

Оценка нагруженности протоколов обмена данными для БПЛА на базе интеллектуальных компонентов

С. Г. Черный^{1,2} ✉, Н. В. Шипаренко², М. В. Чупаков³

¹ Государственный университет морского и речного флота имени адмирала С.О. Макарова, ул. Двинская, д. 5/7, г. Санкт-Петербург 198035, Российская Федерация

² Керченский государственный морской технологический университет, ул. Орджоникидзе, д. 82, г. Керчь 298309, Республика Крым, Российская Федерация

³ Черноморское высшее военно-морское училище имени П.С. Нахимова, ул. Дыбенко, д. 1а, г. Севастополь 299028, Российская Федерация

✉ e-mail: sergiiblack@gmail.com

Резюме

Цель работы заключается в изучении работы протоколов на различных уровнях взаимодействия, предложить варианты улучшения безопасности взаимодействия на основе внедрения интеллектуальных компонентов. В последние годы значительно вырос интерес к использованию дронов в различных сферах. Причины связаны с непрерывным ростом технологий, особенно с появлением быстрых микропроцессоров, которые обеспечивают автономное управление несколькими системами. Фотография, строительство, контроль и наблюдение – лишь некоторые из областей, в которых использование дронов становится обычным явлением.

Методы. В статье рассматриваются протоколы, задействованные в работе БПЛА на разных уровнях, их особенности, преимущества и недостатки, а также их нагруженность и обеспечение безопасности. Использование реалистичных технологических особенностей беспилотных летательных аппаратов для тестирования моделей и методов может оказаться весьма актуальным для практических целей в различных отраслях от гражданских до военных.

Задачами исследования являются детализация протоколов обмена информации в БПЛА по различным уровням, с учетом анализа структуры передаваемой информации.

Результаты. Рациональное использование энергии имеет решающее значение для осуществления эффективного и безопасного процесса работы БПЛА. Перебои в энергоснабжении могут нанести серьезный ущерб району полета. Затронуты и рассмотрены вопросы корреляции загрузки протокола связанных с потреблением энергии и стратегиями зарядки.

Заключение. Исследование изучает различные протоколы обмена данными, которые используются в системах БПЛА, а также анализирует их производительность. Для качественной оценки влияния на безопасность информации, предлагается ввести интеллектуальные компоненты, которые динамически адаптируют протоколы обмена данными на основе анализа угроз в режиме работы в реальном времени и возможностей системы.

Ключевые слова: БПЛА; протокол; уязвимость; реальное время; угроза; информационная безопасность; загруженность протоколов; реальное время.

Конфликт интересов: Авторы декларируют отсутствие явных и потенциальных конфликтов интересов, связанных с публикацией настоящей статьи.

Для цитирования: Черный С. Г., Шипаренко Н., Чупаков М. Оценка нагруженности протоколов обмена данными для БПЛА на базе интеллектуальных компонентов // Известия Юго-Западного государственного университета. 2023; 27(3): 128-151. <https://doi.org/10.21869/2223-1560-2023-27-3-128-151>.

Поступила в редакцию 10.08.2023

Подписана в печать 02.09.2023

Опубликована 28.10.2023

Assessment of Data Exchange Protocol Workload in Uavs on Intelligent Components

Sergey G. Cherny ^{1,2} ✉, Nikita V. Shiparenko ², Maksim V. Chupakov ³

¹ State University of Marine and River Fleet named after admiral S.O. Makarov
5/7 Dvinskaya str., St. Petersburg 198035, Russian Federation

² Kerch State Marine Technological University
82 Ordzhonikidze str., Kerch, 298309, Republic of Crimea, Russian Federation

³ Black Sea Higher Naval School named after P.S. Nakhimov,
1a, Dybenko str., Sevastopol 299028, Russian Federation

✉ e-mail: sergiiblack@gmail.com

Abstract

Purpose of research. In recent years, interest in the use of drones in various fields has grown significantly.

Methods. The reasons are due to the continuous growth of technology, especially the advent of fast microprocessors that enable autonomous control of multiple systems. Photography, construction, inspection and surveillance are just some of the areas in which the use of drones is becoming commonplace. The purpose of the work is to study the operation of protocols at various levels of interaction, to propose options for improving the security of interaction based on the introduction of intelligent components. The article discusses the protocols involved in the operation of UAVs at different levels, their features, advantages and disadvantages, as well as their load and security. Using realistic technological features of unmanned aerial vehicles to test models and methods can be very relevant for practical purposes in various industries from civil to military.

The objectives of the study are to detail the protocols for information exchange in UAVs at various levels, taking into account the analysis of the structure of the transmitted information.

Results. The study examines various communication protocols used in UAV systems and also analyzes their performance. To qualitatively assess the impact on information security, it is proposed to introduce intelligent components that dynamically adapt data exchange protocols based on real-time threat analysis and system capabilities.

Conclusion. Efficient use of energy is critical to efficient and safe UAV operation. Power outages can cause serious damage to the flight area. The issues of correlation of protocol load associated with energy consumption and charging strategies are touched upon and considered.

Keywords: UAV; protocol; vulnerability; real time; threat; information security; protocol load; real time.

Conflict of interest. The authors declare the absence of obvious and potential conflicts of interest related to the publication of this article.

For citation: Cherny S. G., Shiparenko N., Chupakov M. Assessment of Data Exchange Protocol Workload in Uavs on Intelligent Components. *Izvestiya Yugo-Zapadnogo gosudarstvennogo universiteta = Proceedings of the Southwest State University*. 2023; 27(3): 128-151 (In Russ.). <https://doi.org/10.21869/2223-1560-2023-27-3-128-151>.

Received 10.08.2023

Accepted 02.08.2023

Published 28.10.2023

Введение

В наше время беспилотные летательные аппараты (БПЛА) стали неотъемлемыми инструментами в самых различных областях, начиная от гражданских задач – визуальное наблюдение при помощи в случае ЧП, элементами транспортной логистики, заканчивая военной разведкой и участием в полноценных военных конфликтах.

БПЛА зачастую маневренны, малогабаритны и могут быть управляемы с больших расстояний. Самой логичной боевой задачей стали сбор и обработка чувствительной визуальной и параметрической информации во время выполнения заданий, что в свою очередь требует обеспечения быстродействия и должного уровня безопасности нагруженных протоколов обмена данными в БПЛА с акцентом на повышение безопасности информации с использованием интеллектуальных компонентов [1-3].

В ходе реализации работы использованы самодельные основы БПЛА в качестве материнских плат на основе Raspberry Pi 3 и STMicroelectronics STM32F407 Discovery Board на базе ARM Cortex M4 процессора (STM32F407VGT6). В качестве оболочки для самодельного дрона использовался Ardupilot. Для анализа протоколов передачи информации низкого (логического) уровня использовался осциллограф Hantek DSO-6022BE и логический анализатор ChipDipDac

RDC2-0064. При работе с высокими скоростями передачи данных или большим объемом данных, возникают некоторые проблемы с записью и декодированием в реальном времени, декодирование зашифрованных каналов связи также занимает время и вычислительные ресурсы, поэтому в своей работе будет упор на работу логического анализатора.

1. Отображение записанных выборок:

- Захваченные данные отображаются на экране в виде временных диаграмм или графиков.

- Во время отображения данных, анализатор может обнаруживать и декодировать протоколы передачи данных, такие как UART, SPI, I2C, CAN и другие. Это позволяет пользователю анализировать передаваемые данные и последовательность действий между устройствами.

2. Анализ выборок в специализированном ПО (PulseView):

- PulseView — программное обеспечение с открытым исходным кодом для анализа данных, полученных от логических анализаторов. Оно позволяет визуализировать данные, анализировать протоколы и искать аномалии в сигналах.

- Непосредственно в PulseView можно настраивать параметры анализа, фильтровать данные, синхронизировать различные каналы сигналов и производить другие операции для более детального анализа данных.

- Можно визуально анализировать захваченный трафик, проверять целостность информации, выводить содержимое сообщений захваченных данных, декодировать сообщения протоколов, используя фильтры.

Материалы и методы

Протоколы обмена данными на физическом уровне в БПЛА

В БПЛА различных производителей зачастую используются следующие основные протоколы на логическом уровне:

SPI (Serial Peripheral Interface): синхронный последовательный интер-

фейс, который был разработан для обмена данными между микроконтроллерами и периферийными устройствами, такими, как сенсоры, датчики, дисплеи и другие. Он работает в полудуплексном режиме. SPI не предоставляет встроенных механизмов безопасности, поскольку он только обеспечивает обмен данными между устройствами и не имеет методов шифрования или аутентификации, но обладает высокой скоростью передачи данных и широким применением во многих устройствах, где требуется высокая производительность и простота обмена данными (рис.1).

