**МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРСЬ**

**БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**

**ФАКУЛЬТЕТ ПРИКЛАДНОЙ МАТЕМАТИКИ И ИНФОРМАТИКИ**

**Кафедра компьютерных технологий и систем**

**ГОЛОСОВОЕ УПРАВЛЕНИЕ РОБОТОТЕХНИЧЕСКИМИ КОМПЛЕКСАМИ**

Курсовая работа

Жуковского Павла Сергеевича

Студента 3 курса,

Специальность

«прикладная информатика»

Научный руководитель:

Ассистент кафедры КТС

Грозов И.А.

Минск, 2021

**Оглавление**

[ВВЕДЕНИЕ 3](#_Toc72086298)

[Глава 1. Робототехника 4](#_Toc72086299)

[1.1 Робототехнические комплексы 4](#_Toc72086300)

[1.2 Конструкции промышленных роботов 7](#_Toc72086301)

[1.3 Гибкие автоматизированные производства 11](#_Toc72086302)

[1.4. Структура гибких производственных систем 14](#_Toc72086303)

[Глава 2. Использование технологий обработки голоса для робототехники на практике 18](#_Toc72086304)

[2.1 Библиотеки SpeechRecognition и PyAudio 18](#_Toc72086305)

[2.2 Технология RobotPulse 19](#_Toc72086306)

[2.3 Опыт с реальным роботом 19](#_Toc72086307)

[ЗАКЛЮЧЕНИЕ 22](#_Toc72086308)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 23](#_Toc72086309)

[ПРИЛОЖЕНИЕ А 24](#_Toc72086310)

# ВВЕДЕНИЕ

С приходом информационной эпохи наш мир постепенно развивается и совершенствуется. В последнее время было изобретено множество различных технологий и изобретений, которые во многом улучшили качество нашей жизни в разных аспектах.

В частности, в последнее время очень активно развиваются робототехнические комплексы. Эти технологии позволяют с помощью роботов выполнять различные промышленные задачи, такие как, например, задачи распознавания объектов, создание диалога между людьми, помощь в промышленном производстве, математические сизические расчёты и многие другие.

В теме своей курсовой работы я бы хотел поговорить об этих робототехнических комплексах, а также о том, как с ними можно взаимодействовать, какую выгоду и пользу с них можно получить, как можно улучшить производительность этих комплексов, как их можно модифицировать.

В последнее время очень популярным стало взаимодействие техники и людей при помощи голосовых сообщений и команд. Очень часто на практике возникает такая ситуация, когда некогда или неудобно отдавать команды какому-либо робототехническому комплексу с помощью команд через терминал, поэтому возможность управления подобными комплексами с помощью голоса имеет очень важное и полезное значение. В добавок, в последнее время в различных средах разработки и языках программирования были добавлены библиотеки, позволяющие обрабатывать голосовые сообщения и сигналы, после чего отправлять их робототехническим комплексам.

В своём курсовом проекте я бы хотел обсудить возможность внедрения подобных технологий в робототехнические комплексы, а также то, какую практическую пользу из этого можно вынести.

В данной работе будет отмечена не только теория касаемо робототехнических комплексов, а также взаимодействия с ними с помощью голосовых команд, но и некоторые практические результаты исследования и выводы.

# Глава 1. Робототехника

## 1.1 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.

По мере развития техники роботы потеряли внешнее сходство с челове­ком и превратились в различные автоматические устройства и механизмы, которые освобождают людей от тяжелой и однообразной работы, от работ, опасной для здоровья (в условиях повышенной радиации, высокой или низкой температуры) и в труднодоступных местах (под водой, в космосе).

Такие устройства механизмы называются роботами.

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

В качестве основного технологического оборудования могут использоваться станки, прессы, сварочные автоматы и промышленные роботы. Средствами оснащения (или вспомогательным технологическим оборудованием) могут быть устройства накопления и ориентации деталей, магазины для хранения сменных захватов роботов, обеспечивающих нормальное протекание основного технологического процесса.

Состав оборудования в РТК строго не регламентируется и определяется его функциональным назначением.

По назначению и решаемому классу задач все роботы мо­гут быть разделены на две большие группы: промышленные и исследовательские.

**Промышленные** роботы (ПР) роботы, предназначенные дня выпол­нения тяжелой, монотонной, вредной и опасной для здоровья физической работы, а также для выполнения отдельных видов трудоемкой, напряжен­ной и утомительной умственной работы (проектирование, информацион­ное обеспечение, управление).

Соответственно конкретным областям применения имеется ряд раз­новидностей промышленных роботов.

Промышленные роботы, получившие наибольшее развитие в настоящее время, предназначены для автоматизации основных и вспомогательных операций в различных отраслях промышленности: в машиностроении и приборостроении, в горно-добывающей, нефтехимической, металлургической, атомной и др.

Промышленные роботы подразделяются на три группы по производственно-технологическим признакам:

* производственные, или технологиче­ские (ППР), для основных операций технологических процессов;
* подъемно-транспортные, или вспомогательные (ПТПР), выполняющие действия типа взять – перенести – положить
* универсальные (УПР) для различных операций – основных и вспомогательных

По специализации промышленные роботы подразделяются на:

* специальные выполняющие строго определенные технологические операции или обслуживающие конкретные модели технологического оборудования;
* специализированные, или целевые, предназначенные для выполнения технических операций одного вида (сварки, сборки, окраски и т.п.) или обслуживания определенной группы моделей технологического оборудования, объединенных общностью манипуляционных действий;
* универсальные, или многоцелевые, ориентированные на выполнение как основных, так и вспомогательных технологических операций различных видом и с различными группами, моделей технологического оборудования.

