

**УДК 62-1/-9**

**Звижинская Виктория Николаевна**, Студент кафедры промышленной электроники, Томский государственный университет систем управления и радиоэлектроники, г.Томск

**Сухотерин Максим Витальевич**, Студент кафедры промышленной электроники, Томский государственный университет систем управления и радиоэлектроники, г.Томск

**Горегляд Егор Александрович**, Студент кафедры промышленной электроники, Томский государственный университет систем управления и радиоэлектроники, г.Томск

## **УПРАВЛЕНИЕ УГЛОМ ПОВОРОТА ЛОДОЧНОГО ДВИГАТЕЛЯ**

### **Аннотация**

В статье рассказывается про обеспечение лодочного мотора автономным управлением углом поворота, рассматриваются фрагменты кода. Продемонстрирован механизм работы автономного передвижения угла поворота лодочного мотора двигателя. За основу модернизации был взят двигатель Yamaha 30. Разобран механизм работы передвижения двигателя для поворота лодки. Описаны преимущества удалённого управления надводным устройством, механизм работы шагового двигателя. Расписаны основные пункты на которые необходимо обращать внимание при выборе шагового двигателя. Реализация нахождения центрального положения, срабатывания концевых выключателей и плавного запуска.

### **Annotation**

This article describes how to provide an outboard motor with autonomous steering angle control, including code fragments. The operating mechanism for autonomously adjusting the steering angle of an outboard motor is demonstrated. A Yamaha 30 engine served as the basis for the upgrade. The operating mechanism for

the motor's steering angle is analyzed. The advantages of remote control of a surface device and the operating mechanism of a stepper motor are described. The key points to consider when selecting a stepper motor are outlined. The implementation of center position detection, limit switch actuation, and soft start are discussed.

December 2025

**Ключевые слова:** Yamaha, угол поворота, шаговый двигатель, автопилот, система управления, лодочный мотор.

**Keywords:** Yamaha, steering angle, stepper motor, autopilot, control system, outboard motor.

В современном мире с его технологическим развитием появляется невероятно много возможностей по созданию беспилотных катеров. Автоматизированное, автономное управление системами лодочного мотора является неотъемлемой частью на пути к системе автопилота и удалённого управления надводным устройством.

Автоматизация управления лодочным мотором имеет ряд преимуществ и решает немало значимых задач таких, как:

Снижает физическую нагрузку на оператора, которая может появиться при длительных переходах;

Точность маневрирования при автоматизированном управлении увеличивается (человеческий фактор);

В будущем при автоматизировании других сегментов мотора возможно создание полного автоматического управления надводным устройством (автопилота) и его дистанционное управление с берега;

В реализации данной идеи будет использован лодочный мотор Yamaha 30. Перед нами стоит задача по передвижению рычага управления поворотом мотора из одного положения в другое. За счет движения рычага будет выполняться поворот самого мотора. На рисунке 1 изображена фотография с выделенными некоторыми элементами для более понятного визуального восприятия.



Рисунок 1 – Система управления углом поворота двигателя

Рычаг на данном изображении продублирован оранжевым цветом.

Суть решения поставленной задачи будет заключаться в том, что передвижение рычага будет проводиться с использованием шагового двигателя. Шаговый двигатель на предоставленной фотографии выделен красным цветом. Нам необходимо преобразовать вращательные движения двигателя в линейное перемещение рычага мотора. Данное преобразование можем осуществить с помощью шарико-винтовой передачи (ШВП). В такой конструкции шаговый двигатель необходимо присоединить при помощи муфты к винту ШВП, гайка которого прикреплена к рычагу управления углом поворота. При вращательных движениях мотора будет происходить вращение вала, что в свою очередь приведет к линейному перемещению гайки и соответственно рычага мотора вдоль винта, осуществляя поворот мотора.

Шаговый двигатель преобразует электрическую энергию в механическую, перемещая ротор дискретными шагами. При выборе шагового двигателя важно уделить внимание на крутящий момент, он должен быть достаточным для поворота мотора лодки с учтенным сопротивлением воды, которое будет воздействовать при эксплуатации устройства на воде. В

действительности расчёт нагрузки на валу не представляется возможным так как слишком много факторов внешней среды и самого механизма, влияющих на момент, рассчитать невозможно. В связи с этим параметры шагового двигателя необходимо подбирать экспериментальным путём. Если всё же момент на валу был подобран неправильно то придется программно ограничивать скорость вращения шагового двигателя, так как при больших скоростях будут наблюдаться пропуски шагов что приводит к неправильной работе электроники подсчитывающей шаги, так как обратной связи по шагам не предусмотрено.

Шаговый двигатель имеет ряд преимуществ, например такие, как высокая точность позиционирования, что позволяет точно контролировать угол поворота мотора. Простота управления заключается в подаче всего лишь двух команд сигналов, шага и направления. У шаговых двигателей имеется довольно большой значительный крутящий момент при низких оборотах, это важно для преодоления нагрузок, которые присутствуют при повороте лодочного мотора. Эта особенность будет очень полезна при старте и остановке. Шаговые двигатели имеют достаточно быстрый отклик благодаря чему будет присутствовать оперативное управление с достаточно высокой скоростью переключения.