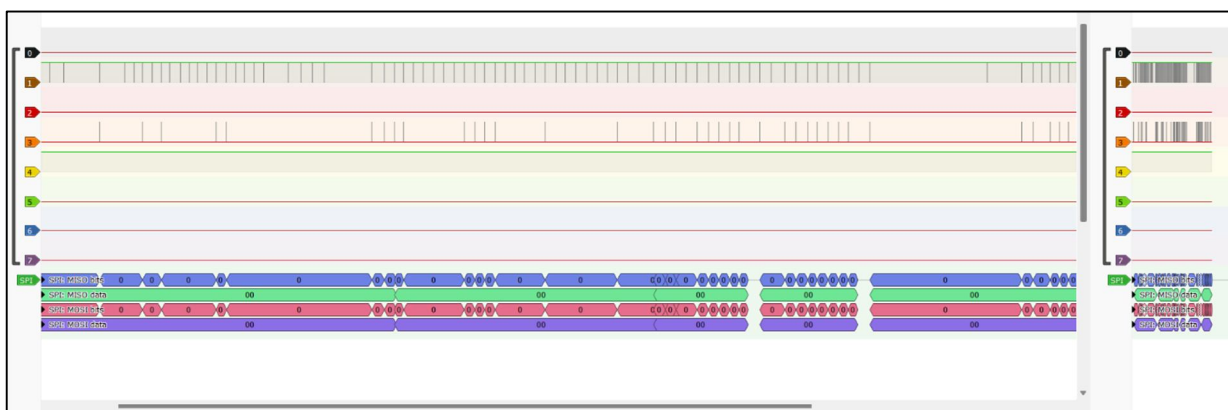


Рис. 1. Захваченная выборка логическим анализатором длиной в 3 секунды, декодированная как содержащая обмен данными по шине SPI

Fig. 1. Logic analyzer captured sample of 3 seconds, decoded as SPI data exchange

Архитектурно, SPI устроен так, что использует 4 сигнальные линии:

SCLK (синхронный тактовый сигнал);
MOSI (Master Out Slave In);
MISO (Master In Slave Out);
SS/CS (Slave Select \ Chip Select).

Сообщение в протоколе SPI представляет собой последовательность бит, которые передаются по линиям MOSI (Master Out Slave In) и MISO (Master In Slave Out).

В общем случае, протокол SPI передает данные по линии MOSI и MISO синхронно с тактовым сигналом SCLK (Serial Clock). Каждый бит данных передается на каждом тактовом переходе.

Также следует отметить, что необходимо согласовать настройки передачи данных между slave и master, полярность сигнала SCLK, так как в ином случае правильное чтение и запись информации могут быть нарушены.

I²C (Inter-Integrated Circuit): является синхронным последовательным интерфейсом, который используется для связи между микроконтроллерами и различными периферийными устройствами, такими как датчики, память, дисплеи, а также для обмена данными между несколькими микроконтроллерами.

В отличие от SPI, *I²C* использует всего 2 сигнальные линии – SDA (Serial Data) и SCL (Serial Clock), но при этом позволяет на высокой скорости, работая в полудуплексном режиме, подключать множество устройств на одной шине (до 128).

Немаловажной особенностью для использования в БПЛА является энергосбережение в работе протокола датчиками. Сообщение в протоколе *I²C* (Inter-Integrated Circuit) представляет собой последовательность бит, передаваемых последовательно по двум линиям – линия данных (SDA) и линия тактирования сигнала (SCL). Протокол *I²C* работает в синхронном режиме, где тактовый сигнал SCL используется для синхронизации передачи данных между устройствами. Выборка снята в момент опроса адресов на шине (рис. 2).

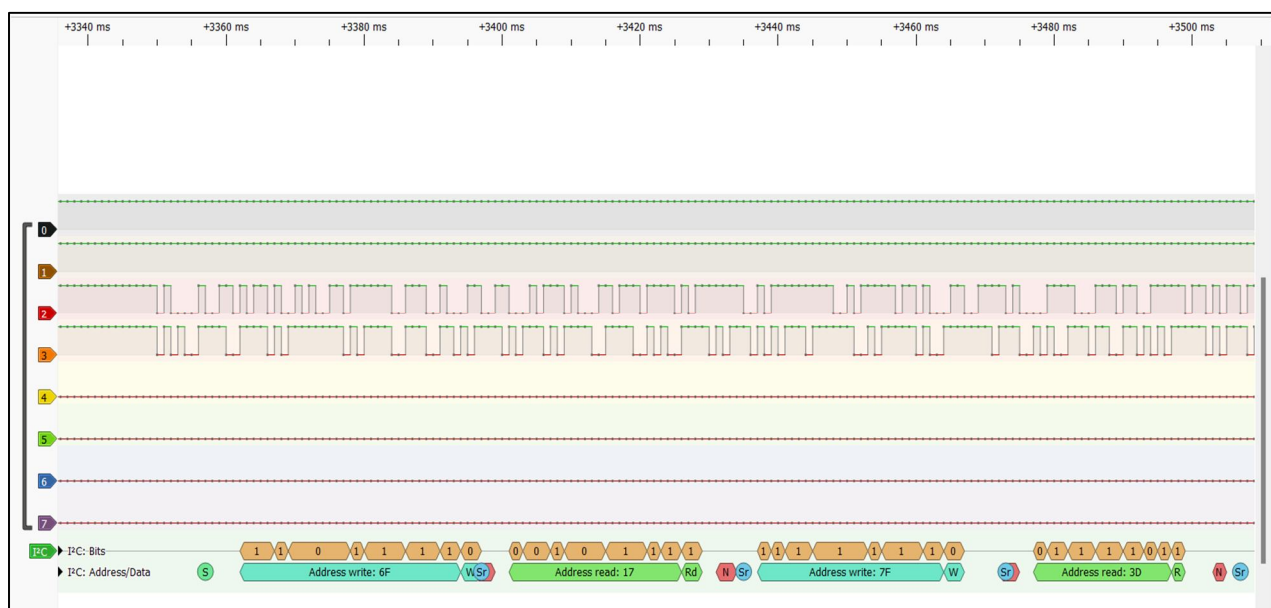


Рис. 2. Захваченная выборка логическим анализатором длиной в 3 секунды, декодированная как содержащая *I²C* сообщения, их содержимое, длина адресов

Fig. 2. Logic analyzer captured sample 3 seconds long, decoded as containing *I²C* (contents, address length)

I²C (Inter-Integrated Circuit) — это протокол обмена данными между микросхемами на коротких расстояниях и поддерживает одновременную работу множества устройств, подключенных к одной шине.

Важно учесть, что *I²C* поддерживает ведущие устройства на линии одновременно, однако в БПЛА стоит придерживаться одного master на линии, чтобы не приводить к конфликтам и коллизиям на шине, или проводить гра-

мотное планирование и управление шиной во избежание ошибок и неправильного чтения \ записи данных [3-5].

UART (Universal Asynchronous Receiver/Transmitter) – асинхронный последовательный интерфейс, который используется для передачи данных между устройствами. Он широко применяется в связке с компьютерами, периферийными устройствами и микроконтроллерами, работая в полудуплексном режиме, он использует 2 сигнальные линии – Tx (сигнал передачи) и Rx (сигнал приёма).

UART, в отличие от вышеизложенных протоколов выше не имеет линии для синхронизации скорости передачи данных, так как он подразумевает, что оба (предельное количество устройств на шине) имеют одинаковые настройки. Реализует механизмы обеспечения безопасности данных, его задача – обеспечить быструю и качественную передачу данных между устройствами. Он прост в настройке и использовании – в БПЛА может использоваться для подключения GPS-модуля, различных радиопередатчиков и с другими периферийными устройствами (рис. 3).

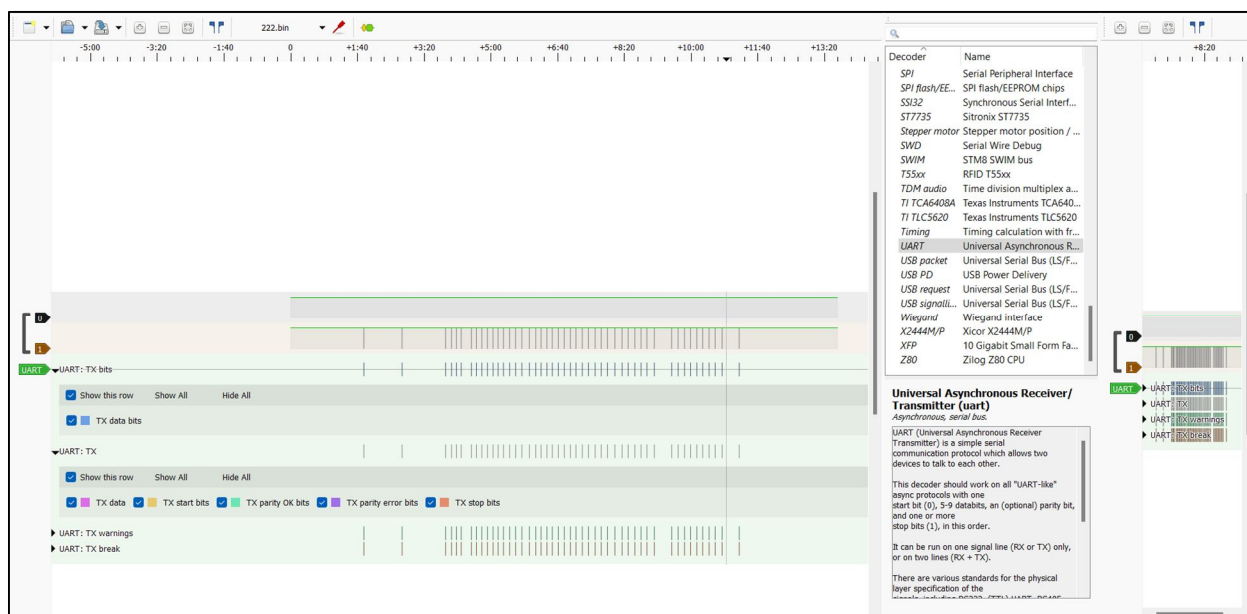


Рис. 3. Захваченная выборка логическим анализатором длиной в 3 секунды, декодированная как содержащая UART сообщения

