# Оглавление

[1. Тема работы и постановка задачи 2](#_Toc164455622)

[2. Подготовка тестовых звуковых файлов 3](#_Toc164455623)

[3. «Визуальный анализ методом пристального наблюдения» 4](#_Toc164455624)

[A. анализ спектра сигнала: 4](#_Toc164455625)

[B. Определение структуры сигнала 6](#_Toc164455626)

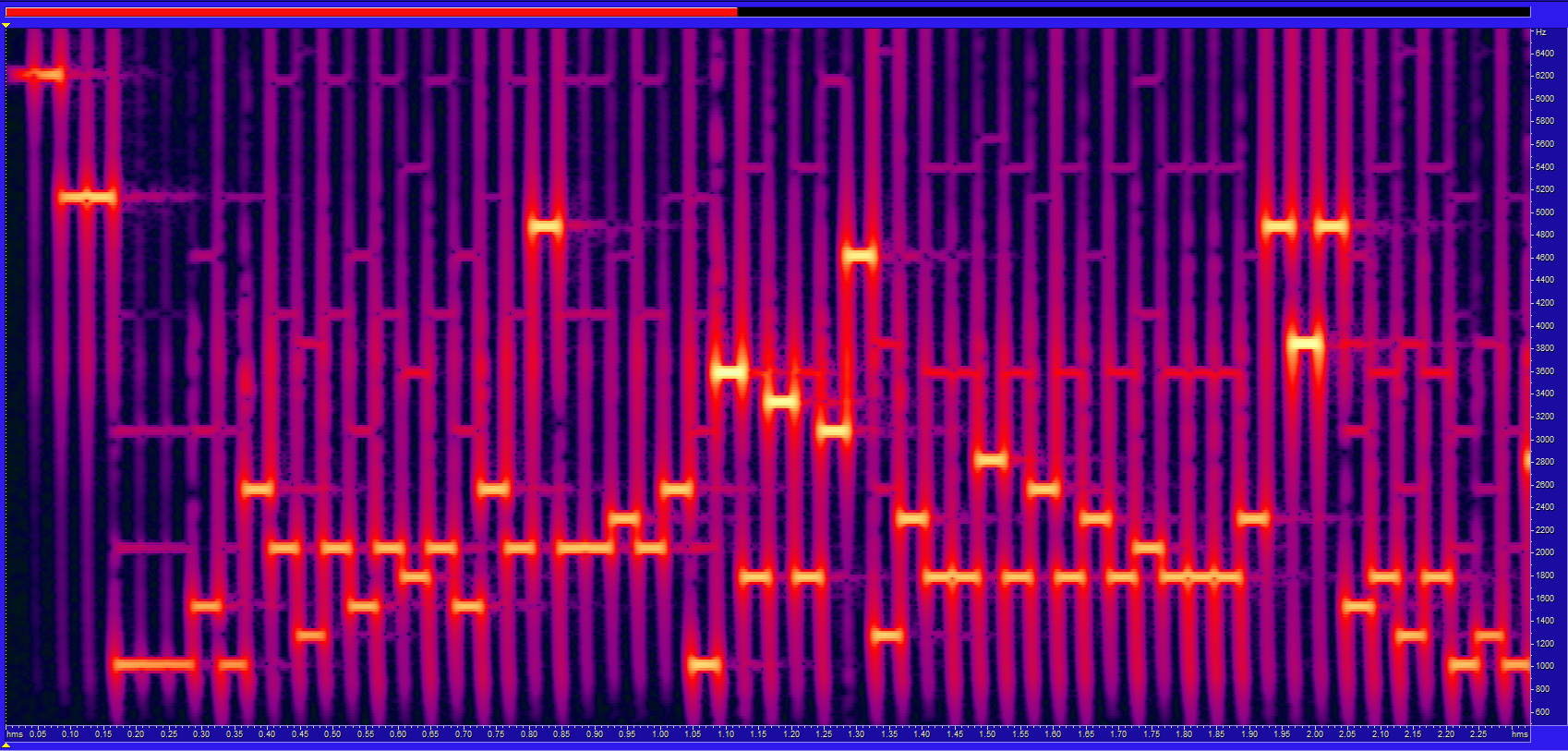
[C. Принцип кодирования сообщений в протоколе 6](#_Toc164455627)

[D. Аналитическая модель формирования сообщений 8](#_Toc164455628)

