

Прием и обработка АРТ сигнала со спутников дистанционного зондирования Земли

Тимофеев Андрей
Лебедев Михаил
Капцевич Ольга

25 января 2024 г.

Содержание

Введение	2
Лаборатория спутникового мониторинга ДВФУ	2
APT (Automatic Picture Transmission)	2
RTL-SDR	2
Аппаратная реализация	3
Варианты архитектуры системы	3
Вариант №1	4
Вариант №2	5
Вариант №3	6
Реализация	7
Программная реализация	8
Автоматизация приема АПТ сигнала на RTL-SDR	8
Расчет времени приема спутника	8
Запись и сохранение радио сигнала на нужной волне во время пролета спутника	10
Декодирование принятого АПТ сигнала спутника NOAA-19 . .	11

Введение

Суть проекта заключается в создании станции (на основе RTL-SDR) по приему и обработки [APT \(Automatic Picture Transmission\)](#) сигнала со спутников дистанционного зондирования Земли на базе лаборатории спутникового мониторинга ДВФУ.

Станция представляет собой Антену SDR

Лаборатория спутникового мониторинга ДВФУ

APT (Automatic Picture Transmission)

Automatic Picture Transmission - это стандарт для аналоговой передачи данных (изображений) по радиоканалу с использованием специализированных радиопередатчиков, установленных на метеорологических спутниках.

Стандарт APT был разработан подразделением [NOAA National Earth Satellite Service](#) в 1960-х. Впервые был описан в статье “[The Automatic Picture Transmission \(APT\) TV camera system for meteorological satellites](#)”. Первый спутник на полярной орбите, передававший изображения в формате APT – TIROS-N.

RTL-SDR

Аппаратная реализация

Для приёма сигнала используются две антенны X-Quad на ??? и ??? МГц закреплённые на поворотном устройстве (TODO) на крыше корпуса G кампуса ДВФУ. Поворотное устройство управляет программно с компьютера, находящегося в кабинете G542 (на крышу протянуты провода).

TODO картинки антенн и контроллера поворота

В рамках проекта стояла задача разработать аппаратную систему, позволяющую принимать и обрабатывать радиосигналы с помощью программно определяемых радиосистем (RTL-SDR). Желаемые качества такой системы были следующие:

- **Возможность непрерывной работы.** Предполагается постоянное функционирование системы.
- **Возможность полной автоматизации.** Предполагается использование станции полностью в автоматическом режиме, включая поворот антенны, начало и конец записи.
- **Отказоустойчивость и простота починки.** Система должна быть достаточно простой, а также простой в ремонте при поломке.
- **Возможность дополнения и обновления.** В данный момент используются модули RTL-SDR, но впоследствии желательна возможность заменить их радиосистемами лучшего качества.
- **Низкая шумность.** Желательно избегать длинных линий передачи слабого аналогового сигнала от антенны к радиоприёмнику.

Руководствуясь этими критериями были разработаны 3 возможные архитектуры системы.

Варианты архитектуры системы

В каждом из вариантов архитектуры предполагается расположение части оборудования непосредственно на вращающейся части поворотного комплекса антенн в влагозащищённом контейнере. На схемах ниже эта часть представлена слева и схематически отделена от оборудования в помещении двумя косыми чертами.

Также в каждом из вариантов предполагается питание этого оборудования от источника постоянного тока расположенного в помещении. Для того, чтобы доставить стабильную мощность оборудованию, по линии питания подаётся напряжение 14В и используется регулятор для понижения напряжения до 5В.

Вариант №1

Наиболее предпочтительным вариантом представлялось использование удлинителя USB для подключения RTL-SDR через длинную цифровую линию непосредственно к компьютеру в помещении, на котором развернута программная часть системы.