Fig. 3. Logic analyzer captured 3 second decoded as containing UART messages

Сообщение в протоколе UART представляет собой последовательность бит, передаваемых последовательно по одной линии (обычно линия TX для передачи и RX для приёма – они соединены «крест-накрест» у передающего и принимаемого устройства). Протокол UART работает в

асинхронном режиме, что означает отсутствие отдельной тактовой линии для синхронизации передачи данных. Вместо этого, каждый символ передается с использованием начального бита, нескольких данных бит и опционального бита четности и/или конечного бита.

Jitter (фазовое дрожание цифрового сигнала данных) — это непредсказуемое изменение временных интервалов между символами в передаче данных через UART и\или другой протокол, которое может возникать из-за неточности внутренних часов устройств, электромагнитных помех, а также шумов на линиях связи и других факторов. Джиттер может приводить к ошибкам в при-

еме данных и вызывать некорректную интерпретацию символов [5-7].

Допустим, одни и те же захваченные выборки можно проанализировать, не зная реальной скорости UART, указав свою предполагаемую.

Так данные выглядят при установке скорости 9600 бод (рис. 4).

А так при 115200 бод (рис. 5).



Рис. 4. Снимок выборки, декодированной как UART на скорости 9600 бод

Fig. 4. Snapshot of sample decoded as UART at 9600 baud

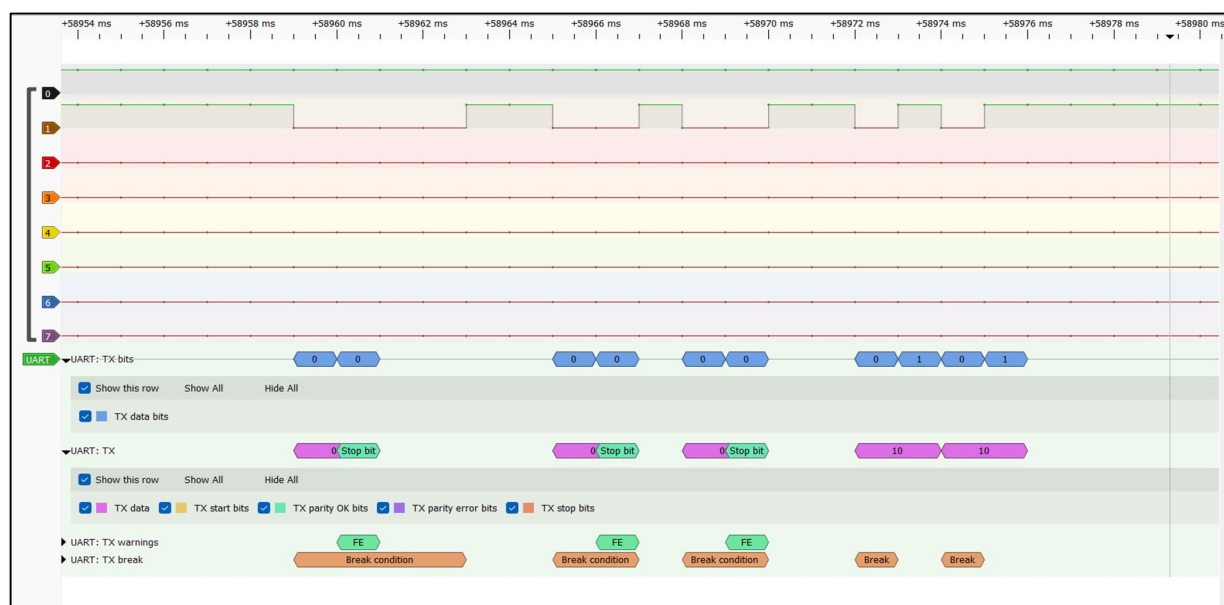


Рис. 5. Снимок выборки, декодированной как UART на скорости 115200 бод

Fig. 5. Snapshot of sample decoded as UART at 115200 baud

Джиттер может также вызывать искажения Frame (пакета), он зависит от точности и стабильности работы системы. В БПЛА же, зачастую посредством электромагнитных помех и воздействий температуры могут создаваться потеря и искажения сообщений. Это можно избежать, используя специальные микросхемы, которые обеспечивают стабильность часов (точное внешнее тактирование с кварцевого резонатора или генератора), выполнять коррекцию уже существующих ошибок (ECC).

Говоря про разные платформы для производства БПЛА, стоит упомянуть, что в них может быть различное количество интерфейсов – в зависимости от центрального процессора, сопроцессоров или другой встроенной логики и периферии [3, 8].

Применимо к Raspberri Pi 3 и дрону, основанному на микроконтроллере STM32F407VGT6, можно составить сравнительную табл. 1.

Таблица 1. Сравнительная характеристика поддержки UART

Table 1. Comparative characteristics of UART support

Характеристика / Characteristic	Raspberry Pi 3	STM32F415VGT6
Количество UART	1 встроенный UART (UART0)	Несколько встроенных UART портов
Скорость передачи данных	Различные скорости до 921600 bps	Различные скорости до нескольких мегабит в секунду
Биты данных	8	8 или 9 (по выбору)
Биты контроля четности	Отсутствует	Можно выбрать или отключить
Стоп-биты	1 или 2	1, 1.5 или 2 (по выбору)

Стоит обратить внимание, что Raspberri Pi 3 имеет только один встроенный UART порт (UART0), в то время как STM32F407VGT6 обладает несколькими встроенными UART портами [4, 6]. Кроме того, STM32F407VGT6 предоставляет больше гибкости в выборе количества бит данных (8 или 9) и стоп-битов (1, 1.5 или 2) (рис. 6).

Джиттер также может вызывать смещение временных интервалов в выборке логического анализатора между символами, что приводит к неправиль-

ному распознаванию символов. Могут искажённо интерпретироваться стартовый бит, биты данных и стоп-бит, что обязательно приведёт к ошибкам в декодировании передаваемой информации. В случае, если джиттер слишком велик, логический анализатор может потерять символы, что приведет к потере части или всего сообщения. Потеря данных в джиттере особенно опасна в случае передачи критически важной информации, такой как управляющие команды или показания датчиков, а это,

в свою очередь, может привести к неправильным действиям или потере ценных данных для управления БПЛА.

Результаты и их обсуждение

Ошибки интерпретирования стартового и стоп-битов приведут к неправильному определению границы символов и протокольных пакетов, а это, в свою очередь, помешает декодированию всего сообщения [5,7-9].

Ardupilot — открытое программное обеспечение (распространяемое по лицензии «Open Source» — открытые (бесплатные и опубликованные) исходные коды для некоммерческого использования) для автономного управления БПЛА и другими робототехническими

платформами. Он предоставляет набор функций и алгоритмов для управления полётом и автономного перемещения, что делает его популярным выбором для создания автономных и ручных систем БПЛА.

Однако, чтобы собрать собственный дрон недостаточно просто установить программный комплекс или выполнить заливку прошивки на микроконтроллер, нужны и другие компоненты:

Интерфейс Ardupilot в браузере обычно реализуется через веб-сервер, который работает на борту БПЛА или наземной станции и предоставляет доступ к управлению и настройке БПЛА через веб-браузер на устройствах, поддерживающих интернет-браузеры [6-8].