[4. Реализация алгоритма декодирования протокола 12](#_Toc164455629)

# Тема работы и постановка задачи

Одним из возможных способов подключения Яндекс.Станции к домашней сети WiFi является беспроводная передача данных сети (ssid и password) при помощи акустического сигнала. Данный сигнал представляет собой набор тональных импульсов, последовательность которых несёт в себе данные о подключении к сети, закодированные определённым образом в импульсной последовательности. Пример фрагмента передаваемого сигнала (сонограмма):



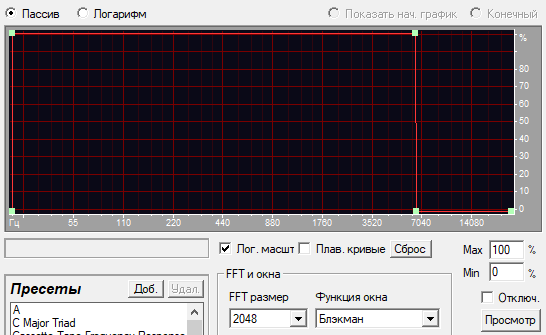
Механизм передачи: некое устройство, способное управлять Яндекс.Станцией (например, смартфон) генерирует и предаёт с помощью собственного встроенного динамика импульсную последовательность (пакет данных); сама Яндекс.Станция принимает данный сигнал с помощью встроенного микрофона и декодирует его. Для передачи данных используется протокол Яндекс.Алисы.

Задача: смоделируем ситуацию, при которой злоумышленнику удалось перехватить акустический сигнал, передающий данные о подключении к сети. Целью работы является создание ПО для автоматизации декодирования передаваемых данных.

# Подготовка тестовых звуковых файлов

Для записи тестовых звуковых сигналов был использован микрофонс функцией «бум – сглаживания», которая позволяет записывать импульсные акустические сигналы без образования разрывов 1-го рода в полученной временной огибающей. Запись акустических сигналов производилась с помощью программного пакета AdobeAudition (v1.5).

Также был использован «П» фильтр с функцией окна – «Хаффман» (КИХ фильтр с применением окна сглаживания), что позволяет избежать искажений по частоте, в отличии от использования фильтров с конечной импульсной характеристикой БЕЗ окон сглаживания:

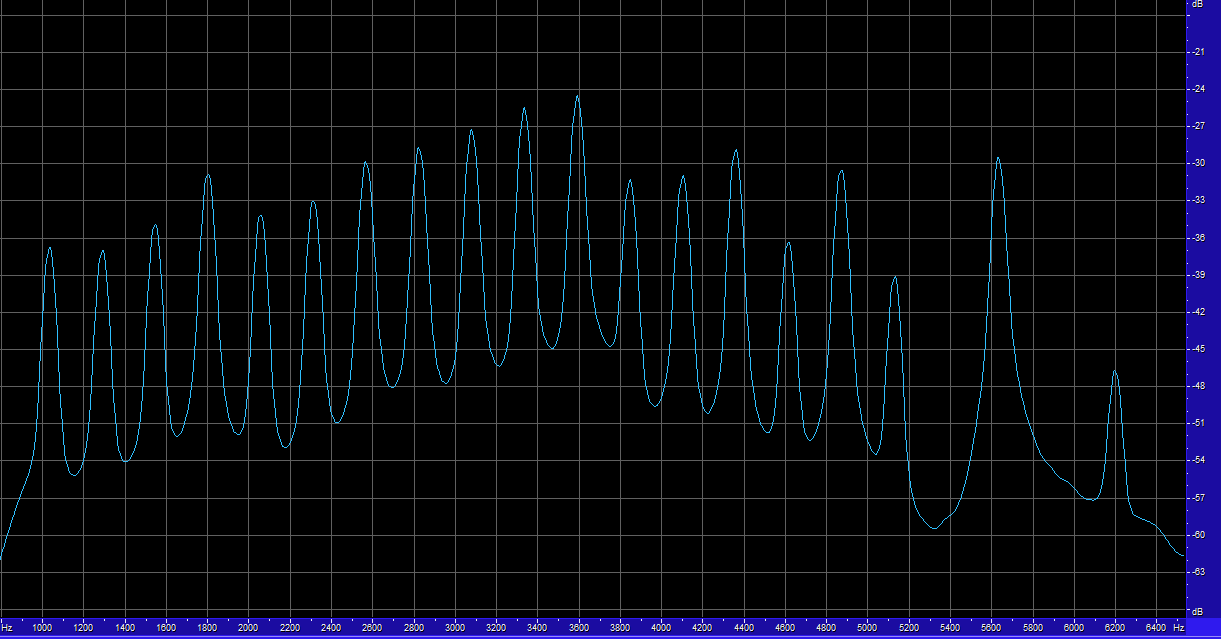


Все тестовые файлы были записаны в формате .wavс частотой дискретизации 48кГц.