**Исследовательские** роботы – это роботы, предназначенные для поиска, сбора, переработки и передачи информации об исследуемых объектах. Такими объектами могут быть труднодоступные или недоступные для человека сферы (космическое пространство, океанские глубины, недра Земли, экстремальные лабораторные условия) либо области, где требуются выявление, переработка и анализ огромной по объему информации, на пример информационный поиск и разведка, искусство и литература.

Основными факторами, определяющими целесообразность применения РТК являются:

* вредные, физически тяжелые и опасные для жизни ручные операции, механизация и автоматизация которых традиционным методом невозможна;
* погрузочно-разгрузочные и другие работы, выполнение которых ограничено быстродействием движения рук рабочего и быстрой его утомляемостью;
* необходимость повысить качество изготовляемых изделий, увеличить объем их выпуска, сократить затраты времени на их изготовление за счет высоких технических показателей РТК;
* снижение уровня производственного травматизма и профессиональных заболеваний.

Значительная часть промышленных роботов (ПР) выполняет основные технологические операции дуговой и контактной сварки, сборки, окраски поверхности и др.

Автоматизируя выполнение вспомогательных операций, промышленные роботы объединяют оборудование в единые системы, обладающих гибкостью, универсальностью, быстротой переналаживаемости.

Существуют следующие схемы обслуживания промышленными роботами технологического оборудования.

* один ПР обслуживает одну единицу оборудования;
* один ПР обслуживает несколько единиц оборудования;
* несколько ПР обслуживают одну единицу оборудования, когда на нем выполняется несколько операций;
* несколько ПР одновременно обслуживают несколько единиц технологического оборудования.

Производительность РТК – это количество изготовленных изделий за определенный период времени. Различают цикловую и сменную производительность РТК.

**Цикловая производительность РТК** определяется по формуле:

https://ok-t.ru/helpiksorg/baza5/120021231886.files/image428.gif , (1)

где https://ok-t.ru/helpiksorg/baza5/120021231886.files/image430.gif ‑ количество изделий, изготовленных за один цикл РТК;

https://ok-t.ru/helpiksorg/baza5/120021231886.files/image432.gif ‑ продолжительность одного цикла РТК, https://ok-t.ru/helpiksorg/baza5/120021231886.files/image181.gif ;

**Сменная производительность РТК** определяется по формуле:

https://ok-t.ru/helpiksorg/baza5/120021231886.files/image435.gif , (2)

где https://ok-t.ru/helpiksorg/baza5/120021231886.files/image437.gif ‑ продолжительность смены в https://ok-t.ru/helpiksorg/baza5/120021231886.files/image181.gif ;

https://ok-t.ru/helpiksorg/baza5/120021231886.files/image440.gif ‑ коэффициент, учитывающий потери времени на техническое и организационное обслуживание, переналадку оборудования.

Универсальность РТК характеризуется возможностью изготовления изделий, различных по назначению, но имеющих конструктивно – технологическое сходство, а также переходом от изготовления одного вида изделия к другому с небольшими затратами подготовительно – заключительного времени.

## 1.2 Конструкции промышленных роботов

Промышленный робот (IIP) состоит из исполнительного устройства (собственно манипулятора) и устройства управления.

Манипулятор IIP предназначен для выполнения всех его двигатель­ных функций и представляет собой многозвенный механизм с разомкну той кинематической цепью, оснащенный приводами и рабочим органом, а также в общем случае устройством передвижения. Манипулятор состо­ит из опорных (несущих) конструкций, манипуляционной системы, рабочих органов, привода и устройства передвижения.

Устройство управления ПР необходимо для формирования и выдачи управляющих воздействий манипулятору в соответствии с управляющем программой. Оно состоит из собственно системы управления, информа­ционно-измерительной системы с устройствами обратной связи и систе­мы связи.

Структурная схема промышленного робота представлена на рис. 1.

. 

Рис. 1. Структурная схема промышленного робота

Опорные конструкции служат для размещения всех устройств и агре­гатов IIP, а также для обеспечения необходимой прочности и жесткости манипулятора. Опорные конструкции выполняются в виде оснований, корпусов, стоек, рам тележек, порталов и т.п. (рис. 2.).

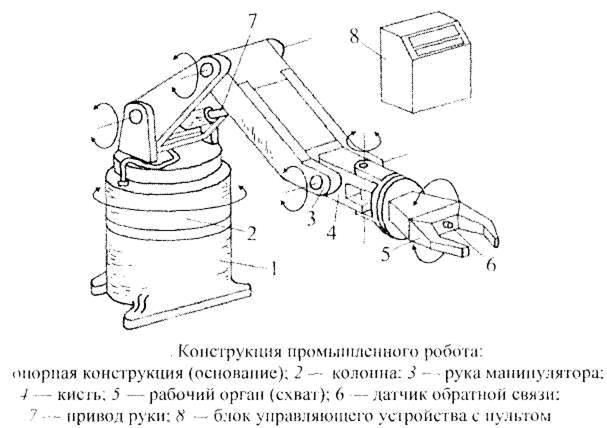


Рис. 2. Конструкция промышленного робота

Манипуляционная система предназначена для переноса и ориентации рабочего органа или объекта манипулирования в заданной точке рабочей зоны и представляет собой многозвенный пространственный механизм с разомкнутой кинематической цепью.

