

УДК 004

ПРИМЕНЕНИЕ ГЛОБАЛЬНОЙ НАВИГАЦИОННОЙ СИСТЕМЫ ДЛЯ УПРАВЛЕНИЯ КОЛЕСНОЙ ПЛАТФОРМОЙ

Черных А.А.

Научный руководитель: Тутов И.А.

*Национальный Исследовательский Томский политехнический университет,
634050, Россия, г. Томск, пр. Ленина, 30
E-mail: russk1j@mail.ru*

В статье рассказывается про разработку колесной платформы и применение глобальной навигационной системы для ее навигации в пространстве.

The article is about the wheel platform development and its navigation using global navigation systems.

Ключевые слова: микроконтроллер, курс, GPS модуль, колесная платформа.

Key words: microcontroller, course, GPS module, wheeled platform.

Введение

Мобильные робототехнические системы применяются сегодня в самых разных отраслях. Большое количество роботов, ориентированных на работу на открытом пространстве и в идеале должны уверенно перемещаться в незнакомой и непредсказуемой обстановке реального мира. Для успешной навигации в пространстве бортовая система робота должна уметь строить маршрут, управлять параметрами движения (задавать угол поворота колес и скорость их вращения), правильно интерпретировать сведения об окружающем мире, получаемые от датчиков, и постоянно отслеживать собственные координаты. Один из вариантов определения пространственных координат – использование GPS навигации.

Конструкция платформы и элементная база

В качестве экспериментальной платформы использовалась мобильная полноприводная колесная платформа с рулевым управлением на основе принципа Аккермана.

Оборудование:

- Отладочная плата с микроконтроллером Atmega 2560;
- GPS модуль u-blox 6;
- Мобильная колесная платформа 30×40 см с рулевым управлением на основе принципа Аккермана (рис. 1).
- Силовой модуль с драйвером L298N.

Для управления вращением двигателя используется силовая часть, основанная на драйвере двигателя постоянного тока L298N. Управляющий сигнал подается с микроконтроллера на драйвер. К драйверу подключен внешний источник питания 5В. Данный драйвер способен выдерживать ток до 2А на канал, что достаточно для управления данным двигателем. Также драйвер позволяет управлять скоростью двигателей с помощью широтно-импульсной модуляции (ШИМ).



Рис. 1. Мобильная колесная платформа 30×40 см

Принцип движения робота по глобальной навигационной системе

Прием географических координат и вычисление требуемого курса для робота по формулам. Рассматриваем две точки, координата одной из которых определена GPS модулем (текущая координата робота), а вторая – это наша цель (путевая точка). Находим курс в градусах. Ниже представлен листинг фрагмента кода, написанный на языке C++ по расчету требуемого курса.

```

dlon = radians(targetLong-currentLong); // targetLong - долгота путевой точки
// currentLong - долгота текущей точки
cLat = radians(currentLat); // currentLat - широта текущей точки
tLat = radians(targetLat); // targetLat - широта путевой точки
d1 = sin(dlon) * cos(tLat);
d2 = sin(cLat) * cos(tLat) * cos(dlon);
d2 = cos(cLat) * sin(tLat) - d2;
d2 = atan2(d1, d2);
targetHeading = degrees(d2); // targetHeading - курс в градусах
  
```

Требуется также определять текущий курс робота, как и требуемый, и по ошибке компенсировать поворотом рулевого механизма до совпадения текущего курса робота с требуемым. На достоверность курса будет влиять точность GPS. Из проведенных опытов с данным модулем заявленная точность в 2,5 метра была подтверждена, но в некоторые моменты из-за ухудшения приема сигнала наблюдались выбросы. Это может быть с любым подобным модулем, что существенно может повлиять на значение курса в некоторый момент времени при расчете, особенно если робот находится уже непосредственно вблизи первой маршрутной точки. Поэтому дальше будет рассмотрена задача фильтрации в реальном времени.

Заключение

Для дальнейшей отладки и апробации алгоритма системы навигации робота была собрана платформа под управлением AVR микроконтроллера ATmega 2560 с подбором элементной базы. Рассмотрен алгоритм вычисления требуемого курса движения робота по GPS

данным и написана по нему программа на языке C++. Следующим этапом будет рассмотрена задача фильтрации потока GPS данных, а также метод определения текущего курса робота.

Список литературы

1. Навигация мобильных роботов [Электронный ресурс]. URL: <http://www.computer-museum.ru/frgnhist/robonav.htm> Режим доступа: свободный (дата обращения: 15.02.2016)
2. Datasheet на микроконтроллер ATmega 2560.
3. Datasheet на GPS модуль U-blox NEO 6m.
4. Баранов В.Н. Применение Микроконтроллеров AVR: схемы, алгоритмы, программы. – Додэка-XXI.

УДК 004

ЭЛЕКТРОННЫЕ ЧАСЫ С ГАЗОРАЗРЯДНЫМИ ИНДИКАТОРАМИ

Демченко А.А.

Научный руководитель: Тутов И.А., ассистент кафедры ИКСУ

Национальный Исследовательский Томский политехнический университет,
634050, Россия, г. Томск, пр. Ленина, 30
E-mail: aaa55@tpu.ru

Введение

В данной статье рассказывается об опыте создания часов с газоразрядной индикацией. Часы в наше время не утратили актуальности. Темп жизни современного общества требует контроля времени, фиксации его затрат и ресурсного отношения. Мобильный телефон, в современности, имеющийся у каждого, для этой задачи не подходит, – его не удобно каждый раз доставать из кармана или сумки. Поэтому удобней было бы использовать настольные часы. Кроме того, настольные часы считаются необходимой частью любого интерьера с точки зрения психологии – наблюдение за сменой секунд на табло помогает сконцентрироваться и перевести дух.

Разработка конструкции часов

Была поставлена задача создать часы, отвечающие следующим требованиям: индикация осуществляется с помощью газоразрядных индикаторов; установка времени должна производиться с помощью кнопки-энкодера; должен присутствовать будильник; при отключении питания от сети не должно обнуляться время; часы должны быть реализованы на микроконтроллере семейства AVR; при отключении питания от сети не должен прекращаться отсчет времени, должна только отключаться индикация.

Часы представляют собой несколько таймеров, каждый из которых отвечает за свои функции. *Timer0* отвечает за динамическую индикацию цифр на табло. *Timer1* отвечает за сигнал будильника. *Timer2* отвечает за непосредственный отсчет времени в часах. Функции *time_h_inc* и *time_h_dec* отвечают за счет часов времени в диапазоне от 0 до 23, а функции *time_m_dec* и *time_m_inc* отвечают за то, чтобы минуты времени не выходили за диапазон 0-59. *Time_s_dec* и *time_s_inc* выполняют аналогичную функцию для секунд. Функции *time_alarm_h_inc*, *time_alarm_h_dec*, *time_alarm_m_inc*, *time_alarm_m_dec*, *time_alarm_s_inc*, *time_alarm_s_dec* выполняют функции, аналогичные функциям, используемым в часах без приставки «alarm». Вращение энкодера по часовой стрелке инициирует запуск функции