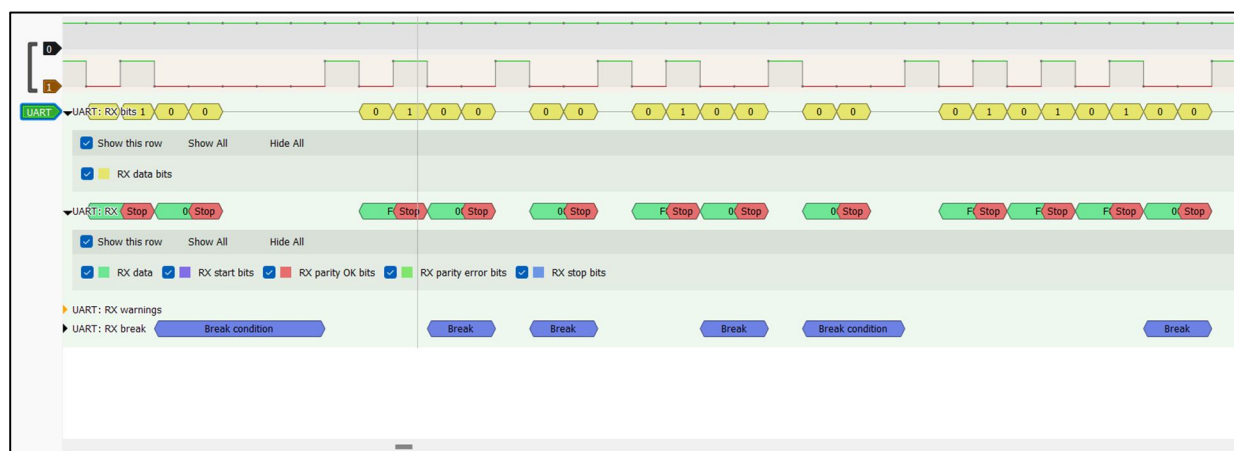


Рис. 6. Снимок выборки, декодированной как UART на скорости 9600 бод с битовым содержанием сообщения

Fig. 6. Snapshot of the sample decoded as UART at 9600 baud with message bit content

Информационные протоколы

MAVLink (Micro Air Vehicle Communication Protocol) – это протокол обмена сообщениями для беспилотных лета-

тельных аппаратов, который разрабатывался для обеспечения связи между автономными системами и внешними устройствами [7-9].

MavLink разрабатывался с учетом ограниченных ресурсов микроконтроллеров, которые используются в автономных системах, что, несомненно, полезно в работе с БПЛА.

MavLink использует сообщения различных типов:

- о состоянии (*HEARTBEAT*): Сообщение с основной информацией о состоянии БПЛА, такой как режим полета, заряд батареи и качества связи;

- о позиции (*GLOBAL_POSITION_INT*): Сообщение, содержащее информацию о координатах положения БПЛА (широта, долгота, высота);

- о команде управления (*COMMAND_LONG*): Сообщение, которое позволяет отправить команды управления БПЛА,

такие как взлет, посадка, изменение скорости, операции маневрирования и другие действия;

- о состоянии батареи (*SYS_STATUS*): Сообщение с информацией о заряде батареи и других состояниях системы.

MavLink является важным протоколом для обеспечения связи между беспилотными летательными аппаратами и внешними устройствами, позволяя передавать и получать данные о состоянии, навигации и управлении, что делает его основой для разработки автономных и управляемых БПЛА. Запись сообщения в битовой нотации (48 бит всего), не считая конечного блока контрольной суммы с учётом содержимого полей и диапазонов данных (таб.2).

Таблица 2. Запись сообщения в битовой нотации (48 бит)

Table 2. Recording a message in bit notation (48 bits)

№	Название (RU) / Name (EN)	Название (EN) / Name (EN)	Размер (бит) / Size (bit)	Значение (по умолчанию) / Value (default)	Диапазон / Range
1	Старт	Start	8	0xFE	Фиксированное значение (0xFE)
2	Длина	Length	8	Обычно равна 9	0-255
3	ID	ID	8	0	0-255
4	Версия	Autopilot type	8	Идентификатор автопилота	0-255
5	Система ID	System ID	8	Идентификатор БПЛА/системы	1-255
6	Статус	Custom mode	8	Статус и режим работы БПЛА	1-255

Это представление не включает контрольную сумму, которая является конечным блоком в записи сообщения

MavLink и используется для проверки целостности данных.

Контрольная сумма в данном протоколе вычисляется при помощи алгоритма XOR (оно же – «исключающее

ИЛИ»), применяемого для всех байт данных сообщения, за исключением поля стартового байта (таб.3).

Таблица 3. Отправленное сообщение Heartbeat с БПЛА в шестнадцатеричной записи

Table 3. The Heartbeat message sent from the UAV in hexadecimal notation

Старт: 0xFE	Длина: 0x09	ID: 0x00	Версия: 0x03	Система ID: 0x01	Статус: 0x01	Контрольная сумма: 0x0C
----------------	----------------	-------------	-----------------	---------------------	-----------------	----------------------------

Контрольная сумма сообщения Heartbeat из табл. 6 вычисляется так:

XOR (0x09, 0x00, 0x03, 0x01, 0x01)

Детально:

Запишем байты в двоичной системе и обозначим их как переменные B1, B2, B3, B4, B5

B1= 0x09 (00001001 в двоичной системе)

B2= 0x00

(00000000 в двоичной системе)

B3= 0x03

(00000011 в двоичной системе)

B4=0x01

(00000001 в двоичной системе)

B5= 0x01 (00000001 в двоичной системе)

Для вычисления XOR ($\oplus$) мы начинаем сравнивать каждый бит в позиции 0 (младший бит) у всех байтов (двигаясь справа налево):

Позиция 0 (младший бит):

$$B1[0] \oplus B2[0] \oplus B3[0] \oplus B4[0] \oplus B5[0] = 1 \oplus 0 \oplus 1 \oplus 1 \oplus 1 = 0$$

Позиция 1:

$$B1[1] \oplus B2[1] \oplus B3[1] \oplus B4[1] \oplus B5[1] = 0 \oplus 0 \oplus 0 \oplus 0 \oplus 0 = 0$$

Позиция 2:

$$B1[2] \oplus B2[2] \oplus B3[2] \oplus B4[2] \oplus B5[2] = 0 \oplus 0 \oplus 0 \oplus 0 \oplus 0 = 0$$

Позиция 3:

$$B1[3] \oplus B2[3] \oplus B3[3] \oplus B4[3] \oplus B5[3] = 0 \oplus 0 \oplus 0 \oplus 0 \oplus 0 = 0$$

Позиция 4:

$$B1[4] \oplus B2[4] \oplus B3[4] \oplus B4[4] \oplus B5[4] = 1 \oplus 0 \oplus 0 \oplus 0 \oplus 0 = 1$$

Позиция 5:

$$B1[5] \oplus B2[5] \oplus B3[5] \oplus B4[5] \oplus B5[5] = 0 \oplus 0 \oplus 0 \oplus 0 \oplus 0 = 0$$

Позиция 6:

$$B1[6] \oplus B2[6] \oplus B3[6] \oplus B4[6] \oplus B5[6] = 0 \oplus 0 \oplus 0 \oplus 0 \oplus 0 = 0$$

Позиция 7:

$$B1[7] \oplus B2[7] \oplus B3[7] \oplus B4[7] \oplus B5[7] = 0 \oplus 0 \oplus 1 \oplus 0 \oplus 0 = 1$$

После вычисления значений в позициях, собираем все биты обратно в байт: 0b00001100 (в двоичной системе) = 0x0C (в шестнадцатеричной).

Таким образом, результат операции XOR на каждой позиции для всех байтов (0x09, 0x00, 0x03, 0x01, 0x01) равен 0x0C – это и есть контрольная сумма.

Если при передаче сообщения произойдёт искажение, то контрольная сумма, вычисленная на стороне приёмника, не совпадёт с переданной в сообщении и такое сообщение будет отброшено, и отправителю передается запрос на повторную передачу [11-13].

MQTT (Message Queuing Telemetry Transport)

MQTT (Message Queuing Telemetry Transport) — это протокол обмена сообщениями, который был разработан для передачи данных между устройствами в условиях ограниченной пропускной способности или ненадежным соединением.

Архитектурно он сильно отличается от Mavlink - MQTT основан на модели публикации или подписки, в которой ведущие устройства могут публиковать сообщения на определенные темы, а ведомые устройства могут подписаться на эти темы и получать сообщения. Такая модель не требует постоянно оставаться на линии и сканировать эфир на наличие новых сообщений [3, 5, 9].

В MQTT доступно несколько уровней QoS:

QoS 0 («At most once» («Максимум одно»)): Брокер сообщений не подтверждает доставку сообщения. Сообщение доставляется только один раз и может быть потеряно.

QoS 1 («At least once» («Хотя бы одно»)): Брокер подтверждает получение сообщения. Если клиент не получает подтверждение от брокера, он повторно отправляет сообщение.

QoS 2 («Exactly once» («Точно одно»)): Брокер подтверждает доставку сообщения. Клиент и брокер взаимодействуют для гарантированной доставки сообщения ровно один раз.