Рабочий орган манипулятора ПР, необходимый для непосредственно­го воздействия на объект манипулирования при выполнении технологи­ческих операций или вспомогательных переходов, представляет собой захватное устройство или рабочий инструмент.

Привод предназначен для преобразования подводимой энергии в ме­ханическое движение исполнительных звеньев манипулятора в соответ­ствии с командными сигналами, поступающими от системы управления и в общем виде содержит энергоустановку, двигатели и передаточные устройства.

Устройство передвижения служит для перемещения манипулятора или ПР в целом в необходимое место рабочего пространства и состоит из ходовой части и приводных устройств.

Система управления необходима для непосредственного формирования и выдачи управляющих сигналов и состоит из пульта управлении, запоминающего устройства, вычислительного устройства, блоков управления приводами манипулятора и технологическим оборудованием.

Информационно-измерительная система, предназначенная для сбора и первичной обработки информации для системы управления о состоянии элементов и механизмов ПР и внешней среды, входит в состав устройства управления IIP и включает в себя устройство обратной связи, устройство сравнения сигналов и датчики обратной связи.

Систему связи используют для обеспечения обмена информацией между ПР и оператором или другими роботами и технологическими устройствами с целью формулировки заданий, контроля за функционированием систем IIP и технологического оборудования, диагностики неисправностей, регламентной проверки и т. п.

Манипуляционная система (МС) промышленного робота является составной частью манипулятора IIP, обеспечивающей перенос и ориентацию рабочего органа или объекта манипулирования в заданной точно пространства и определяющей форму и объем рабочей зоны ПР, а также характер движений рабочего органа. В совокупности с опорной конст­рукцией, приводом, передаточными механизмами и рабочим органом манипуляционная система образует манипулятор ПP, при этом часть элементов опорной конструкции, привода и передаточных механизмом может непосредственно входить в состав манипуляционной системы и качестве ее звеньев, что обусловливает применения в робототехнике об­щего понятия «манипулятор», как в отношении собственно манипулятора ПР, так и его манипуляционной системы.

На рис. 3. представлен один из типов манипулятора.

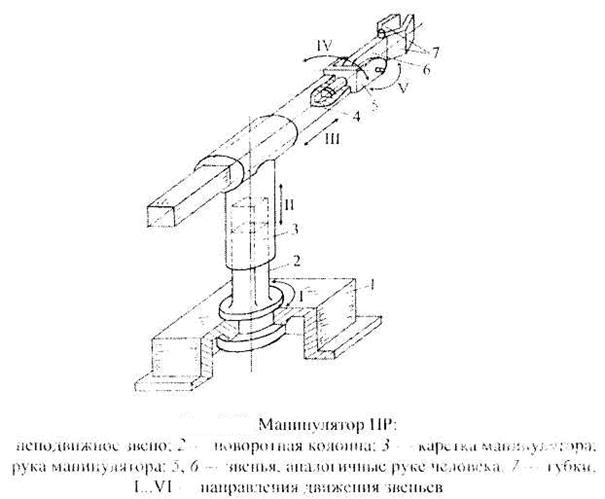


Рис. 3. Манипулятор ПР

В рассмотренном манипуляторе движения колонны, каретки и руки и направлениях I, II и III являются переносными, обеспечивающими пере­мещения рабочего органа или объекта манипулирования в заданное место рабочей зоны ПР; перемещение кисти захвата в направлениях IV и V ориентирующими, необходимыми для ориентации рабочего органа или объекта манипулирования. Отдельные движения элементов рабочею орга­на, например, в направлении VI зажатия-разжатия губок захвата, относятся к внутренним, поскольку не изменяют пи положения рабочего органа в ра­бочей зоне, ни его ориентации

## 1.3 Гибкие автоматизированные производства

**Гибкие автоматизированные производства (ГАП)** – представляет собой сложные технико – организационные системы, содержащие оборудование с ЧПУ, робототехнические комплексы, обрабатывающие центры, микропроцессорную технику, автоматизированные склады и транспортные системы.

**Гибкий производственный модуль (ГПМ) – это** единица технологического оборудования, автоматически осуществляющая технологические операции в пределах ее технологических характеристик, способная работать автономно и в составе гибких производственных систем или ячеек. В ГПМ входят устройства: ЧПУ, адаптивного управления, контроля и измерения, диагностики.

**Гибкая производственная ячейка (ГПЯ) –** управляемая средствами вычислительной техники совокупность нескольких ГПМ и систем обеспечения функционирования, способная работать автономно и в составе гибкой производственной системы при изготовлении изделий в пределах подготовленного запаса заготовок и инструмента. В систему обеспечения функционирования **ГПЯ**входят автоматизированная система управления технологическим процессом, автоматизированная система управления технологическим оборудованием, автоматизированная транспортная, складская система, система автоматического контроля, автоматизированная система инструментообеспечения, автоматизированная система удаления отходов и др.

**Гибкая производственная система (ГНС) –** управляемая средствами вычислительной техники совокупность технологического оборудования, состоящая из разного сочетания ГПМ и/или ГПЯ, автоматизированной системы технической подготовки производства и системы обеспечения его функционирования, обладающая возможностью автоматизированной переналадки при изменении программы производства изделий, разновидности которых ограничены технологическими возможностями оборудования.

Внедрению ГАП должны предшествовать:

* повышение уровня технологического проектирования на основе САПР;
* создание программируемой технологии основных и вспомогательных производственных процессов, а также процессов управления информацией;
* обеспечение сопряженности программ управления и быстрой переналадки компонентов ГАП.

