

**РАЗРАБОТКА СИСТЕМЫ УПРАВЛЕНИЯ  
МАЛОРАЗМЕРНЫМ ВЕРТОЛЕТОМ****В.О. Клебан, А.А. Шалыто**

Приводится пример построения системы автоматического управления беспилотным малоразмерным вертолетом с соосной схемой расположения винтов. В данной работе методы теории управления и цифровой обработки сигналов успешно дополнены парадигмой автоматного программирования, которая позволяет эффективно строить системы, обладающие сложным поведением.

**Ключевые слова:** имитационное моделирование, автоматное программирование, соосный вертолет.

**Введение**

В настоящее время ряд исследовательских групп изучает вопросы построения систем управления для беспилотных летательных аппаратов, действующих внутри помещений [1–4]. При этом можно выделить три основных направления исследований: аэродинамика летающих роботов, разработка алгоритмов планирования и управления полетом, разработка сенсорики для оценки состояния окружающего пространства и обеспечения безопасности полета в стесненных условиях. Например, в работе [5] представлена реальная система управления легким вертолетом, реализованная на находящейся в продаже серийной модели легкого вертолета.

Многие из проводимых исследований направлены на создание летающих роботов для спасательных операций, где одной из задач является обеспечение автономности и применение любых внешних ориентиров, таких как видеокамеры, невозможно. Разработка летающих моделей связана с тем, что наземные роботы медлительны, а их способность преодолевать пересеченную местность сильно ограничена. В отличие от наземного робота, летающий робот является практически идеальным инструментом для оценки обстановки на местности. В связи со спецификой решаемых задач такой робот должен иметь возможность совершать полет в закрытом помещении для обеспечения возможности сбора информации внутри зданий и сооружений.

**Устройство микровертолета соосной схемы**

В целях исследования динамики соосного микровертолета и разработки системы управления был приобретен соосный радиоуправляемый вертолет фирмы Walkera.

Управление вертолетом осуществляется по четырем каналам. Вертикальное движение осуществляется при помощи оборотов двигателя, управление курсом происходит при помощи рыскания, горизонтальный полет регулируется при помощи крена и тангажа. В силу того, что управление по высоте возможно лишь при помощи изменения оборотов двигателей, вертолет имеет достаточно медленный отклик по каналу вертикальной скорости.

Два несущих винта продуцируют все основные силы и моменты, действующие на вертолет. Управление верхним винтом производится при помощи аппарата Хиллера, который замедляет его реакцию на резкие изменения крена и тангажа за счет гироскопического момента, создаваемого стабилизатором. Нижний винт управляется посредством двух микросервомоторов, которые соединены с ним при помощи тарелки перекося. Скорость вращения винтов контролируется двумя бесколлекторными электродвигателями. Приобретенный вертолет спроектирован специально для полетов внутри помещения, что привело к ослаблению некоторых характеристик, не требующихся для подобных полетов.

Силовая установка и имеющаяся система радиосвязи приобретенного вертолета подверглись существенной модернизации. Также на вертолет были установлены бортовая ЭВМ и твердотельная курсовертикаль собственной разработки.

Модернизация силовой установки включала в себя замену коллекторных двигателей постоянного тока на бесколлекторные двигатели повышенной мощности. В дополнение к двигателям были установлены усилительно-преобразовательные устройства для управления двигателями при помощи широтно-импульсной модуляции и карманы для перехвата воздушного потока с целью охлаждения электродвигателей. Законцовки лопастей были утяжелены для снижения конусности несущего винта и позволили избавиться от флаттера, возникающего при больших оборотах двигателя.

**Устройство бортовой ЭВМ**

Бортовая ЭВМ (рис. 1) представляет собой контроллер на базе архитектуры ARM7, снабженный тремя датчиками угловых скоростей, акселерометром, магнитометром, а также входами для подключения датчиков высотомера и дальномеров.

На базе данной ЭВМ была разработана курсовертикаль, показывающая ориентацию вертолета в пространстве и вычисляющая углы Эйлера в режиме реального времени.

Основными проблемами при разработке курсовертикали оказались [1, 6, 7]: температурный дрейф и «уход нулей» показаний датчиков угловых скоростей, что приводило к линейно возрастающей угловой

ошибке; ошибки калибровки, приводящие к росту угловых ошибок в зависимости от количества движений вертикали; собственный шум датчика и шум датчика вследствие вибраций корпуса вертолета, приводящие к росту угловой ошибки.