БПЛА через брокера сообщений получает сообщение и высылает в ответ выполняемые инструкции на шину Instructions (взлёт, полёт на координаты Санкт-Петербурга, центр города, посадка):

PUBLISH

Topic: instructions

Data: "Take off, fly to coordinates (59.9343, 30.3351), land"

QoS: 1

Синтаксис команд и названия топиков зависят от версии управляющего ПО, платы контроллера дрона и т.д., важно чтобы они были синхронизированы. Протокол MQTT может использоваться поверх различных физических интерфейсов, таких как UART, SPI и Ethernet.

MQTT, будучи разработанным для работы в мире IoT («Internet of Things» с англ. – Интернета вещей) – более гибкое архитектурное решение, он может использоваться в системах реального времени (операционных системах реального времени). Более энергосберегающий и адаптирован под централизованные системы с ведущими и ведомыми устройствами, где брокеры сообщений связывают всех воедино.

MavLink – разработан для БПЛА, его архитектура и структура сообщений оптимизированы для использования в ограниченных ресурсах микроконтроллеров, что делает его предпочтительным для автономных систем с высокой задержкой и пропускной способностью.

Негативные факторы ухудшения связи в БПЛА

Негативные факторы, которые могут ухудшить связь в БПЛА (Беспилотные летательные аппараты), включают холодную пайку, дребезг контактов и помехи электромагнитного излучения (ПЭМИН).

Холодная пайка. Ошибка в процессе пайки, когда место пайки и контакт недостаточно прогревается или не образуется надлежащее соединение между элементами. Негативно влияет на передачу информации, в месте плохого контакта может искажаться проходящий сигнал (рис. 7).

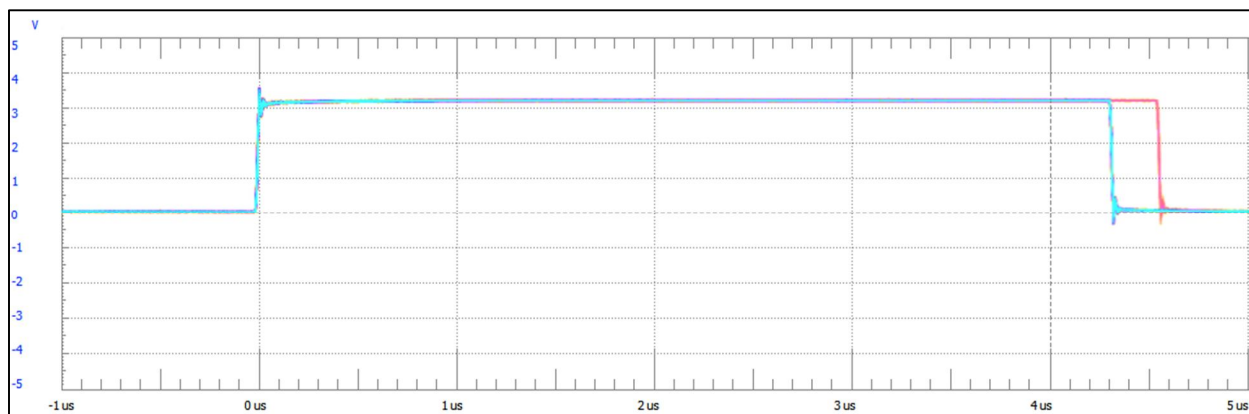


Рис. 7. Осциллограмма перекрёстно замкнутых TX и RX линий в момент отправки сообщения по протоколу UART

Fig. 7. Oscillogram of cross-connected TX and RX lines at the moment of sending message of UART protocol

Холодная пайка может привести к ненадежным соединениям и повышенному сопротивлению на паянных контактах, что может вызвать некачественную передачу сигналов и ухудшение связи. Для борьбы с данными сбоями используют качественные припои, флюсы, нагревать место соединений до правильной температуры и обеспечивать надлежащую связь между паяными контактами, прибегать к автоматизации процесса пайки [3, 7, 9].

Помехи электромагнитного излучения (ПЭМИН)

ПЭМИН — электромагнитные помехи, которые могут возникнуть вокруг электронных устройств или кабелей, искажая или прерывая передачу сигнала.

Они могут привести к ошибкам в передаче данных и потере сигнала, особенно на высоких частотах. Для борьбы с ПЭМИН используются экранированные кабели, ферритовые кольца, а также правильное размещение и экранирование электронных компонентов.

Подтяжки к питанию и земле

Подтяжки к питанию и земле (Pull-up и Pull-down) — это резисторы, которые подключаются к пинам линий данных устройства, чтобы обеспечить определенный уровень напряжения (высокий или низкий) при отсутствии других внешних сигналов. На линии I²C используется обязательная подтяжка к питанию (PullUp) 2.5 вольт резисторами R259 и R261 номиналом 4.7 КоМ (рис. 8).

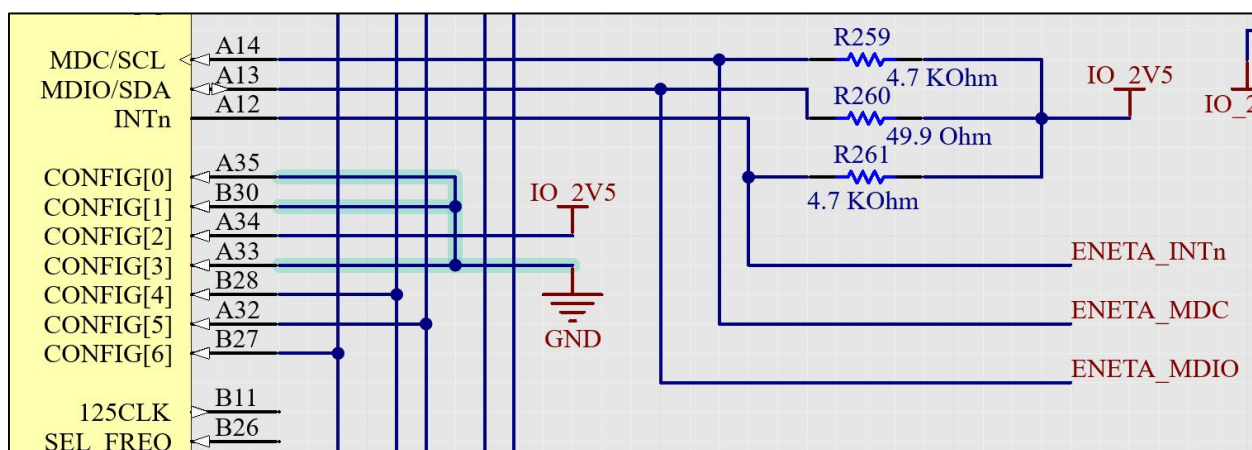


Рис. 8. Электрическая схема сетевого адаптера в БПЛА

Fig. 8. Electrical diagram of the network adapter in the UAV

Подтяжки используются для стабилизации линий связи и предотвращения случайных изменений уровней напряжения, они обеспечивают определенное начальное состояние линий связи, что позволяет предотвратить нежелательные переключения сигналов при подключении или отключении устройств [7-10].

В результате исследования были выявлены следующие механизмы обеспечения безопасности:

1. Шифрование данных: в протоколах MQTT и CoAP могут использоваться механизмы шифрования TLS/SSL (Transport Layer Security / Secure Sockets Layer), которые обеспечивают защищенное соединение между клиентами и серверами. Это позволяет защитить передаваемые данные от перехвата и несанкционированного доступа [11].

2. Аутентификация: в том же протоколе MQTT есть возможность использования имени пользователя и пароля для аутентификации клиентов. Это

позволяет удостоверить легитимность участников передачи данных.

3. Цифровая подпись: в протоколе MAVLink реализована поддержка цифровой подписи сообщений для обеспечения их подлинности и целостности. Цифровая подпись гарантирует, что сообщения не были изменены после отправки и что их отправитель является доверенным.

4. Проверка контрольных сумм: в протоколах низких уровней: UART, SPI и I2C контрольные суммы или CRC (циклический избыточный код) могут использоваться для обнаружения ошибок в передаче данных по физическим интерфейсам. Позволяет компонентам в БПЛА обнаружить поврежденные или потерянные данные и предпринять соответствующие меры, например, повторно запросить поврежденные данные.

5. Обмен ключами: возвращаясь к MQTT, при использовании шифрования TLS/SSL может выполняться обмен ключами.

чами для установления безопасного канала связи между клиентами и брокером. Обмен ключами обеспечивает конфиденциальность и защиту передаваемых данных.

6. Ограничение прав доступа: в противовес, в протоколе MAVLink могут быть определены различные уровни доступа к командам и параметрам, что позволяет ограничить права доступа различных компонентов в БПЛА. Например, некоторые команды могут быть доступны только для авторизованных устройств [12-14].

7. Обновления прошивки и защита от вторжений: Интеллектуальные компоненты в БПЛА, такие как микроконтроллеры, могут быть обновлены с использованием защищенных каналов связи. Это позволяет обновлять прошивку, исправлять уязвимости и улучшать безопасность системы.

