

РАЗРАБОТКА СИСТЕМЫ ОРИЕНТИРОВАНИЯ В ПРОСТРАНСТВЕ С ИСПОЛЬЗОВАНИЕМ МОДУЛЯ GY-531

Н.А. Горин, В.Н. Струнилин

Донецкий национальный технический университет, г. Донецк
кафедра компьютерная инженерия
MegaNikolai@yandex.ua

Аннотация

Горин Н.А., Струнилин В.Н. Разработка системы ориентирования в пространстве с использованием модуля GY-531. Выполнена разработка электронного прибора отслеживающего при помощи акселерометра и гироскопа положение объекта и генерирующего сигналы управления для корректировки работы двигателей.

Ключевые слова: микроконтроллер, акселерометр, гироскоп, микросхема, драйвер.

Постановка проблемы. При взаимодействии роботизированной техники с окружающими объектами возникает необходимость в системах ориентирования в пространстве. Эти системы необходимы так же в тех случаях, когда оператор какой-либо движущейся машины или устройства не может достоверно оценивать перемещение управляемого объекта в пространстве и его взаимодействие с другими объектами. С развитием научно-технического прогресса данная проблема становится всё актуальнее. Системы ориентирования в пространстве всё больше проникают в нашу повседневную жизнь: от мобильных телефонов (датчики приближения, GPS, акселерометр и многое другое) до автомобилей (GPS, парктроник). Внедрение данных систем в аэрокосмическую отрасль является неотъемлемой частью прогресса в этой области. Следовательно, исследование и разработка систем ориентирования в пространстве является актуальной.

Цель статьи – разработка электронного прибора, отслеживающего при помощи акселерометра и гироскопа положение объекта в пространстве, генерирующего управляющие сигналы для корректировки работы двигателей.

Выбор платформы. Разработка системы ориентирования в пространстве выполнена на платформе ArduinoUno (рис.1). Данная платформа построена на основе микроконтроллера ATmega328 [1]. Платформа имеет 14 цифровых входов/выходов (шесть из которых могут использоваться как выходы широкополосной импульсной модуляции), шесть аналоговых входов, кварцевый генератор 16 МГц, разъем USB, силовой разъем, разъем ICSP и кнопку перезагрузки. Для работы необходимо подключить платформу к

компьютеру посредством кабеля USB, либо подать питание при помощи адаптера AC/DC или батареи [1].



Рисунок 1 – Внешний вид ArduinoUno

Платформа программируется посредством программного обеспечения Arduino. Язык программирования устройств Arduino основан на C/C++ [2, 3].

Для программирования микроконтроллера используется среда разработки Arduino 1.0.5. Микроконтроллер ATmega328 поставляется с встроенным загрузчиком, облегчающим запись новых программ без использования внешних программаторов.

Обоснование выбора модуля GY-531. Микросхема MPU6050 содержит акселерометр, гироскоп и температурный сенсор. MPU6050 является главным элементом модуля GY-531 (рис.2). Помимо этой микросхемы на плате модуля расположена необходимая обвязка MPU6050, в том числе подтягивающие резисторы интерфейса I2C, а также стабилизатор напряжения на 3,3 вольта с малым падением напряжения (при питании в 3,3 вольта на выходе стабилизатора будет 3 вольта) с фильтрующими конденсаторами. Также на плате имеется SMD светодиод с ограничивающим резистором, выполняющие функцию индикации питающего напряжения.



Рисунок 2 - Внешний вид GY-531

Математический аппарат. Данные, приходящие с гироскопа имеют следующий вид:

$$gyro_x_scaled = \frac{d}{dt} \theta_x^{gyro};$$

$$gyro_y_scaled = \frac{d}{dt} \theta_y^{gyro};$$

$$gyro_z_scaled = \frac{d}{dt} \theta_z^{gyro}.$$

`gyro_x_scaled`, `gyro_y_scaled` и `gyro_z_scaled` – это переменные, в которые библиотечная функция `MPU6050_ReadData()` заносит значения, считанные с гироскопа с учётом калибровки.

В дальнейшем будем рассматривать все расчёты на примере оси *x*. Для расчёта угла необходимо проинтегрировать переменную “`gyro_x_scaled`”.

Для реализации дискретного интегрирования, будем использовать метод Эйлера как один из самых популярных алгоритмов. Математически интегрирование методом Эйлера можно записать следующим образом:

$$\theta_x^{gyro}(t_n) = gyro_x_scaled * T + \theta_x^{gyro}(t_{n-1});$$

$$T = t_n - t_{n-1},$$

где $n=\{1, 2, 3, \dots\}$ это количество итераций.

Предполагаем, что начальные углы относительно осей *x*, *y*, *z* после калибровки равны 0, 0 и 90 градусов соответственно, так что для итерации $n=0$:

$$\theta_x^{gyro}(t_0) = 0^0, \theta_y^{gyro}(t_0) = 0^0, \theta_z^{gyro}(t_0) = 90^0.$$

Из-за неидеальной калибровки гироскопа, “`gyro_x_scaled`” никогда не равна нулю и со временем “`angle_x_gyro`” изменяет свои значения. Для решения данной проблемы, проводится расчёт угла с помощью акселерометра и полученные значения сравниваются с углом гироскопа. Так как модуль MPU6050 располагается горизонтально, ускорение по оси *z* равно $1g$ (то есть, $9,81 \text{ м/с}^2$). Используем этот вектор ускорения и его проекцию на ось *y* для расчёта угла между осями *x* и *y*.