Прием управляющих команд производится через широтно-импульсная модуляция (ШИМ). ШИМ позволяет передавать команды от внешних устройств к системе. На входные пины поступают импульсы, которые преобразуются в управляющую величину.

В результате использования ШИМ мы получаем то, что система свободно может “общаться” с пультами дистанционного управления, датчиками и всеми необходимыми устройствами. Благодаря универсальному ШИМ происходит отсекание некорректных сигналов, а также появляется немало важная возможность масштабировать диапазон движения двигателя.

При использовании шагового двигателя необходимо учитывать и ограничения, которые могут негативно сказаться на работоспособности

конструкции. К минусам можно отнести то, что при высоких скоростях крутящий момент снижается, а это может ограничить производительность в динамичных режимах. В моменты, когда требуется быстрая реакция и высокая активность, подобное ограничение может оказывать негативное влияние на общую производительность. Шаговый двигатель потребляет энергию даже в статическом режиме, что может привести к избыточному нагреву и сократить срок службы изоляционному слою, а нагрев в свою очередь приводит к снижению коэффициента полезного действия (КПД) системы.

В нашей системе на противоположных концах винта ШВП необходимо установить концевые выключатели, их наличие даст понимание конкретного местоположения гайки. Концевые выключатели будут полезны в нахождении центрального положения путем калибровки. Данные элементы выделены на фотографии синим цветом.

Нахождение центрального положения необходимо для подачи конкретных координат при работе с двигателем. Изначально калибровка производится исходя из координат двух концевиков. Гайка доходит до одного из выключателей сбрасывая свое положение устанавливает новое как нулевое, затем происходит движение в обратную сторону с отсчетом шагов, доходя до положения второго концевика, исходя из имеющихся данных происходит вычисление центрального положения. При отклонении от установленного центрального значения, при соприкосновении с одним из концевиков будет происходить калибровка по одной точке.

В исходном коде нахождение центральной координаты мы реализуем следующими строками в процедуре калибровки:

Для начала необходимо найти нулевую точку. Для нахождения начальной точки, как мы уже рассуждали нужно гайку довести до “нулевого” концевика. В данных строках описана реализация данного действия:

```
digitalWrite(DIR_PIN1, LOW); // задаём направление к нулевому концевiku  
while (digitalRead(CON1_PIN) == 1) { // пока концевик не сработал
```

```

digitalWrite(PUL_PIN1, HIGH);
delayMicroseconds(40); // Генерирование импульса длительностью 40
микросекунд
digitalWrite(PUL_PIN1, LOW);
delayMicroseconds(TIC); // делаем шаг
}
Pos = 0; // фиксируем нулевую позицию

```

Затем двигатель направляет гайку к противоположному концевiku, считая шаги:

```

digitalWrite(DIR_PIN1, HIGH); // меняем направление к конечному концевiku
while (digitalRead(CON2_PIN) == 1) { // пока концевик не сработал
    digitalWrite(PUL_PIN1, HIGH);
    delayMicroseconds(40);
    digitalWrite(PUL_PIN1, LOW);
    delayMicroseconds(TIC);
    Pos++; // увеличиваем счётчик шагов
}
Pos_MAX = Pos; // сохраняем максимальную позицию

```

Этим действием вычисляется длина винта ШВП в количестве шагов.

Следующим действием максимальную длину необходимо разделить пополам, что даст нам определение центрального положения:

```
Pos_TASK = Pos_MAX / 2; // целевая позиция = середина диапазона
```

Например, если Pos\_MAX равно 1400 шагов, то Pos\_TASK будет равно 700 — это и есть центральная точка.

Перед каждым шагом будем проводиться проверка, не дошла ли гайка до крайних точек (концевиков):

```
if (digitalRead(CON1_PIN) == 0) Pos = 0; // если у нулевого концевика
```

```
if (digitalRead(CON2_PIN) == 0) Pos = Pos_MAX; // если у конечного концевика
```

Когда система получает команду к движению, происходит сравнение текущей позиции (Pos) с целевой (Pos\_TASK):

```
if ((Pos_TASK > Pos + (POSTASK_GIS / 2)) && digitalRead(CON2_PIN) == 1)
{
    // двигаться вперед
}
if ((Pos_TASK < Pos - (POSTASK_GIS / 2)) && digitalRead(CON1_PIN) == 1) {
    // двигаться назад
}
```

В данном коде можно заметить гистерезис (POSTASK\_GIS). Гистерезис, иными словами зона нечувствительности нам необходима для того, чтобы предотвратить осцилляции системы из за шума в сигнале приёмника.