Из основных и вспомогательных гибких производственных модулей комплектуются гибкие автоматические линии, участки, цеха и заводы.

ГАП первого поколения были созданы на базе многооперационных станков типа обрабатывающий центр. За основу был принят блочно – модульный принцип. Гибкий производственный модуль (ГП - модуль) представляет собой, например, совокупность токарных станков с ЧПУ, специализированных роботов ‑ автооператоров и накопителей заготовок.

ГП – это модуль, представляющий собой совокупность основного технологического оборудования с ЧПУ, специализированных роботов, и накопителей заготовок.

На уровне поточной линии, участка ГАП могут состоять из ГП – модулей подготовки производства, включающий обеспечение материалами, заготовками, деталями, приспособлениями, инструментами; из модулей удаления отходов производства.

На уровне цеха ГАП включает гибкие автоматизированные участки и линии основного производства, АСУ технологической подготовкой производства, автоматизированные участки комплектования, складирования и технического обеспечения основного производства.

ГАП обладает следующими преимуществами:

* высокая производственно-технологическая гибкость, обеспечиваемая связью всех модулей в единый производственный комплекс с помощью АСУ технологическими процессами;
* постоянная мобильность производства, достигаемая компактной планировкой оборудования, связью модулей технологического оборудования через автоматические накопители, а также тем, что при смене изделия не всегда требуется переналадка оборудования;
* сокращение производственного цикла изготовления изделия и повышение производительности труда.

Главная особенность ГПС по сравнению с прежними формами организации производства, обеспечивающая высокую производительность (например, автоматические линии), возможность производства целой группы изделий в произвольном (и соответствии с требованиями дня) порядке и не большими партиями, причем такая организация производства не оказывает значительного влияния на его экономические показатели. В настоящее врем и под ГПС понимают системы, включающие средства производства, которые характеризуются легкостью переналадки и адаптации к изменяющимся условиям производства. Цель использования ГПС достижение эффективности производства, сравнимой с эффективностью массового производства, но для небольших партий разнородных деталей. Наиболее часто ГПС рассматривают как комплексную систему технологических машин (в большинстве случаев металлорежущих станков), транспортных средств, оснастки и прочих средств производства, которые полностью управляются с помощью компьютера. Станки в этом случае укомплектованы оснасткой для смены заготовок и режущих инструментов, что обеспечивает возможность обработки различных деталей без потерь времени на переналадку станка.

В настоящее время ГПС трактуют как способ организации производства, обеспечивающий полное управление производственным процессом и рационализирующий тот процесс. Такой подход означает необходимое и анализа каждой составляющей производства продукции на данном предприятии и тщательной временной стыковки с другими составляющими.

Использование ГПС обеспечивает:

* гибкость выбора различных заготовок для обработки в течение определенного времени;
* возможность добавления или удаления конкретной заготовки из разработанного ранее производственного задания на обработку деталей;
* Гибкость технологического маршрута, т.е. возможность замены станка для обработки конкретной детали, например, в случае изменения производственного задания или отказа станка;
* возможность быстрою внедрения в производство конструктивных изменений в обрабатываемых деталях;
* возможность изменений в программе выпуска конкретных деталей;
* возможность производства различных деталей в разных ГПС в рамках одного предприятия.

Возможности ГПС весьма привлекательны для лидеров современной промышленного производства. С одной стороны, они позволяют объединить высокую производительность с малыми объемами партий деталей или изделий, с другой внедрить безлюдные технологии и значительно снизить производственные расходы на заработную плату, что дает предприятию возможность быстрее, чем когда-либо, реагировать на требования рынка.

Известно, что в структуре времени обработки деталей на основе традиционных технологии доля основного времени составляем только около 30%, а оставшаяся часть приходится на вспомогательное и подготовительно- заключительное время. Автоматизация в ряде случаев позволяет сократить их на 80%.

Основные достоинства ГАП:

* возможность быстрой реакции предприятия на требования внутренних и зарубежных рынков;
* повышение и сохранение качества выпускаемой продукции;
* повышение производительности труда, в первую очередь в результате использования станков с высокой концентрацией операций;
* увеличение действительного фонда времени работы машин за счет работы в три смены, работы во время отпусков и болезней персонала;
* уменьшение численности обслуживающего персонала, что приводит к снижению затрат на заработную плату и социальные отчисления;
* значительное улучшение условий труда, в первую очередь за счет исключения тяжелой, неквалифицированной работы;
* возможность снижения себестоимости продукции.

## 1.4. Структура гибких производственных систем

Структурой **ГПС** называется сеть связей между ее элементами, которые могут рассматриваться как в статике (конфигурация и конструкция) так и в динамике (изменения во времени). Каждая ГПС представляет комплекс подсистем, показанных на рис. 4.

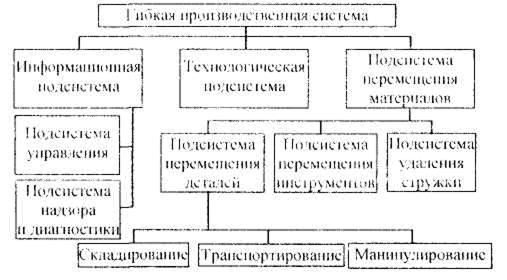


Рис. 4. Функциональные подсистемы ГПС

Взаимодействие функциональных подсистем ГПС показано на рис. 5.