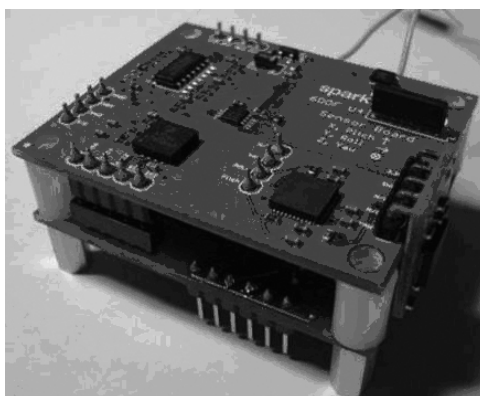


Рис. 1. Бортовая ЭВМ вертолета

В целях обеспечения достоверности показаний прибора производилась его тарировка путем многократных поворотов устройства вдоль различных осей и при различных температурах на специальном стенде. Это позволило добиться высокой стабильности и достоверности показаний курсовертикали.

Для решения задачи фильтрации показаний приборов была разработана схема фильтрации сигналов (рис. 2) позволяющая в значительной степени снизить негативные эффекты от попадания помех в систему [8, 9]. Отметим, что борьба с помехой производилась не только на уровне электронных схем, но и при компоновке вертолета.

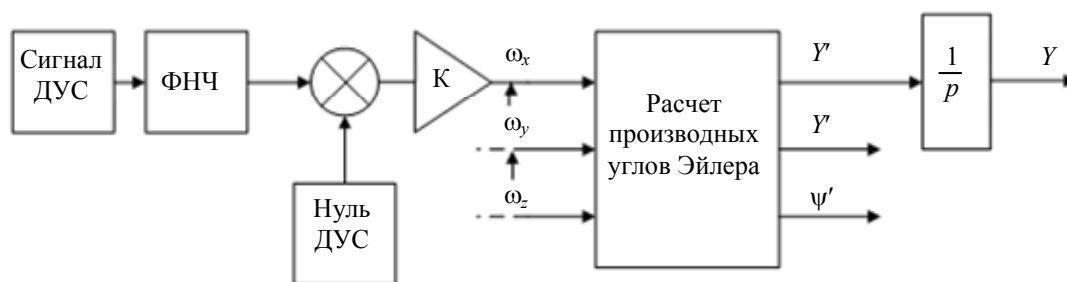


Рис. 2. Схема измерения угла по одному из каналов ДУС

К сожалению, избавиться от эффекта «ухода нуля» в предлагаемой схеме фильтрации не удалось, поэтому был разработан алгоритм, позволяющий производить оценку нулей датчиков угловых скоростей непосредственно в полете (рис. 3).

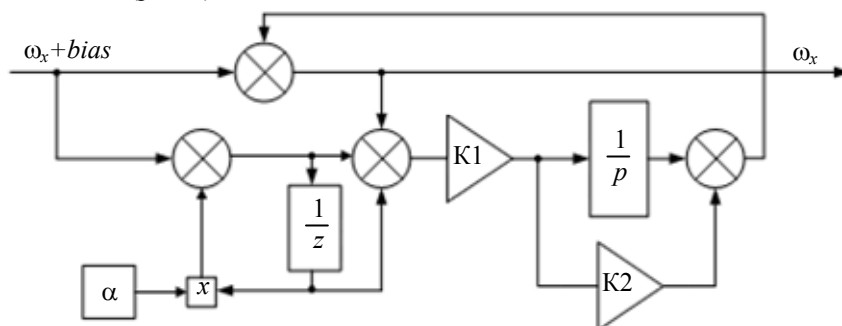


Рис. 3. Схема оценки нуля датчика угловой скорости в полете

Сбор и выдача данных происходят с частотой 200 Гц.

#### Устройство системы связи

Система связи микровертолета представляет собой цифровую радиолинию, работающую на частоте 2 ГГц. Скорость передачи данных составляет 115200 бит/с. Дальность радиосвязи составляет около полутора километров.