Обновление прошивки можно проводить по защищенным каналам напрямую к облачным серверам производителя (такое обновление называется «Обновление по воздуху» - OTA). При установлении соединения требуется отдельно проверить подлинность сервера – на случай, если используется подмена конечного сервера для попытки взлома БПЛА. Также, ввиду того, что зачастую в дронах используются 32-битные процессоры с высокой тактовой частотой, основанные на ARM/RISC и других аналогичных архитектурах, на многих из них возможен запуск ОС Linux после определённых действий по портирова-

нию (перестройке образа системы под определенную архитектуру БПЛА).

В таком случае, благодаря универсальности ОС Linux, возможно установить и использовать систему обнаружения вторжений, например, Snort. Также можно фильтровать трафик на стороне наземной станции.

8. Отказоустойчивость и резервирование: в БПЛА используется множество интеллектуальных компонентов и датчиков, работающих в различных режимах и диапазонах. Чтобы обеспечить высокую отказоустойчивость и надежность работы системы, следует применить стратегию резервирования датчик и питания.

Например, в случае отказа одного из компонентов, система автоматически переключится на работу с резервным компонентом, что обеспечит непрерывность работы БПЛА.

В случае отказа основного блока аккумулятора, можно реализовать систему аварийного ввода резерва и подключение от запасного аккумулятора.

Резервирование также может помочь в случае с комплексными преградами. Предположим, стоит задача в полёте обнаружить подвижные препятствия. Если один из датчиков выходит из строя или предоставляет неточные данные (например, датчик УФ может засветить солнечный луч), система может автоматически переключиться на работу с другим рабочим датчиком и продолжить выполнение задачи навигации.

9. Фильтрация входящих данных [12-15]. Интеллектуальные компоненты в БПЛА могут применять фильтрацию входящих данных, чтобы предотвратить получение и обработку некорректных, аномальных или вредоносных данных от других устройств. Фильтры могут быть реализованы на уровне программного обеспечения или же встроены в аппаратное обеспечение, например, в тот же развёрнутый дистрибутив Linux.

Однако важно понимать, что БПЛА важно делать легковесным – это в первую очередь аппарат, который в основном занят обеспечением полёта, а развёртывание систем проактивного обнаружения и противодействия вторжений может вызвать сильную конкуренцию за ресурсы в БПЛА и в конечном итоге оказать негативное влияние на работу архитектуры в целом, даже когда атаки никакие и не совершаются.

Тем более, аномальное поведение сети в стрессовых условиях, в движении БПЛА и помехах вокруг вполне может быть беспочвенным и не представляющим опасность – например, может приходиться электромагнитная наводка на антенны дрона.

10. Отключение неиспользуемых интерфейсов: если рассматривать процессор\микроконтроллер БПЛА детально как интеллектуальный компонент, то он зачастую имеет несколько интерфейсов для связи с другими устройствами, которые могут не использоваться и просто использоваться в режиме ожидания\отладки. При использовании про-

токолов, таких как SPI, UART или I²C, можно отключить неиспользуемые интерфейсы, чтобы предотвратить возможность несанкционированного доступа или атак на неиспользуемые порты, допустим если БПЛА был перехвачен и подвергается реверс-инжинирингу.

11. Проверка и ограничение размера сообщений: в протоколах MQTT и CoAP можно настроить максимальный размер сообщений для передачи. Это позволяет предотвратить передачу чрезмерно больших пакетов данных, которые могут привести к переполнению буферов и атакам на переполнение буфера.

12. Защита от ретрансляции: для протоколов, таких как MAVLink и MQTT, можно применять механизмы защиты от повторной передачи (replay attack). Это предотвращает возможность воспроизведения старых или повторных команд и сообщений, что может быть использовано злоумышленниками для внедрения нежелательных команд или данных [14-17].

13. Проверка подлинности источников: в протоколах MQTT и MAVLink можно применять механизмы проверки подлинности источников сообщений. Это позволяет убедиться, что полученные данные и команды действительно исходят от доверенного и аутентифицированного источника.

14. Физическая безопасность: Компоненты в БПЛА могут быть размещены в защищенных от несанкционированного доступа корпусах или креплениях. Также можно использовать методы маркиров-

ки и аутентификации компонентов, чтобы обеспечить физическую безопасность и предотвратить возможность замены или подмены устройств – это могут быть пломбы, легкоплавкие предохранители, собственные системы защиты.

Протоколы прикладного уровня передачи потоковых данных [15-18]

RTSP (Real Time Streaming Protocol)

RTSP (Real-Time Streaming Protocol) – это протокол передачи потокового видео и аудио данных в реальном времени через сети IP. Он был разработан для обеспечения доставки мультимедийных данных с минимальными задержками, что делает его особенно подходящим для применения в системах, где требуется стриминг данных в реальном времени, таких как БПЛА для передачи видео с камеры в операционный центр. RTSP имеет ряд особенностей, благодаря которым он может применяться при работе с БПЛА:

- **Управление потоком:** RTSP предоставляет возможность управлять потоками мультимедийных данных – это функции паузы, воспроизведения, остановки и перемотки видео потоков.

- **Является протоколом управления данными** – он не транспортирует сам поток данных, а только управляет воспроизведением и передачей

- **Гибкий:** может использоваться как при трансляции видео, так и в видеоконференциях и дистанционным управлением.

- **Открытый:** RTSP является открытым стандартом, что обеспечивает

совместимость с различными устройствами и программными решениями.

- **Работает в реальном времени:**

RTSP обеспечивает передачу видео данных в реальном времени, что позволяет операторам получать актуальную информацию и принимать решения на основе последних данных.

Недостатки:

- **Чувствительные задержки:** несмотря на стремление к минимизации задержек, передача видео данных по сети ввиду физического канала всегда сопряжена с некоторой задержкой, что может быть критически важно в некоторых миссиях БПЛА.

- **Зависим от качества сети:** RTSP требует стабильного и надежного подключения к сети, что может быть ограничивающим фактором в условиях ограниченной доступности сетевой инфраструктуры.

- **Нет шифрования:** RTSP по умолчанию не реализует функции, связанные с безопасностью трафика

RTP (Real-Time Transport Protocol)

– это протокол транспортировки потоковых данных, таких как аудио и видео, в реальном времени через сети IP.

Он является основным протоколом для передачи данных во время аудио- и видеоконференций, стриминга мультимедиа и других приложений, где требуется передача данных в режиме реального времени с минимальной задержкой и потерями.

RTP является более выгодным для применения в БПЛА протоколом, по

сравнению с RTSP и имеет свои положительные стороны:

1. Реализация механизмов синхронизации: RTP включает в себя механизмы временной метки и синхронизации для обеспечения правильного воспроизведения мультимедийных данных в реальном времени.

2. Обработка потерь пакетов: RTP позволяет восстанавливать данные в случае потери некоторых пакетов, что особенно важно для обеспечения качества передачи в условиях непостоянной сети.

3. Минимизация задержек и джиттера: RTP стремится минимизировать задержки и джиттер (переменные задержки пакетов), чтобы обеспечить непрерывную и плавную передачу данных.

4. Реальное время: RTP обеспечивает передачу потоковых данных в реальном времени, что делает его идеальным для передачи видео с БПЛА на наземные станции

5. Адаптивность: RTP способен адаптироваться к изменениям в сети и динамически изменять параметры передачи данных, чтобы минимизировать задержки и потери.

6. Открытый стандарт: RTP является открытым стандартом, что обеспечивает его совместимость с различными устройствами и программами.

Конечно, RTP не идеален, из минусов можно выделить [17-20]:

1. Значительная нагрузка на сеть при потоковой передаче данных, особенно при высоком разрешении видео

или большой пропускной способности данных.

2. Зависимость от качества сети: качество передачи данных с помощью RTP сильно зависит от качества сети, и низкое качество связи может привести к потерям данных или ухудшению качества видео.

3. Безопасность: при использовании RTP для передачи данных важно обеспечить соответствующие меры безопасности, чтобы предотвратить несанкционированный доступ к потоковым данным.

WebRTC (Web Real-Time Communication)

WebRTC (Web Real-Time Communication) — открытый стандарт, который предоставляет возможность передачи аудио, видео и данных в режиме реального времени между веб-браузерами и мобильными приложениями без необходимости установки дополнительных плагинов или программного обеспечения. Данный протокол используется в видеоконференциях и веб-камерах стационарного исполнения, для потоковой трансляции картинки в браузер подключающегося к ним клиентов.

В БПЛА он тоже находит своё применение, благодаря характерным особенностям:

- 1) простота использования - картинка в браузере у клиента;
- 2) снижение задержки в реальном времени;
- 3) может передавать одновременно аудио, видео и произвольные данные

между пользователями – можно передавать таким образом служебную информацию;

4) пиринговая коммуникация (peer-to-peer, точка-точка) позволяет БПЛА соединяться напрямую с клиентом на стороне центра управления;

5) не требует установки дополнительного ПО и поддерживается многими ОС по умолчанию.