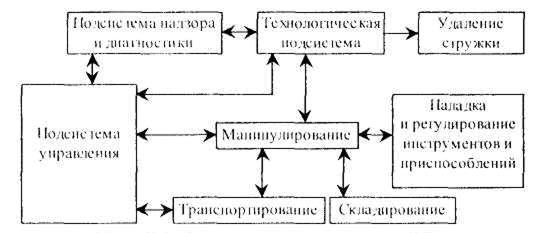


Рис. 5. Взаимодействие подсистем ГПС

**Технологическая подсистема**ГПС включает основные технологиче­ские устройства, использование которых определяется типом обрабаты­ваемых деталей и технологией их обработки. К таким устройствам отно­сятся машины и приспособления на соответствующих РМ, с помощью которых реализуются операции технологических процессов обработки, сборки, а также операции мойки, чистки и т.д.

**Информационная подсистема** на локальном уровне состоит из под­системы управления и подсистемы надзора и диагностики. Она обеспе­чивает возможность функционирования ГПС без непосредственного уча­стия оператора. Подсистема управления может содержать центральный (главный) компьютер и автономные подсистемы для управления работой каждой из составляющих подсистем. Подсистема надзора и диагностики обеспечивает проверку, как корректности работы машины, так и точности обработанных де талей.

Движение информации в ГПС имеет иерархическую структуру, в ко­торой выделяются три уровня принятия решений (рис. 6.). В дальней­шем будет рассматриваться только низший (текущий) уровень.

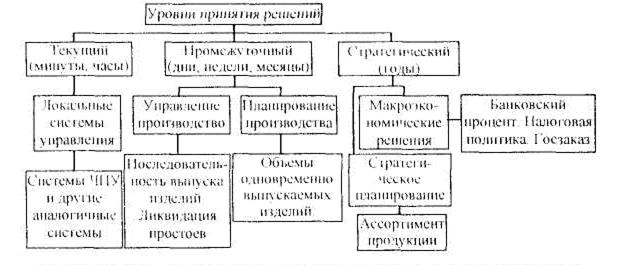


Рис. 6. Уровни принятия решения и управления предприятием

**https://ok-t.ru/helpiksorg/baza5/120021231886.files/image061.gif Подсистема перемещения материалов** включает технические средства обеспечивающие связи между элементами производственной подсистемы и управляет перемещениями инструментов, деталей и приспособлений для их закрепления. Эта подсистема обеспечивает складирование, транспортирование соответствующих элементов и манипулирование ими. *Складирование* заключается в хранении обрабатываемых изделий до и после обработки, а также в обеспечении требуемых перерывов в перемещения инструментов для нормальной работы технических устройств. *Транспортирование* служит для изменения местоположения (перемещения) соответствующих элементов. Манипулирование обеспечивает перемещение одновременной сменой ориентации, например, перемещение детали в рабочую зону станка. Рассмотренные действия могут реализовываться независимо друг от друга либо быть объединены. Последнее перспективнее, и хотя требует разработки более сложной системы управления. Наиболее часто встречается объединение двух функций: транспортирования и манипулирования или транспортирования и складирования.

Разделение ГПС на подсистемы касается производственных ячеек высокого уровня. На нулевом уровне практически все рассмотренные функции выполняет оператор, обслуживающий данное РМ.

# Глава 2. Использование технологий обработки голоса для робототехники на практике

## 2.1 Библиотеки SpeechRecognition и PyAudio

Для реализации своей задачи, связанной с привязкой голосовых команд к реальному роботу, я воспользовался технологиями языка программирования Python. В своей реализации я использовал такие библиотеки, как SpeechRecognition и PyAudio.

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.

В PyPI существует несколько пакетов для распознавания речи. Некоторые из них включают в себя:

* apiai
* assemblyai
* google-cloud-speech
* pocketsphinx
* SpeechRecognition
* watson-developer-cloud
* wit

Некоторые из этих пакетов, такие как wit и apiai, предлагают встроенные функции, такие как обработка на естественном языке для определения намерений говорящего, которые выходят за рамки базового распознавания речи. Другие, такие как google-cloud-speech, сосредоточены исключительно на преобразовании речи в текст.

SpeechRecognition: - пакет, позволяет создавать сценарии для доступа к микрофонам и обработки аудиофайлов с нуля.

Библиотека SpeechRecognition действует как оболочка для нескольких популярных речевых API и, таким образом, является чрезвычайно гибкой.

Гибкость и простота использования пакета SpeechRecognition делают его отличным выбором для любого проекта Python. Тем не менее, поддержка каждой функции каждого API, который он включает, не гарантируется.

## 2.2 Технология RobotPulse

Для того, чтобы обеспечить общение программы с конкретным роботом, я использовал технологию *RobotPulse*. Данная технология является библиотекой языка программирования python и принадлежит разделу *pulseapi*.

С помощью этой технологии мы можем обращаться к конкретному роботу по сети с помощью его IP-адреса, а также следующей команды:

*robot = RobotPulse("http://192.168.1.52:8081")*

Библиотека pulse-api содержит множество различных функций и методов, которые может выполнять робот. В частности, в ней содержатся следующие методы:

* robot.set\_pose(pose\_target, 90);
* robot.run\_positions(position\_targets, 90);
* robot.await\_stop(1);

Таким образом, мы можем задавать роботу те команды, которые нам нужны для решения нашей промышленной задачи.

