

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ  
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ФАКУЛЬТЕТ ПРИКЛАДНОЙ МАТЕМАТИКИ И ИНФОРМАТИКИ**

**Кафедра компьютерных технологий и систем**

**ГОЛОСОВОЕ УПРАВЛЕНИЕ РОБОТОТЕХНИЧЕСКИМИ  
КОМПЛЕКСАМИ**

Курсовой проект

Жуковского Павла Сергеевича  
Студента 3 курса, 12 группы  
Специальность  
«прикладная информатика»  
Научный руководитель:  
Ассистент кафедры КТС  
Грозов И.А.

Минск, 2020

# ЦЕЛЬ И ЗАДАЧИ ПРОЕКТА

## Цель:

Целью данного проекта является разработка способа, позволяющего обеспечить робототехнические комплексы возможностью выполнять свои функции с помощью голосовых команд, чтобы облегчить работу с этими комплексами в тех ситуациях, когда проще дать голосовую команду, нежели прописывать её в терминале.

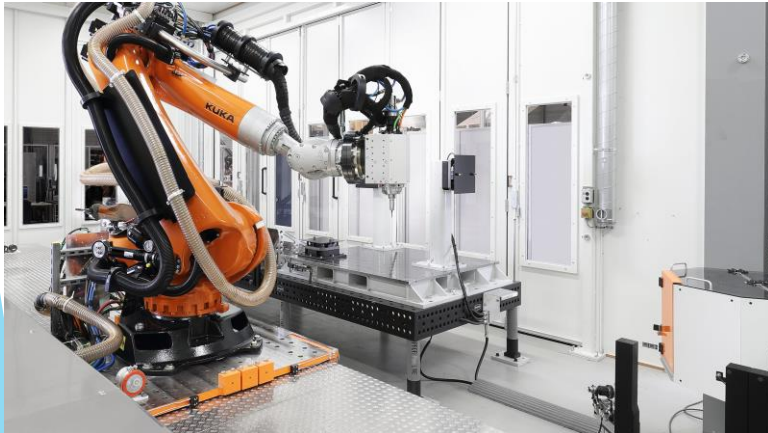
## Задачи:

- 1) Изучение общих сведений о робототехнических комплексах, их конструкции и способах взаимодействия с ней.
- 2) Изучение теории, связанной с технологией обработки голоса, а также библиотек, позволяющих сохранять данные голосовых команд в программе.
- 3) Реализация программы, подключающейся к робототехническому комплексу, которая будет обрабатывать голос и пересылать полученные команды.
- 4) Тестирование программы на реальном роботе

# РОБОТОТЕХНИЧЕСКИЕ КОМПЛЕКСЫ

**Робот** (чеш. robot, от robota - барщина, подневольный труд, rob – раб, слово придумано чешским писателем К. Чапеком и значило у него «искусный в работе человек») - машина с антропоморфным (человекоподобным) поведением, которое частично или полностью выполняет функции человека (иногда животного) при взаимодействии с окружающим миром.

**Робототехнические комплексы (РТК)** - представляют собой совокупность промышленных роботов, технологического оборудования и средств оснащения, обеспечивающих комплексную автоматизацию технологических процессов в многономенклатурном производстве.



# ВЗАИМОДЕЙСТВИЕ С РОБОТОМ

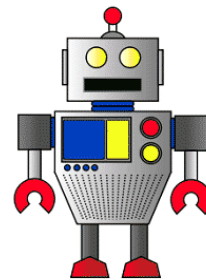
**Pulse Robot Python API** – библиотека, реализованная в языке программирования Python, позволяющая подключаться к реальным робототехническим комплексам и, используя различные функции и методы, отдавать этим комплексам необходимые команды.

## Установка

```
pip install pulse-api
```

## Инициализация API

```
from pulseapi import RobotPulse  
# create an instance of the API wrapper class  
host = "http://127.0.0.1:8081" # replace with a valid robot address  
robot = RobotPulse(host)
```



**Python Robot  
Framework**

# РАСПОЗНАВАНИЕ РЕЧИ

Современные системы распознавания речи могут распознавать речь от нескольких носителей и имеют огромный словарный запас на разных языках. Речь должна быть преобразована из физического звука в электрический сигнал с помощью микрофона, а затем в цифровые данные можно использовать для *транскрибирования* аудио в текст.