# «Визуальный анализ методом пристального наблюдения»

## анализ спектра сигнала:

При анализе различных тестовых записей было обнаружено, что полный алфавит «команд» состоит из 19 элементарных сообщений конкретных частот.



Были обнаружены импульсы следующих частот:

таблица 1.

|  |  |  |
| --- | --- | --- |
| № | Частота f, Гц | Примечания |
| У следующих частот были обнаружены отклонения +-(0…10) Гц | | |
| 1 (0) | 1025 |  |
| 2 (1) | 1280 |  |
| 3 (2) | 1535 |  |
| 4 (3) | 1790 |  |
| 5 (4) | 2045 |  |
| 6 (5) | 2300 |  |
| 7 (6) | 2555 |  |
| 8 (7) | 2810 |  |
| 9 (8) | 3065 |  |
| У следующих частот были обнаружены отклонения +-(10…15) Гц | | |
| 10 (9) | 3320 |  |
| 11 (a) | 3575 |  |
| 12 (b) | 3830 |  |
| 13 (c) | 4085 |  |
| 14 (d) | 4340 |  |
| У следующих частот были обнаружены отклонения +-(15…20) Гц | | |
| 15 (e) | 4595 |  |
| 16 (f) | 4850 |  |
| Далее обнаружены сигналы синхронизации | | |
| 17 | 5105 | Вторая частота синхроимпульса (идёт после 6200 Гц) |
| 18 | 5625 | Частота завершающего синхроимпульса |
| 19 | 6200 | Первая частота синхроимпульса (идёт первой) |

Итого:

1) Шаг частот – 255 Гц

2) Частоты с номерами 17-19 встречаются только в начале и конце сигналов (участвуют в синхронизации), следовательно, информации не несут. Отсюда можно сделать предположение, что частоты 1-16 образуют некий шестнадцатеричный алфавит (являются ансамблем элементарных сообщений данного алфавита); каждый импульс несёт 2 байта информации. В соответствии с этим всем частотам 1-16 были присвоены соответствующие идентификаторы 0-fшестнадцатеричного алфавита.

## Определение структуры сигнала

При анализе множества тестовых сигналов было обнаружено:  
1) Все сигналы начинаются с т.н. «преамбулы», состоящей из 8 тональных импульсов конкретных частот:

таблица 2.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| № импульса | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| Частота, Гц | 6200 | 6200 | 5105 | 5105 | 1025 | 1025 | 1025 | 1535 |

Завершаются все пакеты 6 импульсами частоты 5625 Гц.

Между преамбулой пакета и хвостовиком находится сегмент, в котором передаётся как информация о подключении к сети, так и служебная информация.