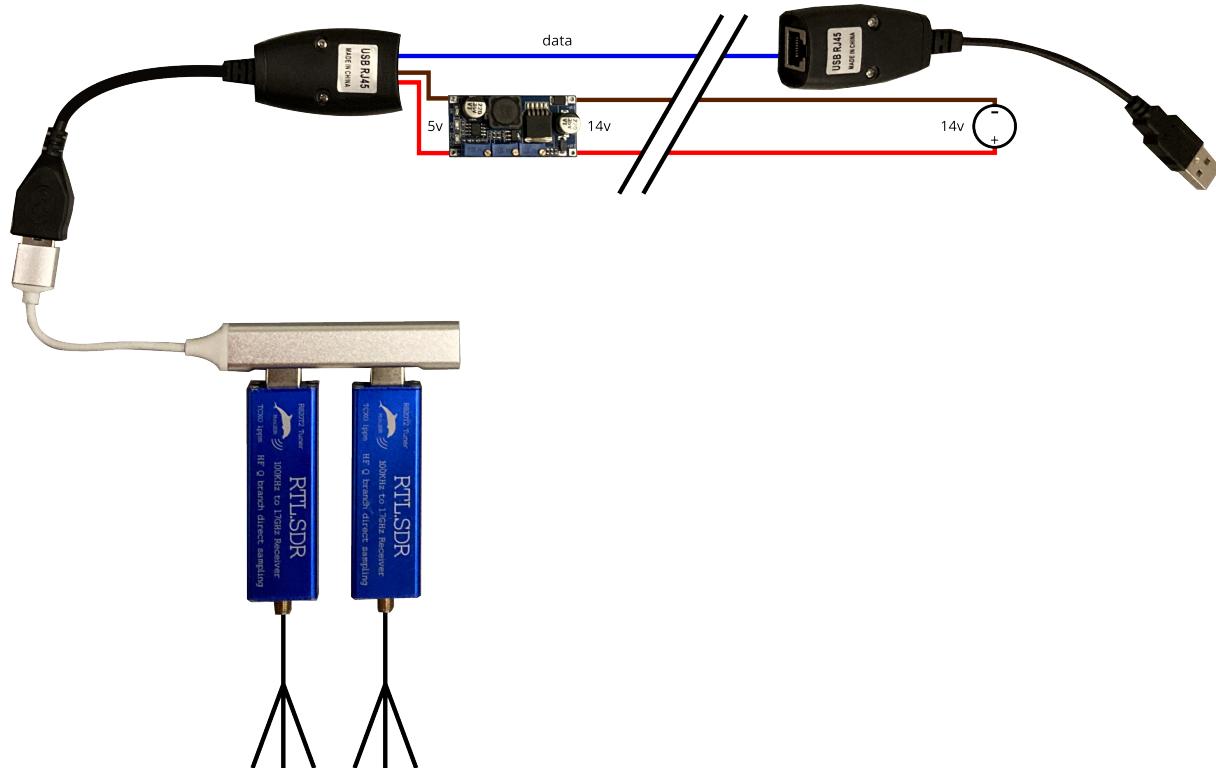


Рис. 1: Вариант архитектуры №1.

Такой вариант позволил бы уменьшить количество оборудования на крыше и его энергопотребление, упростив систему и увеличив отка- зоустойчивость системы по сравнению в вариантом №2. Однако, требовалось проверить, будет ли удлинитель USB предоставлять достаточную скорость передачи данных для стабильной работы системы.

Вариант №2

Другим возможным вариантом представлялось подключение RTL-SDR модулей к микрокомпьютеру Raspberry Pi, который также был бы установлен на крыше, и передачу цифровых данных на управляющий компьютер для декодирования и обработки используя Ethernet.