LEARN HOW TO CREATE SPEECH RECOGNITION  
SYSTEM IN PYTHON

**in 10 Minutes**



- SpeechRecognition – библиотека, позволяющая распознавать аудио-файлы
- PyAudio – библиотека, позволяющая распознавать живую речь прямо с микрофона



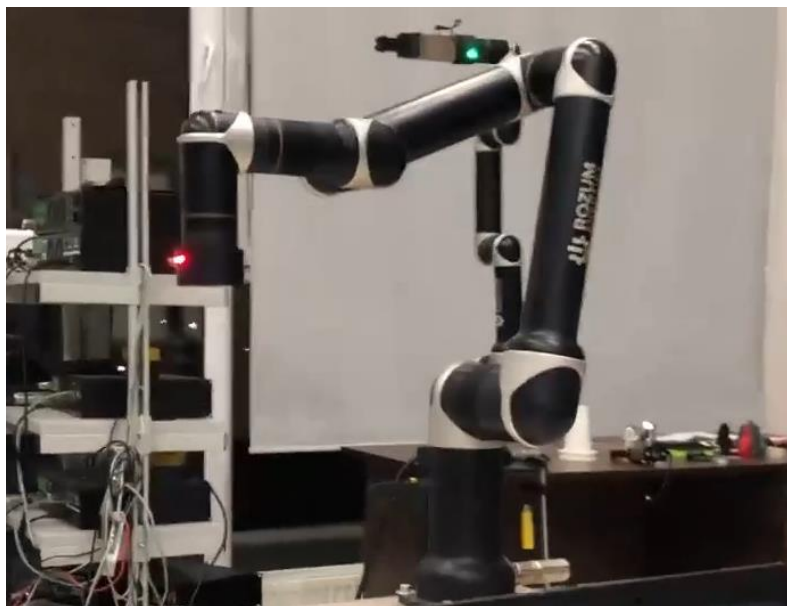
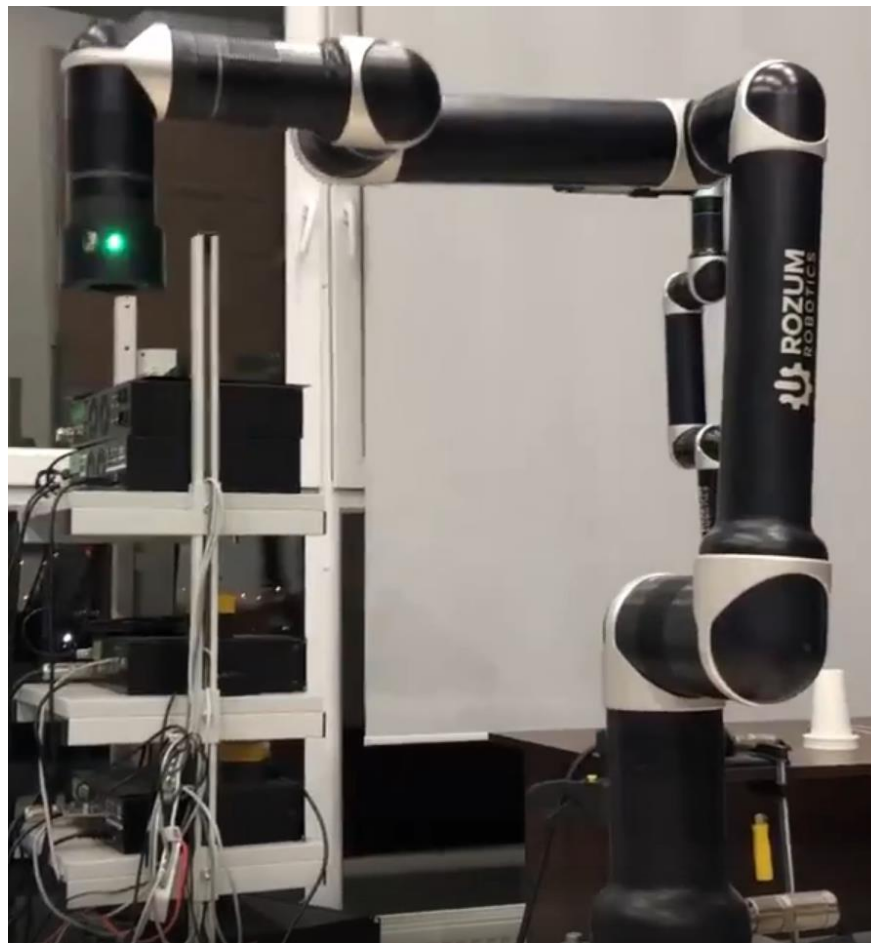
# РЕАЛИЗАЦИЯ ПРОГРАММЫ

- ❑ Для реализации программы, позволяющей обращаться к роботу, я использовал среду разработки **PyCharm**, внутри которой написал программу на языке программирования **Python**.
- ❑ В своей программе я подключил такие библиотеки, как: **pulseapi**, **speech recognition**, **pyaudio** и некоторые другие.
- ❑ Далее я создал объект распознавателя в программе и привязал к нему робота с помощью IP-адреса и сети. Таким образом, робот мог получать сигналы от программы.
- ❑ Для некоторых функций робота я определил определённые команды (строки), таким образом, чтобы робот выполнил эти функции в те моменты, когда ему будет дана соответствующая голосовая команда.