## 2.3 Опыт с реальным роботом

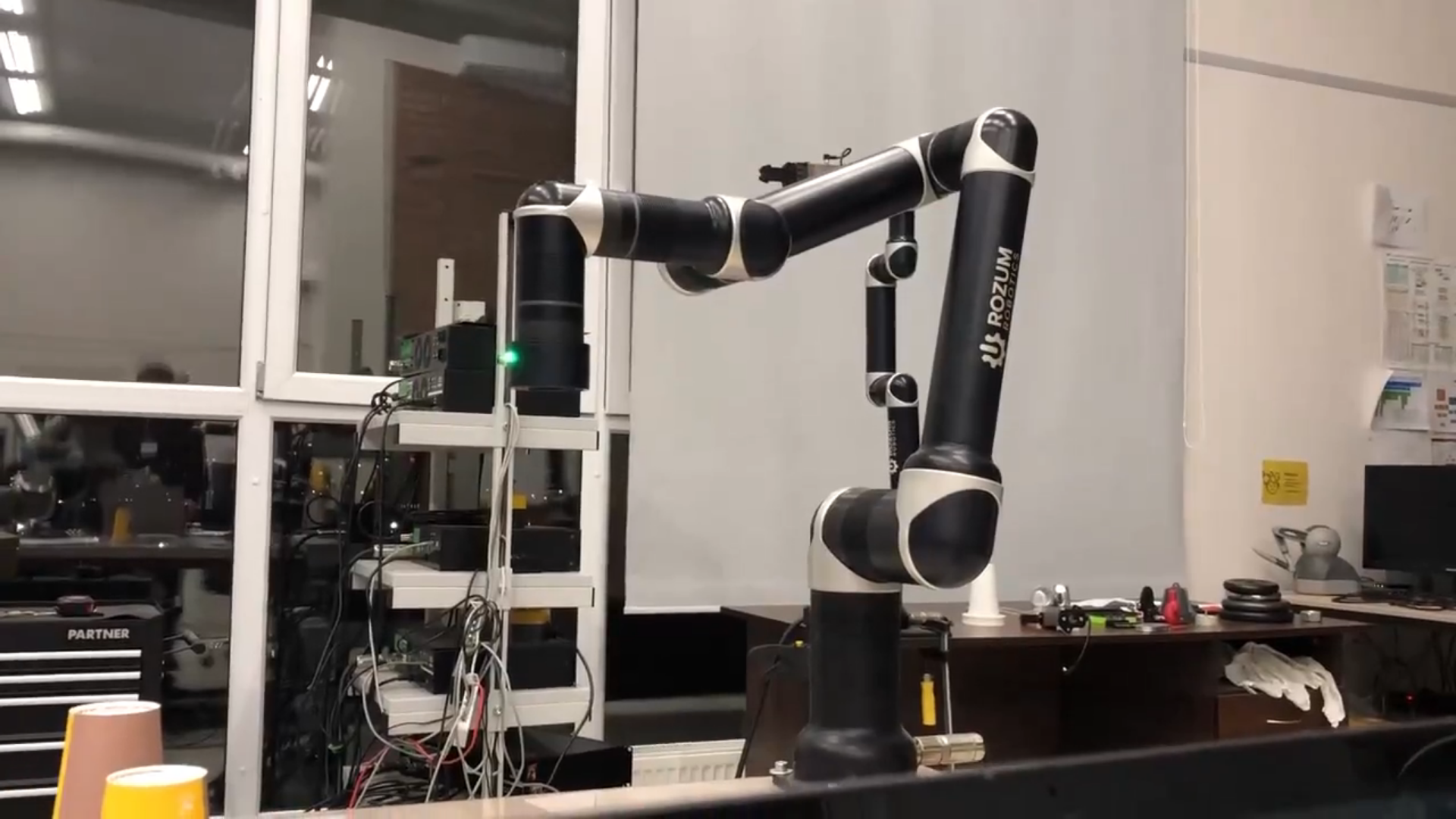
Итак, передо мной стояла задача: собрать вместе все вышеперечисленные технологии и сделать так, чтобы к конкретному промышленному роботу можно было обращаться голосовыми командами с целью той или иной ответной реакции от робота.

Для этого я написал код на языке программирования python (см. приложение А) в среде разработки PyCharm. Я подключил все необходимые библиотеки: SpeechRecognition, PyAudio, RobotPulse и некоторые другие.

Я подключился к роботу по его IP-адресу через технологию Wi-Fi, после чего я мог посылать команды созданному в коде объекты.

Для распознавания речи программа подключается к GoogleTranslate, после чего переводит услышанные фразы в строки и посылает роботу.

В своём опыте я использовал робота, представляющего собой «руку», которая может делать определённые движения, что может быть полезно на практике (см. рис. 7).