К сожалению, имеющиеся средства организации радиосвязи не удовлетворяют условиям разрабатываемой системы. Исследовательский характер проводимых работ делает использование пакетной радиосвязи неудобным. Это проявляется в том, что частые изменения состава передаваемых данных вынуждают каждый раз вносить изменения в состав пакетов. Кроме того, стояла задача реализовать такой протокол передачи данных, который бы не блокировал канал в случае неудавшейся передачи данных, что часто происходит при реализации протоколов, заголовок которых содержит объем принимаемых данных.

Для решения данной задачи было предложено пойти по пути расширения языковых возможностей языка С путем введения нового ключевого слова linked, означающего, что переменная будет синхронизирована между бортовой ЭВМ и наземным компьютером. Например, инструкция static linked int throttle = 0; означает, что целочисленная переменная с именем throttle и стартовым значением 0 будет доступна как на бортовой ЭВМ, так и на наземной, причем ее изменения будут производиться синхронно на обеих машинах. Все переменные, объявленные с модификатором linked, могут контролироваться из менеджера переменных, позволяющего производить изменение переменных вручную, а также наблюдать за их изменением в графическом виде. На программную реализацию данной идеи было получено авторское свидетельство № 2009615456 «Средство обеспечения взаимодействия автоматных программ».

**Устройство системы обнаружения препятствий**

Система обнаружения препятствий представляет собой набор инфракрасных датчиков расстояния [10], расставленных по периметру вертолета, и позволяет осуществлять контроль приближения вертолета к опасным объектам.

**Применение автоматного программирования**

Практически с самого начала экспериментов стало ясно, что объект обладает сложным поведением как с точки зрения логики согласования всех систем, так и с точки зрения непосредственно управления полетом. В связи с этим, для реализации системы управления решено было применить автоматное программирование, эффективность применения которого для реактивных систем и мобильных роботов показана в работах [11–18]. Технология автоматного программирования заключается в представлении программ в виде системы взаимодействующих автоматов. Взаимодействие автоматов может быть реализовано за счет вложенности, вызываемости, обмена сообщениями, работы над общим входом и т.д. Использование автоматного подхода при создании подобных систем управления обладает рядом достоинств:

- документуемость;
- возможность построения компактных моделей Крипке, а значит, верифицируемость;
- возможность осуществления формального и изоморфного преобразования графа переходов в код на языке программирования.

Основной цикл системы управления вертолетом (рис. 4) описывается автоматом из десяти состояний.

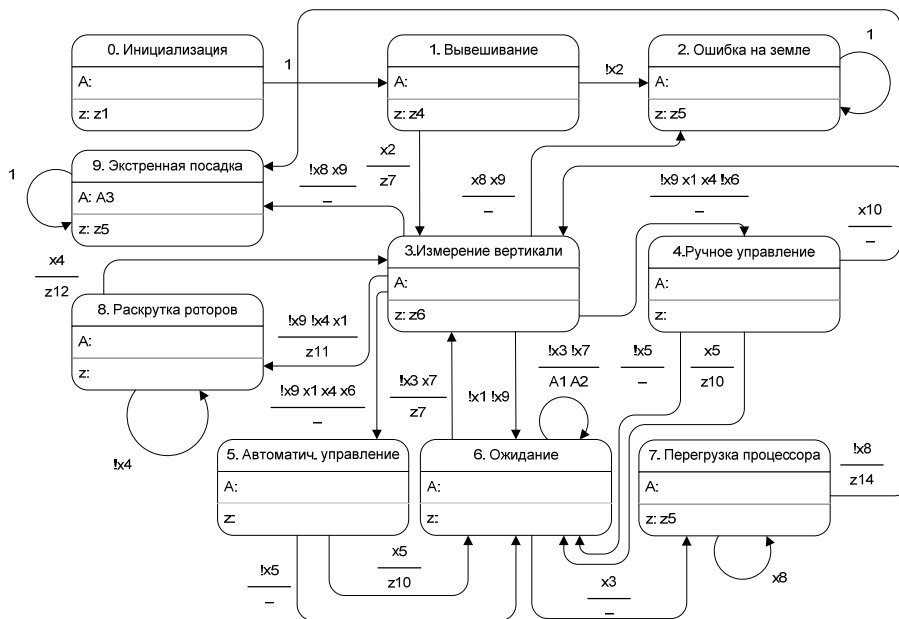


Рис. 4. Главный автомат управления вертолетом

На вход автомату передаются следующие переменные:

- x1 – запуск разрешен;
- x2 – успешное вывешивание;
- x3 – таймер истек до входа в Ожидание;
- x4 – ротор раскручен;
- x5 – такт управления;
- x6 – режим управления «Ручной»;
- x7 – истек таймер;
- x8 – нахожусь на земле;
- x9 – слишком низкая частота работы САУ;
- x10 – установлена связь с землей.