Угол, который рассчитывается с помощью акселерометра, рассчитывается по зависимости:

$$\theta_x^{accel} = \tan^{-1} \frac{accel_x_scaled}{\sqrt{accel_y_scaled^2 + accel_z_scaled^2}};$$

$$\theta_y^{accel} = \tan^{-1} \frac{accel_y_scaled}{\sqrt{accel_x_scaled^2 + accel_z_scaled^2}}.$$

Основными проблемами при определении угла наклона с помощью акселерометра являются: сильная зашумленность сигнала и очень сильная чувствительность к вибрациям, без которых ни один механизм не работает.

Более того, при перемещении MPU6050 вдоль одной из осей координат, полученные значения угловых скоростей будут мешать расчету угла. Для лучшего результата углы с гироскопа и акселерометра обрабатываются с помощью фильтра:

$$\theta_x = Filter_gain * \theta_x^{gyro} + (1 - Filter_gain) * \theta_x^{accel}.$$

Окончательно уравнение для определения угла наклона принимает вид:

$$\theta_x^{gyro}(t_n) = gyro_x_scaled * T + \theta_x(t_{n-1}).$$

В дальнейшем использовался коэффициент усиления 0,95. В зависимости от динамики системы, можно его повышать, но не до единицы, так как значения будут сильно отклоняться от истинных.

Принцип работы. Разработанная система ориентирования в пространстве должна определять угол подъема или спуска движущегося объекта и в зависимости от этого генерировать управляющие сигналы для увеличения/уменьшения напряжения, подаваемого на двигатели. Если объект преодолевает подъем, система определяет угол подъема и увеличивает напряжение, подаваемое на двигатели, чтобы не утратить скорость движения. Если объект преодолевает спуск, система определяет угол спуска и уменьшает напряжение на двигатели чтобы предотвратить неконтролируемый разгон объекта.

Моделирование производилось в утилите Virtual Bread Board. Virtual Bread Board – это симулятор макетной платы Arduino для моделирования цифровых схем. Утилита позволяет писать код для платформы Arduino непосредственно в эмуляторе и одновременно проверить его работоспособность на компьютере без самого контроллера Arduino.

Схема подключения модуля GY-531 для получения системы ориентирования в пространстве представлена в табл.1. [4]

Таблица 1 – Схема подключения модуля GY-531

Выход/вход модуля GY-531	Выход/вход платформы ArduinoUno
VCC	3,3V
GND	GND
SCL	A5
SDA	A4
ADO	GND
INT	D2

Следует отметить, что в данном проекте использовался акселерометр и гироскоп одновременно, чтобы повысить точность полученных данных. Это обусловлено разной спецификой работы акселерометра и гироскопа, а также определяемых ими параметров.

Далее необходимо разработать систему для управления двигателем. В данной системе двигатель будет управлять той же платформой при помощи

микросхемы драйвера двигателей L293D. Система управления двигателем (рис.3) моделировалась отдельно от модуля GY-531 при помощи макетной платы.

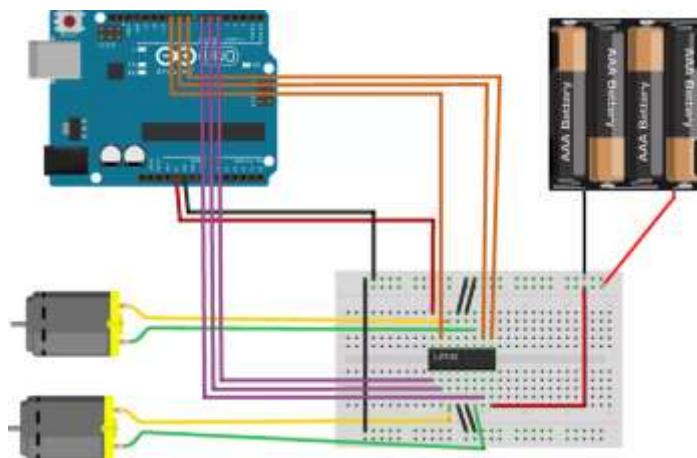


Рисунок 3 – Схема управления двигателями

Выводы. Выполнена разработка и моделирование электронного прибора, определяющего положение объекта в пространстве при помощи акселерометра и гироскопа. Устройство определяет углы подъёма или спуска и генерирует сигналы управления для корректировки работы двигателей. Полученная система ориентирования в пространстве простая в применении и дешёвая при сборке. Меняя микросхемы драйвера двигателей возможно управление данной системой электродвигателями любой мощности. Для программирования микроконтроллера используется язык C. Среда разработки транслирует программу на язык Assembler – понятный для микроконтроллера.

Список литературы

1. Платформа Arduino Uno [электронный ресурс] // Официальный сайт Ардуино в России: [сайт]. [2016]. URL: <http://arduino.ru/Hardware/ArduinoBoardUno>.
2. Официальный сайт Arduino [электронный ресурс]: [сайт]. URL: <http://arduino.cc>.
3. Подбельский В. В. Программирование на языке Си: Учеб. пособие. – 2-е доп. изд./ В. В. Подбельский, С. С. Фомин. – М.: Финансы и статистика, 2004. – 600 с., ил.
4. Arduino Playground [электронный ресурс] // Энциклопедия знаний по MPU-6050: [сайт]. [2016]. URL: <http://playground.arduino.cc/Main/MPU-6050>.