Рис. 7. Опыт с реальным роботом

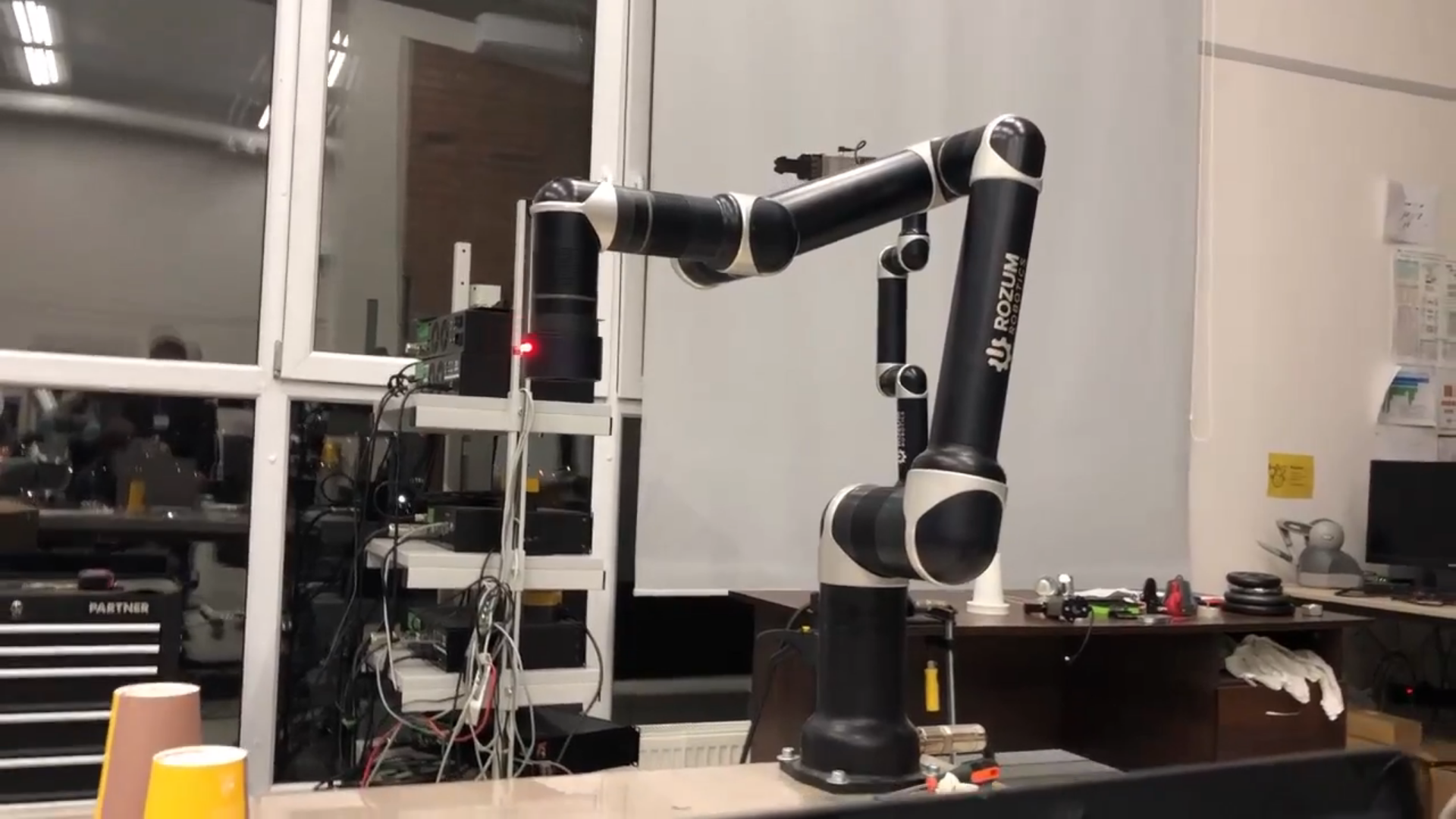
Одной из команд была команда «Stop», которая выключала робота (см. рис. 8):

Рис. 8. Робот в выключенном состоянии

Ещё одной командой была команда «Home», которая переводит робота в начальную позицию (см. рис. 9):

Рис. 9. Робот в своей начальной позиции

Также для робота я использовал и многие другие команды. Все они тем или иным образом приводили робота в движение. В частности, сочетание команд «Start» и «Square» заставляло робота делать движения по траектории квадрата (см. рис. 10):

Рис. 10 Результат работы команды «Square»

# ЗАКЛЮЧЕНИЕ

По итогу с помощью вышеуказанных технологий удалось создать более десятка команд для промышленного робота, которые могут быть даны не только из кода программы, но и с помощью речи, что может быть очень полезно при работе с роботами на практике.

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

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.

# ПРИЛОЖЕНИЕ А

**Листинг файла main.py**

# Python program to translate

# speech to text and text to speech

import speech\_recognition as sr

import pyttsx3

import pyaudio

from pulseapi import RobotPulse

import math

from pulseapi import RobotPulse, pose, position

from commands import ref

robot = RobotPulse("http://192.168.1.52:8081")

# Initialize the recognizer

r = sr.Recognizer()

# Function to convert text to

# speech

def SpeakText(command):

# Initialize the engine

engine = pyttsx3.init()

engine.say(command)

engine.runAndWait()

# Loop infinitely for user to

# speak

command\_lst = []

if \_\_name\_\_ == '\_\_main\_\_':

while True:

# Exception handling to handle

# exceptions at the runtime

try:

# use the microphone as source for input.

with sr.Microphone() as source2:

# wait for a second to let the recognizer

# adjust the energy threshold based on

# the surrounding noise level

r.adjust\_for\_ambient\_noise(source2, duration=0.2)

print("Program\nYou can start speaking. You have only 10 seconds on fraze. Start...\n")

# listens for the user's input

audio2 = r.listen(source2, 10, 5)

print("Program\nOK, I got it, take me some time to translate with google translate on the net...\n")

# Using google to recognize audio

MyText = r.recognize\_google(audio2)

MyText = MyText.lower()

if "stop" in MyText:

ref["stop"]()

elif "activate" in MyText:

ref["activate"]()

elif "run" in command\_lst and "home" in MyText:

command\_lst = []

ref["home"]()

elif "run" in command\_lst and "square" in MyText:

command\_lst = []

ref["square"]()

elif "run" in command\_lst and "line" in MyText:

command\_lst = []

ref["line"]()

elif "run" in command\_lst and "rectangular" in MyText:

command\_lst = []

ref["rectangular"]()

elif "run" in command\_lst and "dance" in MyText:

command\_lst = []

ref["dance"]()

elif "run" in command\_lst and "move" in MyText:

command\_lst = []

ref["move"]()

elif "run" in command\_lst and "slow" in MyText:

command\_lst = []

ref["slow"]()

elif "run" in command\_lst and "fast" in MyText:

command\_lst = []

ref["fast"]()

elif "start" in MyText:

command\_lst.append("run")

