**МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ**

**БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**

**ФАКУЛЬТЕТ ПРИКЛАДНОЙ МАТЕМАТИКИ И ИНФОРМАТИКИ**

**ОТЧЕТ**

по экзаменационному проекту учебной дисциплины

«Сервис-ориентированные программные системы»

Вариант 8

Выполнили:

Лепеева Эмилия Юрьевна, 3 курс, ПИ, 12 группа

Бахмутова Полина Алексеевна, 3 курс, ПИ, 12 группа

Минск, 2025

**Пояснительная записка**

**1. Задачи проекта и роли**

Основные задачи проекта были следующие:

1. Разработать (создать) виртуальную среду, представляющую собой два стола, на одном из которых расположены одноцветные параллелепипеды различных цветов. Второй стол пустой. Между столами может перемещаться колесный робот с исполнительным устройством (манипулятором).

2. Робот должен воспринимать и распознавать звуковую команду: цвет пераллелепипеда.

3. Робот должен перенести с первого стола на второй все параллелепипеды цвета, заданного голосовой командой.

В ходе выполнения проекта были внесены следующие изменения: виртуальная среда представляет собой стол, конвейер и корзину, параллелепипеды заданного цвета перемещаются на стол, а остальные – в корзину.

Среда разработки – Webots R2023b.

Язык программирования ­– Python 3.9.9

Роли проекта:

* Создание виртуальной среды и проектирование конечного автомата – Бахмутова Полина;
* Разработка кода и тестирование робота – Лепеева Эмилия.

**2. Процесс разработки**

* Исходные данные и подготовка:

В проекте использовалась виртуальная среда, созданная с помощью Webots, для моделирования движения робота. Робот состоит из нескольких моторов (для управления рукой и пальцами) и датчиков (камера для распознавания объектов, датчики расстояния и положения). Камера распознает объекты и их цвета, что служит основой для принятия решения о дальнейших действиях робота.

* Проектирование конечного автомата:

Для управления поведением робота был разработан конечный автомат. Этот автомат определяет состояние робота в зависимости от событий, происходящих в системе, таких как обнаружение объекта или завершение действия с манипулятором (описание конечного автомата представлено в следующей главе).

* Реализация взаимодействия с внешними объектами:

На уровне кода реализовано взаимодействие с объектами (параллелепипедами). Камера обнаруживает объект, считывает цвет и ожидает, когда объект приблизиться на необходимое для захвата расстояние. После того как объект оказывается в «клешне», идет определение конечной точки на основе заданного пользователем цвета.

* Реализация задания цвета с помощью голоса и его использование для работы робота:

Для записи цвета написан скрипт get\_color.py. Его необходимо запускать отдельно до основной работы робота. Скрипт считывает название цвета как на русском, так и на английском, а после записывает его в файл color.txt. В основном коде цвет, с которым робот должен взаимодействовать, считывается из этого же текстового файла. Если цвет объекта соответствует требуемому, робот выполняет действия по захвату и перемещению объекта.

* Обработка ошибок:

В процессе разработки были учтены возможные ошибки, такие как отсутствие файла, неверное значение цвета и ошибки в коммуникации с устройствами. Эти ошибки обрабатываются с помощью исключений, что гарантирует стабильную работу программы.

**3. Конечный автомат:**

Для выполнения поставленной задачи робот использует конечный автомат, который управляет его поведением на основе состояний и событий, происходящих в процессе работы. В данном случае, конечный автомат управляет действиями робота, такими как захват объекта, его поворот, отпускание и возврат в исходную позицию, в зависимости от сенсорных данных и текущего состояния.

***Набор состояний***

Робот может находиться в одном из следующих состояний:

* WAITING (Ожидание): Робот находится в ожидании команды. Он не выполняет никаких действий и готов реагировать на события, такие как обнаружение объекта.
* GRASPING (Захват): Робот захватывает объект, выполняя действие сжатия захвата с помощью захвата руки (грипера).
* ROTATING (Поворот): Робот поворачивает свою манипуляторную руку в нужную позицию в зависимости от цвета объекта, который он захватил.
* RELEASING (Отпускание): Робот отпускает объект, открывая захват руки (грипера).
* ROTATING\_BACK (Возврат в исходное положение): Робот возвращает манипуляторную руку в исходное положение после того, как отпустил объект.

***События, при которых происходит переход между состояниями***

* Object Detected (Обнаружен объект): Происходит, когда датчик расстояния фиксирует объект на близком расстоянии (менее 500). Это событие переводит робота в состояние GRASPING.
* Grasp Complete (Захват завершен): Происходит после того, как робот завершил захват объекта (захват был произведен успешно). Это событие переводит робота в состояние ROTATING.
* Rotation Complete (Поворот завершен): Когда манипуляторная рука достигает нужной позиции, что проверяется с помощью датчика положения (например, если значение датчика < -2.3). Это событие переводит робота в состояние RELEASING.
* Release Complete (Отпускание завершено): После того как робот отпустил объект (грипер открыт). Это событие переводит робота в состояние ROTATING\_BACK.
* Rotation Back Complete (Возврат в исходную позицию завершен): Когда робот возвращает манипуляторную руку в исходную позицию (состояние датчика положения > -0.1). Это событие переводит робота в состояние WAITING.

***Действия, выполняемые при переходе между состояниями***

* WAITING → GRASPING:

Считывание цвета объекта. Если цвет объекта совпадает с требуемым (читаем из файла), переходим к захвату объекта.

* GRASPING → ROTATING:

Если объект захвачен, робот поворачивает манипуляторную руку в нужную позицию в зависимости от цвета объекта.

* ROTATING → RELEASING:

После того как манипуляторная рука достигла требуемого положения, робот отпускает объект, открывая грипер.

* RELEASING → ROTATING\_BACK:

После отпускания объекта робот возвращает руку в исходное положение.

* ROTATING\_BACK → WAITING:

После того как рука вернулась в исходное положение, робот снова готов ожидать следующего объекта.