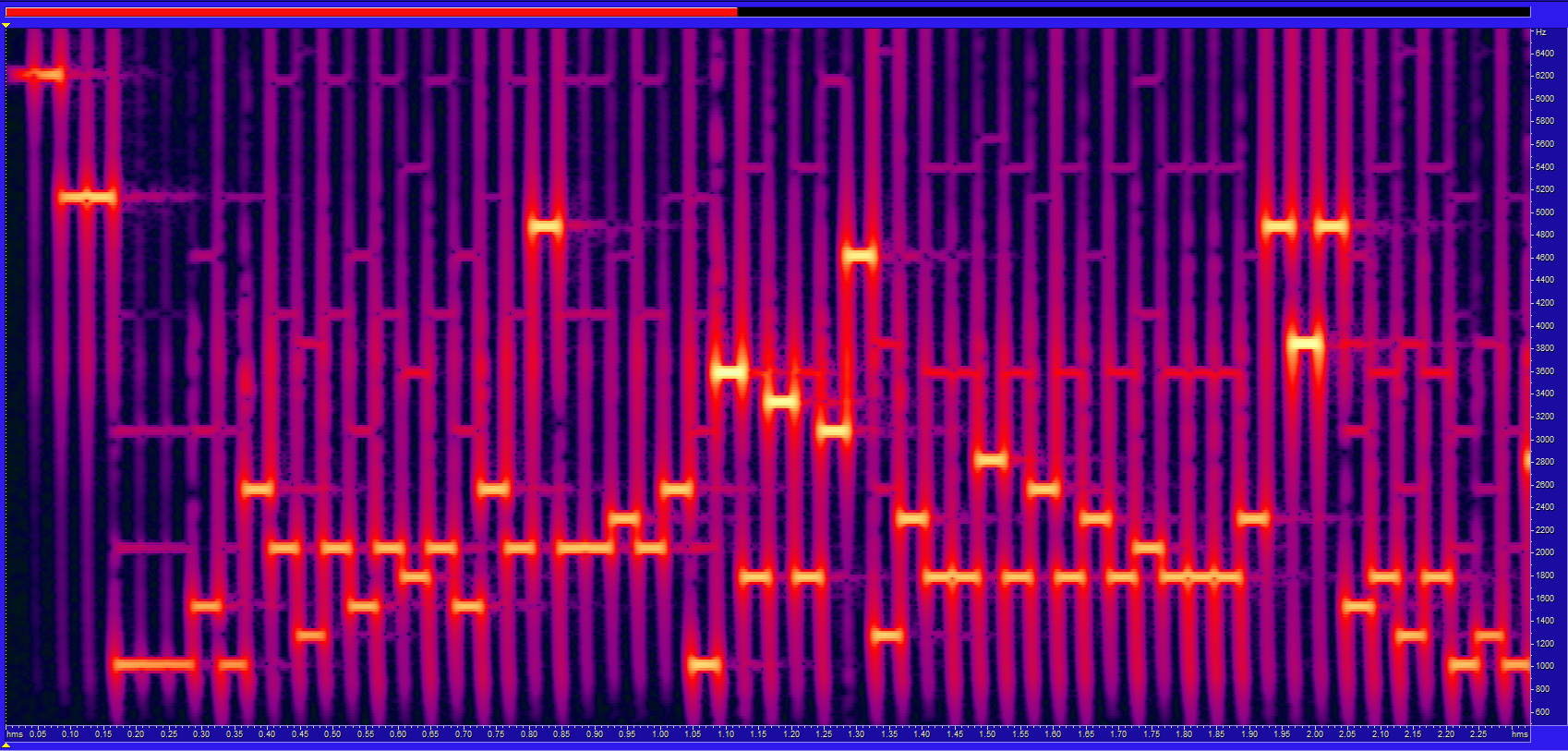
## Принцип кодирования сообщений в протоколе

Было сделано предположение, что коды символов, передаваемых по протоколы Яндекс.Алисы, записаны в кодировке ASCII. Применив данный подход к расшифрованию элементарных сообщений, было обнаружено, что присвоенные частотам идентификаторы 0-fскладываются в символы таблицы ASCII.

Принцип перевода: в таблице ASCII все привычные для нас символы (‘0’, ’h’, ’$’ и т.д.) кодируются двумя «знаками» 0-f из шестнадцатеричного алфавита. Заметим, что количество тональных импульсов в информативной части файла чётное, тогда сгруппируем все тональные импульсы последовательно по два, присвоим импульсам идентификаторы 0-f, а полученные пары сравним с таблицей кодов ASCII.

В качестве примера рассмотрим пакет, переданный по протоколу Яндекс.Алисы, в котором записан ssid сети *ABCDEF* и пароль сети *9876543210*:

(на рис. ниже представлена часть сонограммы файла, на которой присутствует ssidи passwordсети):



Рассмотрим, что удаётся восстановить:

таблица 3.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| № импульса | Частота | Идентификатор | Символ по ASCII | № импульса | Частота | Идентификатор | Символ по ASCII |
| 9. | 1025 | 0 | ? | 23. | 2045 | 4 | ‘E’ |
| 10. | 2555 | 6 | 24. | 2300 | 5 |
| 11. | 2045 | 4 | ‘A’ | 25. | 2045 | 4 | ‘F’ |
| 12. | 1280 | 1 | 26. | 2555 | 6 |
| 13. | 2045 | 5 | ‘B’ | 27. | 1025 | 0 | LF(?) |
| 14. | 1535 | 2 | 28. | 3575 | a |
| 15. | 2045 | 4 | ‘C’ | 29. | 1790 | 3 | ‘9’ |
| 16. | 1790 | 3 | 30. | 3320 | 9 |
| 17. | 2045 | 4 | ‘B’ | 31. | 1790 | 3 | ‘8’ |
| 18. | 1535 | 2 | 32. | 3065 | 8 |
| 19. | 2555 | 6 | ‘d’ | 33. | 4595 | e | - |
| 20. | 2045 | 4 | 34. | 1280 | 1 |
| 21. | 4850 | f | ‘\*’ | 35. | 2300 | 5 | ‘S’ |
| 22. | 2045 | 4 | 36. | 1790 | 3 |