# Result = "Did you say " + str(MyText)

print("Program\nI heard: \"" + MyText + "\"\n")

# SpeakText(MyText)

except sr.RequestError as e:

print("Could not request results; {0}".format(e))

except sr.UnknownValueError:

print("unknown error occured")

**Листинг файла commands.py**

import math

import multiprocessing

from concurrent.futures import ThreadPoolExecutor

from pulseapi import RobotPulse, position, pose

robot = RobotPulse("http://192.168.2.4:8081")

pool = ThreadPoolExecutor(2)

def activate():

pool.submit(robot.recover)

def stop():

pool.submit(robot.stop)

def home():

def wrap():

robot.set\_pose(pose([0, -90, 0, -90, -90, 0]), 100)

robot.await\_stop()

pool.submit(wrap)

def square():

def wrap():

while True:

pose\_target = pose([0, -90, 90, -90, -90, 0])

position\_targets = [

position([-0.37, -0.12, 0.35], [math.pi, 0, 0]),

position([-0.42, -0.12, 0.35], [math.pi, 0, 0]),

position([-0.42, -0.17, 0.35], [math.pi, 0, 0]),

position([-0.37, -0.17, 0.35], [math.pi, 0, 0]),

]

robot.set\_pose(pose\_target, 90)

robot.run\_positions(position\_targets, 90)

robot.await\_stop(1)

pool.submit(wrap)

def line():

def wrap():

while True:

pose\_target = pose([0, -90, 90, -90, -90, 0])

position\_targets = [

position([-0.3, -0.1, 0.35], [math.pi, 0, 0]),

position([-0.3, -0.3, 0.35], [math.pi, 0, 0]),

position([-0.3, -0.5, 0.35], [math.pi, 0, 0]),

position([-0.3, -0.7, 0.35], [math.pi, 0, 0]),

]

robot.set\_pose(pose\_target, 90)

robot.run\_positions(position\_targets, 90)

robot.await\_stop(1)

pool.submit(wrap)

def rectangular():

def wrap():

while True:

pose\_target = pose([0, -90, 90, -90, -90, 0])

position\_targets = [

position([-0.5, -0.1, 0.6], [math.pi, 0, 0]),

position([-0.3, -0.3, 0.7], [math.pi, 0, 0]),

position([-0.7, -0.2, 0.4], [math.pi, 0, 0]),

position([-0.3, -0.4, 0.5], [math.pi, 0, 0]),

]

robot.set\_pose(pose\_target, 90)

robot.run\_positions(position\_targets, 90)

robot.await\_stop(1)

pool.submit(wrap)

def dance():

def wrap():

while True:

pose\_target = pose([0, -90, 90, -90, -90, 0])

position\_targets = [

position([-0.8, -0.3, 0.4], [math.pi, 0, 0]),

position([-0.2, -0.6, 0.6], [math.pi, 0, 0]),

position([-0.9, -0.1, 0.4], [math.pi, 0, 0]),

position([-0.1, -0.5, 0.6], [math.pi, 0, 0]),

]

robot.set\_pose(pose\_target, 90)

robot.run\_positions(position\_targets, 90)

robot.await\_stop(1)

pool.submit(wrap)

def move():

def wrap():

while True:

pose\_target = pose([0, -70, 60, -80, -80, 0])

position\_targets = [

position([-0.8, -0.5, 0.8], [math.pi, 0, 0]),

position([-0.7, -0.4, 0.7], [math.pi, 0, 0]),

position([-0.6, -0.3, 0.6], [math.pi, 0, 0]),

position([-0.5, -0.2, 0.5], [math.pi, 0, 0]),

]

robot.set\_pose(pose\_target, 90)

robot.run\_positions(position\_targets, 90)

robot.await\_stop(1)

pool.submit(wrap)

def slow():

def wrap():

while True:

pose\_target = pose([0, -20, 30, -10, -20, 10])

position\_targets = [

position([-0.5, -0.3, 0.5], [math.pi, 0, 0]),

position([-0.5, -0.3, 0.5], [math.pi, 0, 0]),

position([-0.4, -0.4, 0.6], [math.pi, 0, 0]),

position([-0.4, -0.4, 0.6], [math.pi, 0, 0]),

]

robot.set\_pose(pose\_target, 90)

robot.run\_positions(position\_targets, 90)

robot.await\_stop(1)

pool.submit(wrap)

def fast():

def wrap():

while True:

pose\_target = pose([0, -45, 45, -45, -45, 0])

position\_targets = [

position([-0.9, -0.3, 0.4], [math.pi, 0, 0]),

position([-0.2, -0.7, 0.7], [math.pi, 0, 0]),

position([-0.3, -0.1, 0.4], [math.pi, 0, 0]),

position([-0.7, -0.4, 0.9], [math.pi, 0, 0]),

]

robot.set\_pose(pose\_target, 90)

robot.run\_positions(position\_targets, 90)

robot.await\_stop(1)

pool.submit(wrap)

ref = {

"activate": activate,

"stop": stop,

"home": home,

"square": square,

"line": line,

"rectangular": rectangular,

"dance": dance,

"move": move,

"slow": slow,

"fast": fast

}