Минусы WebRTC в сфере применения в БПЛА весьма аналогичные:

1) зависимость от стабильного интернет-соединения для передачи данных, что мешает дрону в зонах с ограниченным доступом к сети;

2) WebRTC не реализует встроенных механизмов и протоколов безопасности;

3) прямая пиринговая коммуникация в WebRTC может иметь ограничение на количество одновременных пользователей, что может быть проблемой в крупных сетях с множеством пользователей.

SRT (Secure Reliable Transport) — это открытый протокол транспортировки данных, который обеспечивает безопасную и надежную передачу аудио- и видеопотоков в реальном времени через сети IP. Разработан для преодоления проблем низкой пропускной способности, потерь данных и высокой задержки, что делает его особенно подходящим для применения в условиях неблагоприятных сетевых условий, таких как недостаточная пропускная способность или нестабильное соединение.

У SRT много ключевых особенностей:

1. Надежность и восстановление потерь: SRT обеспечивает механизмы для обнаружения и восстановления потерянных пакетов данных, что делает его более надежным в условиях сетевых неполадок и потерь.

2. Адаптивность к сетевым условиям: SRT автоматически адаптируется к изменениям в сети, регулируя скорость передачи данных для обеспечения оптимальной производительности.

3. Защищенность данных: SRT предоставляет возможность для шифрования данных, обеспечивая безопасную передачу потоковых данных через открытые сети.

Недостатки:

1. Некоторую сложность настройки: настройка SRT может потребовать дополнительных усилий и экспертизы, особенно в случае нестандартных или сложных сетевых конфигураций.

2. Зависимость от поддержки устройств: SRT требует поддержки со стороны оборудования, включая БПЛА и приемное оборудование, чтобы обеспечить совместимость и правильную работу протокола.

3. Возможные накладные расходы: внедрение SRT может потребовать дополнительных ресурсов, включая высокоскоростные сети или дополнительное оборудование для обеспечения оптимальной производительности.

Выводы

С помощью серии экспериментов и симуляций в статье оценивается эффективность предложенного подхода по снижению уязвимостей и смягчению потенциальных рисков, связанных с обменом данными в БПЛА. Результаты демонстрируют, что включение интеллектуальных компонентов в процесс обмена данными значительно повышает общую безопасность информации, минимизируя нагрузку на системы БПЛА. Данное исследование вносит вклад в

совершенствование безопасности информации для БПЛА, предоставляя практические знания по оптимизации протоколов обмена данными с использованием интеллектуальных компонентов. Полученные результаты окажут пользу операторам БПЛА, производителям и специалистам по безопасности, способствуя разработке более надежных и безопасных систем БПЛА, что является важным для безопасности и надежности их применения в современных приложениях.

Список литературы

1. Каляев А. И., Коровин Я. С. Комплекс обнаружения и поражения БПЛА-нарушителей с помощью БПЛА-перехватчиков // Вопросы оборонной техники. Серия 16: Технические средства противодействия терроризму. 2021. № 3-4(153-154). С. 101-107. EDN GMUUD.
2. Аблец А. А., Стребков А. Н., Завгородняя Е. В. Опыт создания роя БПЛА в вооруженных силах иностранных государств // Военная мысль. 2022. № 6. С. 137-146. EDN ISYTIR.
3. Пенской И. С., Рогозин О. В. Нейро-нечеткий ПИД-регулятор в задаче угловой стабилизации мультироторного БПЛА // Новые информационные технологии в автоматизированных системах. 2018. № 21. С. 320-327. EDN XNRSJV.
4. Актуальные вопросы управления группой БПЛА / И. Н. Балащенко, Н. А. Горыня, А. С. Гунько, В. И. Никонов // Динамика систем, механизмов и машин. 2022. Т. 10, № 1. С. 14-17. DOI 10.25206/2310-9793-2022-10-1-14-17. EDN ХНЕНРМ.
5. Подоплекин Ю. Ф., Толмачев С. Г., Шаров С. Н. Информационно-управляющая система приведения беспилотных летательных аппаратов на движущееся судно // Информационно-управляющие системы. 2012. № 3(58). С. 22-28. EDN OYIGWZ.
6. Силкин А. А., Ивановский А. Н., Черный С. Г. Автоматизированная система управления БПЛА в пределах локальной системы позиционирования // Состояние и перспективы развития современной науки по направлению "Робототехника": сборник статей IV Всероссийской научно-технической конференции, Анапа, 20–21 июля 2022 г. Анапа, 2022. С. 224-235. EDN WTLPIU.
7. Michels M., Wever H., Musshoff O. 'Drones take flight in forests: Uncovering the 'tree'-mendous potential and 'timber'-rific challenges for German forest managers by apply-

ing the technology acceptance model // *Forest Policy and Economics*. 2023. 157. P. 103077. doi:10.1016/j.forpol.2023.103077.

8. Zhu Z., Jeelani I., Gheisari M. 'Physical risk assessment of drone integration in construction using 4D simulation' // *Automation in Construction*, 2023. 156. P. 105099. doi:10.1016/j.autcon.2023.105099.

9. Garg V. *et al.* Drones in last-mile delivery: A systematic review on efficiency, accessibility, and Sustainability' // *Transportation Research Part D: Transport and Environment*, 2023. 123. P. 103831. doi:10.1016/j.trd.2023.103831.

10. Кутахов В. П., Мещеряков Р. В. Управление групповым поведением беспилотных летательных аппаратов: постановка задачи применения технологий искусственного интеллекта // *Проблемы управления*. 2022. № 1. С. 67-74. DOI 10.25728/ru.2022.1.5. EDN J LXUXI.

11. Мещеряков Р. В., Илюхин Б. В. Обзор соревновательной робототехники. Соревнования спортивных роботов // *Робототехника и техническая кибернетика*. 2022. Т. 10, № 4. С. 255-260. DOI 10.31776/RTCJ.10402. EDN ISXMEV.

12. Емельянов С.Г., Бобырь М. В., Бондаренко Б. А. Нечетко-логическая система распознавания цвета с помощью быстродействующего дефаззификатора // *Известия Юго-Западного государственного университета*. 2022; 26(4): 103-116. <https://doi.org/10.21869/2223-1560-2022-26-4-103-116>.

13. Емельянов С. Г., Бобырь М. В., Крюков А. Г. Исследование свойств алгоритма поиска в ширину для нахождения маршрута передвижения роботов // *Известия Юго-Западного государственного университета*. 2022; 26(4): 39-56. <https://doi.org/10.21869/2223-1560-2022-26-4-39-56>.

14. Alwateer M., Loke S. W., Fernando N. Enabling drone services: Drone crowdsourcing and drone scripting // *IEEE Access*. 2019. Vol. 7. P. 110035-110049. DOI 10.1109/ACCESS.2019.2933234. EDN ICGTGY.

15. Hassija V., Saxena V., Chamola V. Scheduling drone charging for multi-drone network based on consensus time-stamp and game theory // *Computer Communications*. 2020. Vol. 149. P. 51-61. DOI 10.1016/j.comcom.2019.09.021. EDN WAIQY.

16. Hu J., Lanzon A. An innovative tri-rotor drone and associated distributed aerial drone swarm control // *Robotics and Autonomous Systems*. 2018. Vol. 103. P. 162-174. DOI 10.1016/j.robot.2018.02.019. EDN MEFVVB.

17. Kim H., Chang W. Intuitive Drone Control using Motion Matching between a Controller and a Drone // *Archives of Design Research*. 2022. Vol. 35, No. 1. P. 93-113. DOI 10.15187/adr.2022.02.35.1.93. EDN RTZJES.

18. Cocoma-Ortega J. A., Martinez-Carranza J. A compact CNN approach for drone localisation in autonomous drone racing // *Journal of Real-Time Image Processing*. 2022. Vol. 19, No. 1. P. 73-86. DOI 10.1007/s11554-021-01162-3. EDN CFYATS.

19. An Optimization Approach to Minimize the Expected Loss of Demand Considering Drone Failures in Drone Delivery Scheduling / M. Torabbeigi, G. J. Lim, N. Ahmadian, S. J.

Kim // Journal of Intelligent and Robotic Systems. 2021. Vol. 102, No. 1. DOI 10.1007/s10846-021-01370-w. EDN RBLKPP.

20. Noise Attenuation on IMU Measurement for Drone Balance by Sensor Fusion / M. L. Hoang, M. Carratu, V. Paciello, A. Pietrosanto // Conference Record - IEEE Instrumentation and Measurement Technology Conference, Virtual, Glasgow, 17–20 мая 2021 года. Virtual, Glasgow, 2021. P. 9460041. DOI 10.1109/I2MTC50364.2021.9460041. EDN HYQCOI.