В качестве выхода автомат использует следующие воздействия:

- z1 – инициализация;
- z4 – вывешивание;
- z5 – сообщить об ошибке;
- z6 – фильтрация данных;
- z7 – сброс таймера;
- z9 – останов;
- z10 – вывод управления в канал ШИМ;
- z11 – раскрутка ротора;
- z12 – сброс курсовертикали;
- z14 – снижение частоты работы САУ;
- z15 – повышение частоты работы САУ.

Аналогично при помощи автоматов описываются все остальные системы вертолета: опрос датчиков, управление полетом, управление радиосвязью и т.д.

### **Заключение**

Динамика полета вертолета – сложный нелинейный процесс с неучтенной динамикой. К сожалению, построение полной физической модели вертолета на данном этапе недоступно в силу ее сложности, наличия массы специфичных для каждой конкретной модели вертолета (или даже экземпляра) моментов, наличия неучтенной динамики и т.д.

Изложенный автоматный подход, позволяющий проводить имитационное моделирование управления вертолетом, обеспечил возможность создания системы управления без использования модели объекта управления. В данном случае методы теории управления и цифровой обработки сигналов успешно дополнены парадигмой автоматного программирования, которая позволяет эффективно строить системы, обладающие сложным поведением, а также сильносвязанные системы.

Дальнейшие исследования в этой области требуют значительно больших ресурсов. Вот, что, например, пишет А. Гоголь, ректор СПбГУ телекоммуникаций им. М.А. Бонч-Бруевича: «В одном из университетов США я наблюдал за продвижением группы из семи человек, занимающихся геликоптером. Бюджет – миллион долларов на каждого в год, причем без особых отчетов. Не знаю, на каком этапе они сейчас, но тогда их модели спокойно летали брюхом вверх. Я такого в жизни не видел».

Также вертолетами занималось и занимается группа из 15 человек в Цюрихском политехническом институте [19], и это, как отмечено выше, не единственный университет в мире.

Исследование выполнено по Федеральной целевой программе «Научные и научно-педагогические кадры инновационной России на 2009–2013 годы» в рамках государственного контракта П1188 от 27 августа 2009 года.

### **Литература**

1. Castillo P., Lozano R., Dzul A.E. Modelling and Control of Mini-Flying Machines . –Springer, 2005.
2. Watkinson J. The Art of Helicopter. – Elsevier, 2004.
3. Bermes C., Sartori K., Schafroth D., Bouabdallah S., Siegwart R. Control of a Coaxial Helicopter with Center of Gravity Steering // Proceedings of Intl. Conf. on SIMULATION, MODELING and PROGRAMMING for AUTONOMOUS ROBOTS. – 2008. – P. 492–500.
4. Kotmann M. Software for Model Helicopter Flight Control. – Zurich: Institute of Computer Systems, 1999.
5. Барабанов А.Е., Ромаев Д.В. Автопилот вертолета по телевизионным наблюдениям // Гироскопия и навигация. – 2006. – № 4. – С. 106–107.
6. Локк А.С. Управление снарядами. – М.: Физматлит, 1958.
7. Горбатенко С.А., Макашов Э.М. Механика полета. – М.: Машиностроение, 1969.
8. Лайонс Р. Цифровая обработка сигналов. – М.: Бином, 2007.