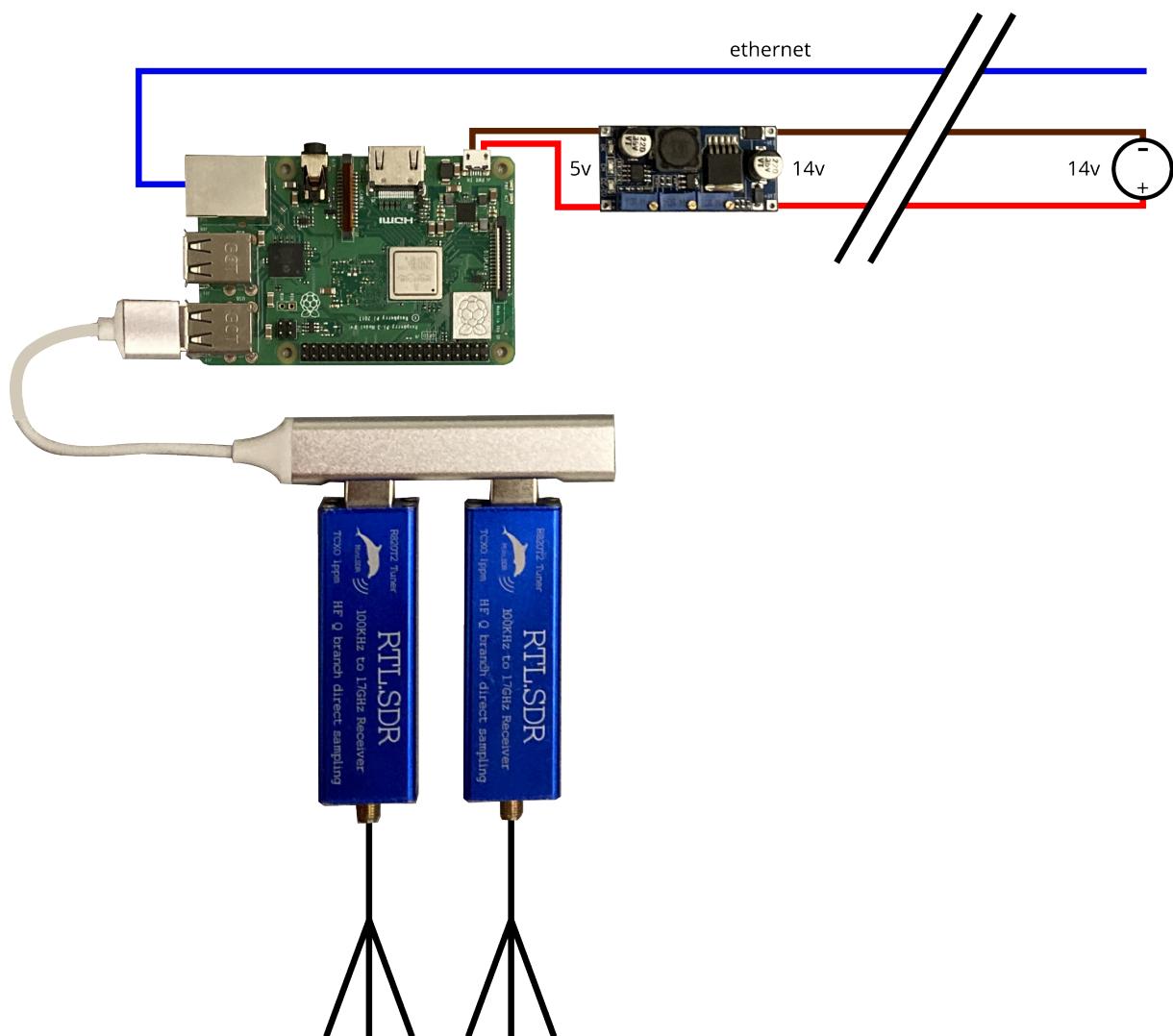


Рис. 2: Вариант архитектуры №2.

Такой вариант обеспечивал бы более высокую пропускную способность и устойчивость к магнитным помехам линии передачи данных. Однако, он требовал бы более высокой мощности питания и дополнительно потребовал бы дополнительного охлаждения, а также усложнил бы программную реализацию, отделив управляющий компьютер и радиоприёмники.