References

1. Kaljaev A. I., Korovin Ja. S. Kompleks obnaruzhenija i porazhenija BPLA-narushitelej s pomoshh'ju BPLA-perehvatchikov [Complex for detecting and defeating intruder UAVs using interceptor UAVs]. *Voprosy oboronnoj tehniki. Serija 16: Tehnicheskie sredstva protivodejstvija terrorizmu = Questions of Defense Technology. Series 16: Technical Means of Countering Terrorism*, 2021, no. 3-4(153-154), pp. 101-107. EDN GMUUOD.

2. Ablec A. A., A Strebkov. N., Zavgorodnjaja E. V. Opyt sozdaniya roja BPLA v vooruzhennyh silah inostrannyh gosudarstv [The experience of creating a swarm of UAVs in the armed forces of foreign states]. *Voennaja mysl' = Military Thought*, 2022, no. 6, pp. 137-146. EDN ISYTIR.

3. Penskoj I. S., Rogozin O. V. Nejro-nechetkij PID-reguljator v zadache uglovoj stabilizacii mul'tirotnogo BPLA [Neuro-fuzzy PID controller in the problem of angular stabilization of a multicopter UAV]. *Novye informacionnye tehnologii v avtomatizirovannyh sistemah = New Information Technologies in Automated Systems*, 2018, no. 21, pp. 320-327. EDN XNRSJV.

4. Balashenko I. N., Gorynja N. A., Gun'ko A. S., Nikonov V. I. Aktual'nye voprosy upravlenija gruppoy BPLA [Topical issues of UAV group management]. *Dinamika sistem, mehanizmov i mashin = Dynamics of Systems, Mechanisms and Machine*, 2022. vol. 10, no. 1, pp. 14-17. DOI 10.25206/2310-9793-2022-10-1-14-17. – EDN XHEHPM.

5. Podoplekin Ju. F., Tolmachev S. G., Sharov S. N. Informacionno-upravljajushhaja sistema privedenija bespilotnyh letatel'nyh apparatov na dvizhushheesja sudno [Information and control system for bringing unmanned aerial vehicles to a moving vessel]. *Informacionno-upravljajushhie sistemy = Information and Control Systems*, 2012, no. 3(58), pp. 22-28. EDN OYIGWZ.

6. Silkin A. A., Ivanovskij A. N., Chernyj S. G. [Automated UAV control system within the local positioning system]. *Sostojanie i perspektivy razvitija sovremennoj nauki po napravleniju "Robototekhnika": Sbornik statej IV Vserossijskoj nauchno-tehnicheskoy konferencii* [The state and prospects of development of modern science in the field of Robotics. Collection of articles of the IV All-Russian Scientific and Technical Conference]. Anapa, 2022, pp. 224-235 (In Russ.). EDN WTLPIU.

7. Michels M., Wever H. Musshoff O. Drones take flight in forests: Uncovering the 'tree'-mendous potential and 'timber'-rific challenges for German forest managers by applying the technology acceptance model'. *Forest Policy and Economics*, 2023, 157, p. 103077. doi:10.1016/j.forpol.2023.103077.
8. Zhu Z., Jeelani I., Gheisari M. Physical risk assessment of drone integration in construction using 4D simulation. *Automation in Construction*, 2023, 156, p. 105099. doi:10.1016/j.autcon.2023.105099.
9. Garg V. et al. Drones in last-mile delivery: A systematic review on efficiency, accessibility, and Sustainability. *Transportation Research Part D: Transport and Environment*, 2023, 123, p. 103831. doi:10.1016/j.trd.2023.103831.
10. Kutahov V. P., Meshherjakov R. V. Upravlenie gruppovym povedeniem bespilotnyh letatel'nyh apparatov: postanovka zadachi primeneniya tehnologij iskusstvennogo intellekta [Management of group behavior of unmanned aerial vehicles: setting the task of using artificial intelligence technologies]. *Problemy upravlenija = Problems of Management*, 2022, no. 1, pp. 67-74. DOI 10.25728/pu.2022.1.5. EDN J LXUXI.
11. Meshherjakov R. V., Iljuhin B. V. Obzor sorevnovatel'noj robototekhniki. Sostjazaniya sportivnyh robotov [Review of competitive robotics. Combinations of sports robots]. *Robototekhnika i tehničeskaja kibernetika = Robotics and Technical Cybernetics*, 2022, vol. 10, no. 4, pp. 255-260. DOI 10.31776/RTCJ.10402. – EDN ISXMEV.
12. Emelianov S. G., Bobyr M. V., Bondarenko B. A. Fuzzy-logic Color Recognition System Using a Fast Defuzzifier. *Izvestiya Yugo-Zapadnogo gosudarstvennogo universiteta = Proceedings of the Southwest State University*. 2022; 26(4): 103-116 (In Russ.). <https://doi.org/10.21869/2223-1560-2022-26-4-103-116>.
13. Emelianov S. G., Bobyr M. V., Kryukov A. G. Research of the Properties of the Breadth-First Search Algorithm for Finding the Movement Route of Robots. *Izvestiya Yugo-Zapadnogo gosudarstvennogo universiteta = Proceedings of the Southwest State University*. 2022; 26(4): 39-56 (In Russ.). <https://doi.org/10.21869/2223-1560-2022-26-4-39-56>.
14. Alwateer M., Loke S. W., Fernando N. Enabling drone services: Drone crowdsourcing and drone scripting. *IEEE Access*, 2019, vol. 7, pp. 110035-110049. DOI 10.1109/ACCESS.2019.2933234. EDN ICGTGY.
15. Hassija V., Saxena V., Chamola V. Scheduling drone charging for multi-drone network based on consensus time-stamp and game theory. *Computer Communications*, 2020, vol. 149, pp. 51-61. DOI 10.1016/j.comcom.2019.09.021. EDN WAIQEY.
16. Hu J., Lanzon A. An innovative tri-rotor drone and associated distributed aerial drone swarm control. *Robotics and Autonomous Systems*, 2018, vol. 103, pp. 162-174. DOI 10.1016/j.robot.2018.02.019. EDN MEFVVB.
17. Kim H., Chang W. Intuitive Drone Control using Motion Matching between a Controller and a Drone. *Archives of Design Research*, 2022, vol. 35, no. 1, pp. 93-113. DOI 10.15187/adr.2022.02.35.1.93. EDN RTZJES.

18. Cocoma-Ortega J. A., Martinez-Carranza J. A compact CNN approach for drone localisation in autonomous drone racing. *Journal of Real-Time Image Processing*, 2022, Vol. 19, no. 1, pp. 73-86. DOI 10.1007/s11554-021-01162-3. EDN CFYATS.
19. Torabbeigi M., Lim G. J., Ahmadian N., Kim S. J. An Optimization Approach to Minimize the Expected Loss of Demand Considering Drone Failures in Drone Delivery Scheduling. *Journal of Intelligent and Robotic Systems*, 2021, vol. 102, no. 1. DOI 10.1007/s10846-021-01370-w. EDN RBLKPP.
20. Hoang M. L., Carratu M., Paciello V., Pietrosanto A. Noise Attenuation on IMU Measurement for Drone Balance by Sensor Fusion. *Conference Record - IEEE Instrumentation and Measurement Technology Conference*, Virtual, Glasgow, 17–20 maja 2021 goda. Virtual, Glasgow, 2021, 9460041 p. DOI 10.1109/I2MTC50364.2021.9460041. EDN HYQCOI.

Информация об авторах / Information about the Authors

Черный Сергей Григорьевич, кандидат технических наук, доцент кафедры электрооборудование судов и эксплуатации производства, профессор кафедры КОИБ, Государственный университет морского и речного флота имени адмирала С.О. Макарова, г. Санкт-Петербург, Российская Федерация, Керченский государственный морской технологический университет, г. Керчь, Российская Федерация, e-mail: sergiiblack@gmail.com

Sergey G. Cherny, Cand. of Sci. (Engineering), Associate Professor of Electrical Equipment of Ships and Production Operation Department, Professor of the Department of COIB, State University of Marine and River Fleet named after admiral S.O. Makarov, St. Petersburg, Russian Federation, Kerch State Marine Technological University, Kerch, Russian Federation, e-mail: sergiiblack@gmail.com

Шипаренко Никита Витальевич, студент, Государственный университет морского и речного флота имени адмирала С.О. Макарова, г. Санкт-Петербург, Российская Федерация, e-mail: nikita4shap@gmail.com

Nikita V. Shiparenko, Student, State University of Marine and River Fleet named after admiral S.O. Makarov, St. Petersburg, Russian Federation, e-mail: nikita4shap@gmail.com

Чупаков Максим Викторович, начальник научно-исследовательской лаборатории, Черноморское высшее военно-морское училище имени П.С. Нахимова, г. Севастополь, Российская Федерация, e-mail: maxim.vor2012@yandex.ru

Maksim V. Chupakov, Head of the Research Laboratory, Black Sea Higher Naval School named after P.S. Nakhimov, Sevastopol, Russian Federation, e-mail: maxim.vor2012@yandex.ru