9. Айфичер Э., Джервис Б. Цифровая обработка сигналов: практический подход. – М.: Вильямс, 2008.
10. Sharp distance sensors [Электронный ресурс]. – Режим доступа: <http://www.parallax.com/dl/docs/prod/acc/SharpGP2D12Snrs.pdf>, свободный. Яз. англ. (дата обращения 09.02.2011).
11. Туккель Н.И., Шалыто А.А. Система управления дизель-генератором (фрагмент). Программирование с явным выделением состояний. Проектная документация [Электронный ресурс]. – Режим доступа: <http://is.ifmo.ru/projects/dg/>, свободный. Яз. рус. (дата обращения 09.02.2011).
12. Туккель Н.И., Шалыто А.А. От тьюрингова программирования к автоматному // Мир ПК. – 2002. № 2. – С. 144–149 [Электронный ресурс]. – Режим доступа: <http://is.ifmo.ru/works/turing/>, свободный. Яз. рус. (дата обращения 09.02.2011).
13. Шалыто А.А. Switch-технология. Алгоритмизация и программирование задач логического управления. – СПб: Наука, 1998. – 628 с.
14. Шалыто А.А. Логическое управление. Методы аппаратной и программной реализации. – СПб: Наука, 2000. – 780 с.
15. Шопырин Д.Г., Шалыто А.А. Синхронное программирование // Информационно-управляющие системы. – 2004. – № 3. – С. 35–42 [Электронный ресурс]. – Режим доступа: [http://is.ifmo.ru/works/sync\\_prog/](http://is.ifmo.ru/works/sync_prog/), свободный. Яз. рус. (дата обращения 09.02.2011).
16. Клебан В.О., Шалыто А.А. Использование автоматного программирования для построения многоуровневых систем управления мобильными роботами // Сборник тезисов 19 Всероссийской научно-технической конференции «Экстремальная робототехника». – СПб: ЦНИИ РТК, 2008. – С. 85–87.
17. Brooks R.A. A Robust Layered Control System for a Mobile Robot // IEEE Journal of Robotics and Automation. – 1986. – V. 2. – P. 14–23.
18. Harel D., Pnueli A. On the development of reactive systems / In «Logic and Models of Concurrent Systems». – NATO Advanced Study Institute on Logic and Models for Verification and Specification of Concurrent Systems. – Springer Verlag, 1985. – P. 477–498.
19. Flying Machine Arena [Электронный ресурс]. – Режим доступа: [http://www.idsc.ethz.ch/Research\\_DAndrea/FMA](http://www.idsc.ethz.ch/Research_DAndrea/FMA), свободный. Яз. англ. (дата обращения 09.02.2011).

*Клебан Виталий Олегович*

– Санкт-Петербургский государственный университет информационных технологий, механики и оптики, аспирант, [vk.developer@gmail.com](mailto:vk.developer@gmail.com)

*Шалыто Анатолий Абрамович*

– Санкт-Петербургский государственный университет информационных технологий, механики и оптики, доктор технических наук, профессор, зав. кафедрой, [shalyto@mail.ifmo.ru](mailto:shalyto@mail.ifmo.ru)

**УДК 004.832.28**

## **ПРИМЕНЕНИЕ ДВУХЭТАПНОГО ГЕНЕТИЧЕСКОГО ПРОГРАММИРОВАНИЯ ДЛЯ ПОСТРОЕНИЯ МОДЕЛИ ТАНКА В ИГРЕ «ROBOCODE» Д.О. Соколов**

Рассматривается применение генетического программирования для построения конечных автоматов, управляющих системами со сложным поведением. Приведен метод двухэтапного генетического алгоритма, основанный на идеях динамического программирования. Рассмотрено применение этого метода на примере игры «Robocode».

**Ключевые слова:** автомат, деревья разбора, генетическое программирование.

### **Введение**

Задача построения управляющих систем для беспилотной техники в настоящее время является актуальной. Чаще всего эта задача решается вручную. Такой подход не всегда является эффективным как ввиду больших затрат ресурсов, так и в связи с низким качеством построенных систем. В некоторых случаях ввиду сложности системы построение управляющей системы при помощи ручного труда невозможно. Естественная идея – поручить построение управляющей системы компьютеру.

Эволюционные вычисления успешно применяются для автоматического создания программ [1]. Эффективность применения эволюционных алгоритмов напрямую зависит от способа представления программы в виде хромосомы [2]. Для многих управляющих систем их поведение удобно представлять в виде конечных автоматов [3, 4]. Одной из таких систем является система управления танком в игре «Robocode», которая и рассматривается в настоящей работе.

Выделим основные проблемы, которые решаются в данной работе, возникающие при использовании генетического программирования для решения задачи управления:

- большое время работы генетического алгоритма;
- необходимость обработки вещественных переменных при помощи конечных автоматов;
- попадание в локальные максимумы.