Как видно из части данных, кодировка символов действительно соответствует стандарту ASCII. На первый взгляд, информацию удаётся восстановить не полностью. Всё дело в присутствии импульсов синхронизации, а также служебной информации, «подмешанной» в данные о *ssid*и *password* сети.

## Аналитическая модель формирования сообщений

В ходе анализа (использовались тестовые записи, содержащие данные *ssid/password*различной длины, состава букв/цифр/символов и т.д.) была определена и сформирована модель структуры передаваемой информации по протоколу Яндекс.Алисы.

(Далее встречаются следующие введённые термины: **информативные символы** – это либо части ssid/password, либо числовое значение длины ssid или password; **неинформативные или синхроимпульсы** – это наборы/элементы кодов ASCII, которые встроены в «информативные фразы» пакета, их части могут нести информацию [о том, как – далее].)

Итоговый вид:

таблица 4.

|  |  |
| --- | --- |
| Импульсы 1-8 | Преамбула (структура фиксированная) |
| Импульсы 9-10 | Длина ssidв 16-ом виде, является информативным символов; например, для сети ABCDEF/9876543210 (в т. 3) это 0616 = 610  (равно длине строки ABCDEF) |
| Импульсы 11-16 | 3 импульса, несущие информацию (являются информативным сиволом), после них идёт блок синхронизации |
| Импульсы 9-N(Nсоответствует инструкции DLE из ASCII) | Передаются данные в формате <длина ssid> + <ssid> + <длина password> + <password>  При этом, как упоминалось ранее, длины ssid/passwordили сами их значения являются информативными символами, а между ними вставлены блоки синхроимпульсов по определённым правилам (описано далее) |
| Символ-инструкция DLE | Послеблока<длинаssid> + <ssid> + <длинаpassword> + <password> следует два тональных импульса с частотами 1280 и 1025 Гц, что соответствует 1016, что в свою очередь соответствует инструкции DLEиз ASCII. |

После символа-инструкции DLE последующая информация, передаваемая в пакете данных, не представляет интереса для данной работы.

Основная часть пакета данных – блок <длина ssid> + <ssid> + <длина password> + <password>. Он состоит из блоков («пятёрок») по 5 информативных символов (соответственно «пятёрка» символов состоит из 10 тональных импульсов). Блоки (пятерки) разделены вставками по 3 или 2 синхросимвола, которые чередуются, начиная с синхровставки длиной 3 (т.е., соответственно, 3 синхросимвола = 6 тональных импульсов, либо 2 синхросимвола = 4 тональных импульса).

Как строятся «пятёрки» символов:

таблица 5.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 3 синхросимвола | 5 информативных символов | 2 синхпросимвола | 5 информативных символа | 3 синхросимвола | 5 информативных символов | 2 синхпросимвола | 5 информативных символа | … | DLE – конец интересующего блока |

При этом в «пятёрку» могут попасть как сами значения ssid/password, так и значение их длины как строки (как уже упоминалов ранее).

Отдельно стоит обратить внимание на структуру блоков синхронизации:

1) Блок из 2 синхросимволов (4 импульсов) не несёт информации в целом;

2) В структуре блока из 3 синхросимволов скрыт один информативный символ. 3 синхросимвола представляются 6 тональными импульсами, при этом 1-й и 6-й импульсы вместе дают информативный символ (!). Рассмотрим пример из т. 3:

|  |  |  |  |
| --- | --- | --- | --- |
| 11. | 2045 | 4 | ‘A’ |
| 12. | 1280 | 1 |
| 13. | 2045 | 5 | ‘B’ |
| 14. | 1535 | 2 |
| 15. | 2045 | 4 | ‘C’ |
| 16. | 1790 | 3 |
| 17. | 2045 | 4 | ‘B’ |
| 18. | 1535 | 2 |
| 19. | 2555 | 6 | ‘d’ |
| 20. | 2045 | 4 |
| 21. | 4850 | f | ‘\*’ |
| 22. | 2045 | 4 |
| 23. | 2045 | 4 | ‘E’ |
| 24. | 2300 | 5 |
| 25. | 2045 | 4 | ‘F’ |
| 26. | 2555 | 6 |

