазовая (PM) модуляция, [Электронный ресурс]. Режим доступа: <http://www.dsplib.ru/content/pmfm/pmfm.html>.
5. Arduino Uno, [Электронный ресурс]. Режим доступа: <http://www.arduino.ua/ru/hardware/Uno>.
6. Программирование Arduino с использованием Simulink, [Электронный ресурс]. Режим доступа: <http://www.youtube.com/watch?v=lb8nPw3ZoA8>.