Вариант №3

Последним рассмотренным вариантом являлось расположение на крыше только предусилителей, и передачи аналогового сигнала, от них по коаксиальному проводу к радиоприёмникам, расположенным в помещении.

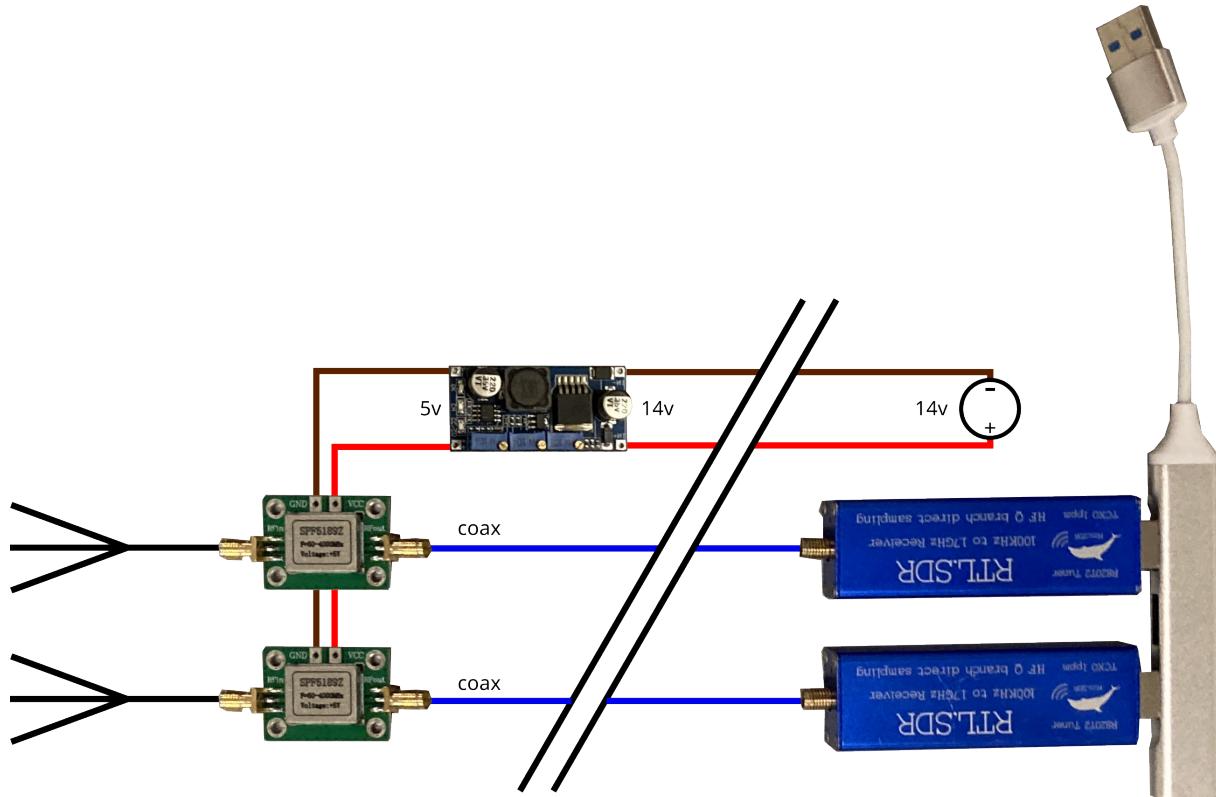


Рис. 3: Вариант архитектуры №3.

Этот вариант был бы наиболее простым в реализации, и обеспечивал бы более простую возможность ремонта и обновления системы. Однако, он был бы наиболее подвержен помехам в виду передачи аналогового а не цифрового сигнала, что дополнительно усугублялось бы возможными наводками по линии питания от инверторного понижающего регулятора.