При управлении данной системой через пульт, когда рычаг управления на пульте будет находиться в нейтральном положении, мы можем наблюдать функцию возврата гайки в центральное положение, двигатель будет приходить так же в центральную точку (слежение за сигналом в реальном времени). Движение катера в данном случае будет плавно выравниваться и переходить к движению по прямому пути.

При работе с данной задачей можно столкнуться с проблемой срыва шагов, которые выдает двигатель. Срыв шагов при резком запуске может привести к потере позиционирования, двигатель сбивает за счет чего фактический угол поворота руля не соответствует командному сигналу. Для устранения проблемы было принято решение сделать плавный запуск. Плавный запуск двигателя также снижает механические нагрузки и электрические перегрузки, которые возникают при резком старте, а резкий старт в свою очередь нуждается в скачкообразном увеличении тока в обмотках. Плавный запуск шагового двигателя сделает систему более надежной. При резком повороте лодки может произойти сильное отклонение

корпуса от горизонтального положения, появиться крен. Если запуск будет плавно нарастающим, то данный эффект можно избежать.

В написании кода был реализован процесс, который будет начинаться с медленных шагов и постепенно ускоряться до номинальной скорости. В нашем случае мы сделали это следующим образом:

Ниже продемонстрированы ключевые параметры кода. В программе задаются три критических параметра:

```
#define TIC 90          // базовая задержка (мкс) — рабочая скорость
#define TIC_START 250  // начальная задержка (мкс) — старт
#define Smooth_STEP 1600 // количество шагов разгона
```

Сброс счётчика шагов при начале движения:

```
smooth_c = Smooth_STEP; // начинаем разгон с полного счётчика
```

На следующем пункте нам необходимо рассчитать текущую задержку, для этого на каждом шаге происходит вычисление актуальной задержки между импульсами:

```
smooth_t1 = map(constrain(smooth_c, 0, Smooth_STEP), 0, Smooth_STEP, 0, TIC_START);
```

Из базовой и динамической составляющей складывается итоговая пауза между шагами:

```
time2 = micros() + TIC + smooth_t1;
```

Производим декремент нашего счетчика:

```
if (smooth_c > 0) {
  smooth_c--;
}
```

В данных строках после каждого шага происходит уменьшение счётчика. Когда переменная `smooth_c` достигает 0, система приходит к начальному состоянию, двигатель начнет работать на полной скорости.

После необходимо произвести сброс при остановки двигателя. Сброс выполняется за счет того, что при достижении целевой позиции счётчик обнуляется для реализации следующего плавного старта:

```

if ((Pos_TASK < Pos + POSTASK_GIS && Pos_TASK > Pos - POSTASK_GIS)
|| (SUU_PFO == 1)) {
    smooth_c = Smooth_STEP;
    SUU_PFO = 0;
}

```

В данной системе управление двигателем и соответственно лодкой будет производиться дистанционно с помощью пульта управления. При управлении водным аппаратом с берега появляется необходимость в трансляции изображения камеры оператору для понимания местоположения на водном пространстве, не мало важно также оборудовать устройство эхолотом для определения глубины водоёма, изучения рельефа и обнаружения препятствий под водой. Данное оборудование позволит обеспечить безопасность окружающим и содержимого беспилотного катера.

### Список литературы

1. Концевые выключатели: виды, устройство и принцип работы // <https://www.elec.ru/publications/promyshlennoe-oborudovanie/7473/> .
2. Лодочный мотор yamaha 30hwcs | лодочный мотор ямаха 30hwcs // [https://dzen.ru/a/ZoLTr6PxnliyMy\\_O](https://dzen.ru/a/ZoLTr6PxnliyMy_O) .
3. Шаговые двигатели: принцип работы, применение и управление // <https://inner.su/articles/shagovye-dvigateli/> .
4. Шаговые двигатели. Принцип работы и управление. // <https://darxton.ru/wiki-article/shagovye-dvigateli-printsip-raboty-i-upravlenie/?ysclid=mj3u4odk9c717869307> .
5. ШИМ-регулирование скорости электродвигателей // <https://inner.su/articles/shim-regulirovanie-skorosti-elektrodvigateley/> .

### References

1. Limit Switches: Types, Design, and Operating Principle // <https://www.elec.ru/publications/promyshlennoe-oborudovanie/7473/> .

2. Yamaha 30hwc Outboard Motor | Yamaha 30hwc Outboard Motor // [https://dzen.ru/a/ZoLTr6PxnliyMy\\_O](https://dzen.ru/a/ZoLTr6PxnliyMy_O) .
3. Stepper Motors: Operating Principle, Application, and Control // <https://inner.su/articles/shagovye-dvigateli/> .
4. Stepper Motors. Operating Principle and Control. // <https://darxton.ru/wiki-article/shagovye-dvigateli-printsip-raboty-i-upravlenie/?ysclid=mj3u4odk9c717869307> .
5. PWM Speed Control of Electric Motors // <https://inner.su/articles/shim-regulirovanie-skorosti-elektrodvigateley/> .