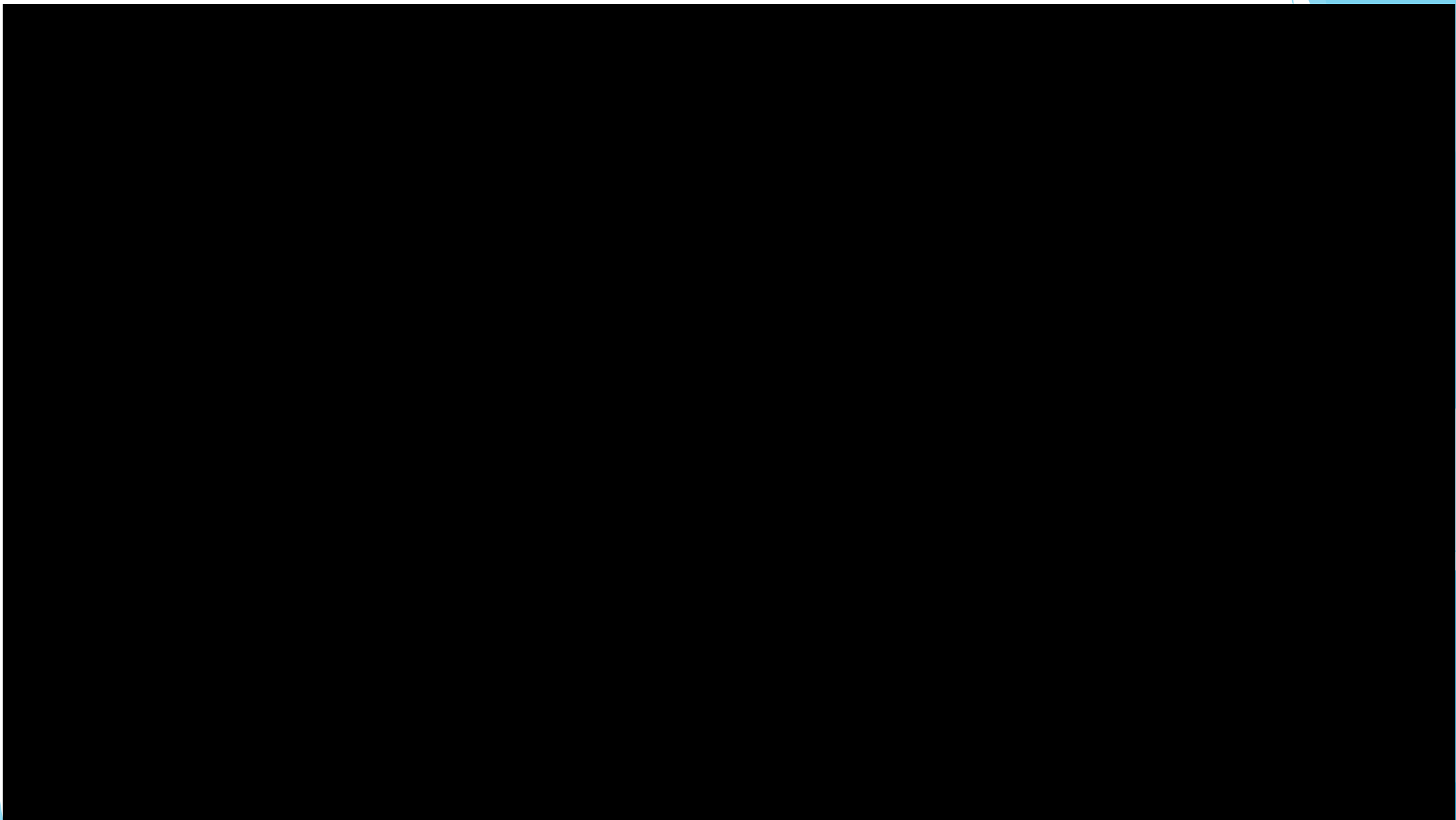
```
robot = RobotPulse("http://192.168.1.52:8081")  
  
# Initialize the recognizer  
r = sr.Recognizer()
```

```
ref = {  
    "stop": stop,  
    "activate": activate,  
    "circle": circle,  
    "home": home,  
}
```

# СОСТОЯНИЯ РОБОТА



# ДЕМОНСТРАЦИЯ РАБОТЫ РОБОТА





# ЗАКЛЮЧЕНИЕ

Таким образом, я представил способ, с помощью которого можно привязать определённые голосовые команды к промышленному роботу с помощью библиотек языка Python.

Данное исследование может очень эффективно применяться на практике, так как люди, работающие с робототехническими комплексами, могут привязать определённый список голосовых команд к каким-либо конкретным задачам для робота и использовать их без нужды в терминале и устройств для ввода.

# СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1) Робот-манипулятор серии Pulse
- 2) Руководство пользователя для робота-манипулятора серии Pulse
- 3) Corke P. Robotics, Vision and Control, Springer Tracts in Advanced Robotics, 2017 – 693 p. – ISBN: 978-3-319-54413-7
- 4) Saudabayev, A., Varol, H.A.: Sensors for robotic hands: a survey of state of the art, 2015 – 17p. - doi: 10.1109/ACCESS.2015.2482543.