

Применение популярных протоколов и свободного ПО в управлении мобильным роботом

Андрей Дунец, Брест, Беларусь*

We consider the problems of telemetry, positioning and control of the mobile robot for rivers and lakes monitoring. The project uses Bluetooth technology to transmit the telemetry data. RTKLIB library implementation of Real-Time Kinematic algorithm is used for positioning. Current version of the control system is based on the Ardupilot project.

Мониторинг водоемов актуален для решения самых разных задач. Начиная от построения профиля дна водоема для прогнозирования паводков и обеспечения безопасности судоходства и заканчивая контролем состояния воды в интересах рыболовецких хозяйств и служб охраны природных ресурсов. Применение мобильных водоплавающих роботов позволяет автоматизировать процесс мониторинга и значительно снизить его себестоимость.

На данный момент робот и его система управления находится в разработке. Текущие решаемые задачи это телеметрия, позиционирование и управление роботом.

Телеметрия обязательная часть подобного комплекса, так как без неё нормальная разработка ПО и отладка оборудования сильно затруднена. Выбор был сделан с расчетом на доступные популярные решения, так как особенности этих технологий хорошо известны и они недороги в отличии от решений, которые основаны на закрытых протоколах.

Первоначальным вариантом было использование Bluetooth Class 1. Заявленная дальность до 100 метров, дешевизна оборудования, простота настройки позволили поставить несколько экспериментов по передаче телеметрических данных. Использовался профиль SPP (Serial Port Profile). На стороне робота микроконтроллер автопилота связывался с компьютером через Bluetooth модем BTM-222, который полностью реализует весть стек необходимых беспроводных протоколов профиля SPP, предоставляет TTL

*dunets@gmail.com, <http://lvee.org/ru/abstracts/143>

UART-интерфейс и управляется AT-командами. На стороне компьютера применялся встроенный адаптер Bluetooth. Выяснилось, что приемо-передатчики, кроме заявленных характеристик должны быть оснащены соответствующими антеннами. Удалось использовать на стороне робота антенну от адаптера WiFi (частотный диапазон один и тот же) и практическая полученная дальность составила 30 метров.

Эксперименты показали, что Bluetooth можно использовать для передачи телеметрической информации на небольшие расстояния. Но профиль SPP, будучи простым в использовании, позволяет передавать только один поток данных. На практике нужно как минимум два потока: поток с приемника GPS и поток консоли. Можно использовать протокол, который поддерживает подобное мультиплексирование. Возможны так же экзотические варианты: пробросить TCP трафик через PPP.

Дополнительно выявились проблемы с восстановлением связи при утере соединения: пересопряжение устройств требует поддержки в прикладном ПО, что неудобно. Сейчас как наиболее оптимальное решение для следующего шага рассматривается переход на WiFi. WiFi обеспечит большую скорость передачи данных. Можно использовать более мощные антенны. Штатно доступный стек TCP/IP позволит мультиплексировать данные. При этом возрастет энергопотребление, что является минусом данного решения. Эксперименты покажут, какой вариант наиболее оптимален.

Для позиционирования робота в акватории водоема применяется GPS. Для уточнения координат используется алгоритм RTK (Real-Time Kinematic), который реализован в открытой библиотеке RTKLIB (rtklib.com). Для получения данных GPS применяется приемник NEO-6M фирмы u-blox и штатные утилиты из состава RTKLIB. Приемник настраивается с помощью хаков, разработанных сообществом OpenStreetMap. БД корректировок берутся с серверов проекта IAG Reference Frame Sub-Commission for Europe (<http://www.euref.eu/>), базовые станции которого находятся на достаточно близком расстоянии от Беларуси.

Первоначальное решение по управлению, на которое был сделан упор — использовать существующие автопилоты, в которых используется открытое ПО. Одним из таких проектов является APM Autopilot Suite (<http://ardupilot.com>). Его особенности:

- открытые исходные тексты ПО автопилота и базовой станции (лицензия GNU GPL);

- базовые аппаратные решения основаны на открытой платформе Arduino (Atmel Atmega), но более продвинутые используют 32-разрядные контроллеры ARM Cortex и платы с закрытым дизайном;
- активное сообщество пользователей DIY Drones, которые делятся наработками и обмениваются опытом (diydrones.com);
- используется открытый протокол MAVLink (Micro Air Vehicle Communication Protocol, <http://qgroundcontrol.org/mavlink/start>);
- реализовано управление бесплотниками с разной кинематикой: несколько видов коптеров, вертолет, самолет, автомобиль, что положительно сказалось на архитектуре ПО: код представляет из себя несколько программ управления для разной кинематики и общую библиотеку алгоритмов;
- в алгоритмах доступно множество полезных решений для создания собственного автопилота: от драйверов устройств (датчики, исполнительные механизмы, телеметрия и т.п.) до математики (ПИД, фильтры Калмана и т.д.).

Был приобретен комплект оборудования APM 2.5 и поставлено несколько экспериментов программированию автопилота. Выяснилось, что хотя код написан довольно качественно, документирован он слабо и отлаживать на микроконтроллере сложную математику весьма трудоемко. Также оказалось, что решение на базе микроконтроллеров не подходит для задач управления с нашими требованиями по позиционированию. Для достижения автономности на борту должен быть 32-разрядный компьютер с тактовой частотой в несколько сотен мегагерц, чтобы вести расчеты позиции по алгоритму RTK. На такой компьютер можно установить полноценную ОС, которая упростит остальные эксперименты. При этом от микроконтроллера для управления вводом/выводом нельзя отказываться, так как есть перечень задач, которые оптимально выполнять именно на нем. Сейчас рассматривается возможность дополнить бортовую вычислительную систему платой микрокомпьютера под управлением Linux. Выбор будет сделан из двух вариантов: Raspberry Pi и Beagle Bone Black. Скорее всего, выбор будет остановлен на Raspberry Pi, так как у этой платы проще подсистема питания, что упрощает стыковку со штатным аккумулятором робота.

Результаты экспериментов доступны на Bitbucket:
<https://bitbucket.org/pondskaterteam/pondskater>