Реализация

Было проведено тестирование работоспособности вариантов №1 и №2 для необходимой длины линий питания и данных. Было обнаружено, что оба варианты архитектуры обеспечивают достаточное питание, однако удлинителю USB не располагают достаточной шириной канала для работы с RTL-SDR. В силу этого решено было использовать вариант архитектуры №2. Вариант №3 не был протестирован ввиду того, что у нас не было коаксиального кабеля необходимой длины.

В данный момент система собрана, протестирована, и готова к влагоизоляции и установке на крышу, что будет сделано как только нами будет получено соответствующее разрешение от вуза.

Программная реализация

Автоматизация приема АРТ сигнала на RTL-SDR

Для автоматизации приема АРТ сигнала был реализован скрипт на языке программирования Python.

Автоматизация заключается в следующем:

- Расчет времени приема спутника.
- Запись и сохранение радио сигнала на нужной волне во время пролета спутника.

Расчет времени приема спутника

Что бы узнать когда мимо нас пролетает конкретный спутник, можно обратиться к уже готовым сервисам предоставляющих эту информацию и забирать её от туда. Таким сервисом стал [N2YO](#).

N2YO - это сервис на котором можно отслеживать положение спутников на орбите в реальном времени, получать о спутниках справочную информацию, а так же предсказывать время пролета спутника в конкретной точке мира, что нам и интересно. Для обмена этой информации у сервиса N2YO есть [API](#).

Для получения этой информации автоматически, была написана функция `get_start_record_time` которая обращается к API сервиса N2YO и возвращает ближайшее время пролета выбранного нами спутника.

```
CONFIG_PATH = 'config.ini'

# N2YO API -> https://www.n2yo.com/api/
def get_start_record_time(satellite_id: str,
                           lat: str,
                           lon: str,
                           days: int
) -> tuple[datetime.datetime, datetime.datetime]:
    """
    Time of passage of the specified satellite in UTC + 10h

    :param satellite_id: satellite id in n2yo
    :param lat: observer's latitude
    :param lon: observer's longitude
    :param days: number of days of prediction (max 10)

    :return: satellite flyby start time and satellite
            flyby end time
    """

```

Полную реализацию функции, можно найти [тут](#).

Запись и сохранение радио сигнала на нужной волне во время пролета спутника

Запись сигнала осуществлялась на RTL-SDR. Для взаимодействия Python с RTL-SDR использовалась библиотека [SoapySDR](#).

SoapySDR — это набор кроссплатформенных библиотек, написанных на C++, предоставляющих унифицированный доступ к SDR-устройствам. Так же для этой библиотеки существует [оболочка на Python](#), которая и была использована.

Функция *sdr_record* для записи и сохранения радио сигнала на выбранной частоте.

```
def sdr_record(device,
                frequency,
                sample_rate,
                gain,
                blocks_count
) -> None:
    """
    Receiving and recording a radio signal

    :param device: sdr device object
    :param frequency: value of frequency
    :param sample_rate: value of sample rate
    :param gain: value of gain
    :param blocks_count: number of record blocks
    """
```

Полную реализацию функции, можно найти [тут](#).

Декодирование принятого АРТ сигнала спутника NOAA-19

Для превращения сигнала от спутника NOAA-19 в изображение реализован класс *NOAADecoder*.

```
class NOAADecoder:  
    def __init__(self,  
                 black_point: int,  
                 white_point: int,  
                 components: Optional[List[str]],  
                 ) -> None:  
        ...  
  
    Class for decode NOAA APT data  
  
    :param black_point: dynamic range lower bound,  
                       percent  
    :param white_point: dynamic range upper bound,  
                       percent  
    :param components: portions of the image to  
                      preserve/filter  
    ...
```

Полную реализацию функции, можно найти [тут](#).

Декодирование сигнала в изображение происходит следующим образом:

- Передискретизация до частоты 20800 Гц.
- Преобразование в аналитический сигнал ([Преобразование Гилберта](#)).
- Нормирование значений сигнала от 0 до 255.
- Нахождение положения кадров синхронизации сигнала АРТ.
- Выбор сектора для формирования изображения.