Как видно, импульсы 17-й и 22-й (т.е. 1-й и 6-й из синхроблока) дают код 44 – буква ‘D’ из ASCII. Итого, складывается последовательность ‘ABCDEF’ – ssidнашей сети. В синхронизации на 2 синхроимпульса такой приём не используется.

Как только в «пятёрке» заканчивается строка ssid, тут же передаётся значение длины поля password, а затем и само значение пароля.

В качестве примера полной структуры вернёмся к таблице, представляющей файл с ssid=ABCDEFи password=9876543210, и дополним её:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № импульса | Частота, Гц | Идентификатор | Символ по ASCII | Примечание | |
| 1. | 6200 |  |  | Преамбула | |
| 2. | 6200 |
| 3. | 5105 |
| 4. | 5105 |
| 5. | 1025 |
| 6. | 1025 |
| 7. | 1025 |
| 8. | 1535 |
| 9. | 1025 | 0 | - | 0616 = 610 – длина ssid | Блок до первой синхропоследовательности не образует «пятёрку», состоит строго из 4 информативных символов |
| 10. | 2555 | 6 |
| 11. | 2045 | 4 | ‘A’ |  |
| 12. | 1280 | 1 |
| 13. | 2045 | 5 | ‘B’ |  |
| 14. | 1535 | 2 |
| 15. | 2045 | 4 | ‘C’ |  |
| 16. | 1790 | 3 |
| 17. | 2045 | 4 | - | 44ASCII – символ ‘D’ | Синхронизация  (3 синхросимвола) |
| 18. | 1535 | 2 |
| 19. | 2555 | 6 | - |
| 20. | 2045 | 4 |
| 21. | 4850 | f | - |
| 22. | 2045 | 4 |
| 23. | 2045 | 4 | ‘E’ |  | Информативная «пятёрка» символов |
| 24. | 2300 | 5 |
| 25. | 2045 | 4 | ‘F’ |  |
| 26. | 2555 | 6 |
| 27. | 1025 | 0 | - | 0a16 = 1010 – длина password |
| 28. | 3575 | a |
| 29. | 1790 | 3 | ‘9’ |  |
| 30. | 3320 | 9 |
| 31. | 1790 | 3 | ‘8’ |  |
| 32. | 3065 | 8 |
| 33. | 4595 | e | - |  | Синхронизация  (2 синхросимвола) |
| 34. | 1280 | 1 |
| 35. | 2300 | 5 | - |  |
| 36. | 1790 | 3 |
| 37. | 1790 | 3 | ‘7’ |  | Информативная «пятёрка» символов |
| 38. | 2810 | 7 |
| 39. | 1790 | 3 | ‘6’ |  |
| 40. | 2555 | 6 |
| 41. | 1790 | 3 | ‘5’ |  |
| 42. | 2300 | 5 |
| 43. | 1790 | 3 | ‘4’ |  |
| 44. | 2045 | 4 |
| 45. | 1790 | 3 | ‘3’ |  |
| 46. | 1790 | 3 |
| 47. | 1790 | 3 | - | 32ASCII – символ ‘2’ | Синхронизация |
| 48. | 2300 | 5 |
| 49. | 4850 | f | - |
| 50. | 3830 | b |
| 51. | 4850 | f | - |
| 52. | 1535 | 2 |
| 53. | 1790 | 3 | ‘1’ |  | Продолжение некой информативной «пятёрки» символов |
| 54. | 1280 | 1 |
| 55. | 1790 | 3 | ‘0’ |  |
| 56. | 1025 | 0 |
| 57. | 1280 | 1 | DLE | Инструкция конца интересующего блока |
| 58. | 1025 | 0 |
| 59. | … | … | … | … |

# Реализация алгоритма декодирования протокола

Для автоматизации декодирования данных был написан скрипт на языке Python. Программа также использует библиотеки scipy, numpy для работы с данными и модуль wave для обработки .wavфайлов.