

Федерация Спортивной и Образовательной
робототехники

Российская Робототехническая Олимпиада 2024
Творческая категория

«Инновационные решения в промышленности»

ОТЧЕТ по проекту
«Коллаборативный робот-манипулятор «Collabaka».
Автоматизация производственных процессов»

Команда «Техномейкер»

Средняя возрастная категория

Выполнил:

Герасименко Георгий Андреевич,
ученик 8 «А» класса
БОУ г.Омска «Лицей БИТ»;

Руководитель:

Аллагулов Станислав Сайфуллович,
педагог дополнительного образования
БУ ДО «Омская областная СЮТ»
детский технопарк «Кванториум»

г. Омск,
2024

КОЛЛАБОРАТИВНЫЙ РОБОТ-МАНИПУЛЯТОР «COLLABAKA»

Герасименко Георгий Андреевич
Омская область, г.Омск, БОУ г. Омска "Лицей БИТ", 8 класс

Аннотация

Коллаборативный робот-манипулятор «Collabaka» — это промышленное оборудование, которое выполняет функции человеческой руки. Роборука может быть как отдельным устройством и работать бок о бок с человеком, так и составной частью сложного роботизированного комплекса.

Моя промышленная роборука с тремя степенями свободы может выполнять движения вдоль трёх осей координат. Каждая ось оснащена отдельным сервоприводом. Также роборука имеет редукторный захват.

Для того чтобы Робот-манипулятор «Collabaka» мог выполнять автономно задания, я его оснастил машинным зрением. Для этого подключил web-камеру к raspberry pi 4 и написал скрипту на Python с использованием открытой библиотеки OpenCV.

При обнаружении объекта в заданной области заданным цветом Робот-манипулятор «Collabaka» начинает выполнять отведённое ей задание. Сейчас моя роборука способна сортировать вещи на складе, обнаруживать и перемещая необходимый предмет на сборочной стол оператора.

Примеры применения робота-манипулятора «Collabaka»:

- В нефтеперерабатывающей отрасли роботы-манипуляторы используются для автоматизации работы с опасными веществами.
- Автоматизация производственных процессов на производстве, таких как сборка, упаковка, погрузка-разгрузка и т.д.
- Коллаборативная автоматизация процессов на производстве, где часть работы выполняет робот-манипулятор, а часть работы люди. Так, например, я хочу его научить помогать мне в 3D-печати. Как известно после окончания печати модель необходимо извлечь из принтера, перед тем как запустить новую, и я уверен, что мой коллаборативный робот-манипулятор «Collabaka» сможет мне в этом помочь. И благодаря роботу мне не придётся подходить к принтеру каждый раз после окончания печати, что неизбежно поможет мне делать свои проекты гораздо быстрее 😊

Ключевые слова: Коллаборативная робототехника, робот-манипулятор, 3D моделирование, проектирование, FDM печать, программирование на Arduino/Python, машинное зрение.

Содержание

Обоснование актуальности	3
Цели и задача	3
Анализ существующих решений	3
Дорожная карта	3
Техническое описание механизмов робота-манипулятора	4
Разработка роботизированной руки	4
Расчет максимальной полезной нагрузки роборуки	5
Дизайнерское решение	6
Проектирование схемы электроники и программирование	6
Сборка, тестирование. Экспериментальная часть.	7
Анализ полученных результатов	7
Коммерциализация	8
Выводы	8
Список литературы:	9
Приложение 1. Технические характеристики	10
Приложение 2. Схема электрическая принципиальная	11
Приложение 3. Фрагмент программного кода для Arduino	12

Обоснование актуальности

Автоматизация процессов очень важна на любом производстве, т.к. от этого зависит качество, количество и себестоимость товаров, которые производит то самое производство. Люди предрасположены делать ошибки, есть даже такое понятие как человеческий фактор, и его можно сократить до минимума автоматизируя однотипные процессы. Поэтому у меня возникла идея спроектировать коллаборативного робота-манипулятора, который бы помогал работникам на производстве.

Цели и задача

До сих пор на производстве, будь то конвейерная лента на сборочном производстве или сортировка товаров на складе, можно встретить людей, которые каждую смену делают однотипную работу.

Повсеместное внедрение коллаборативных роботов-манипуляторов позволит сократить человеческий ресурс, который затрачивается на выполнение примитивных задачах и позволит переквалифицировать персонал в более наукоёмкие профессии, что в конечном счете позитивно отразится на ситуации и предприятие сможет выпускать продукцию и быстрее и качественнее, а за счет переквалификации персонала также сможет увеличить ассортимент и рынок сбыта, что в конечном счете только увеличит прибыль предприятию.

Коллаборативный робот-манипулятор позволит выйти на качественно новый уровень производства!

Анализ существующих решений

Проанализировав существующие решения, я пришел к выводу, что рынок промышленных роботов-манипуляторов уже достаточно сформирован, но решения очень дорогие и требуют сложнейшей настройки. Одними из самых популярных являются роборуки KUKA и FANUC.

Дорожная карта

Я проанализировал существующие конструкции и решения и выработал свою дорожную карту по разработке коллаборативного робота-манипулятора Collabaka:

1. Проектирование трех осевой роборуки

Я спроектировал роборуку в САПР Компас 3D. Получилось больше 20 деталей, которые я собрал в сборку и получил 3D модель робота-манипулятора.

2. Разработка электрической схемы и сборка электроники

Мною была разработана схема на базе arduino и raspberry pi.

3. Изготовление деталей и сборка функционального прототипа

Детали робота-манипулятора я распечатал на 3D принтере. Запрограммировал arduino+raspberry pi, реализовав движение и захват с помощью машинного зрения с распознаванием предмета по цвету.

Техническое описание механизмов робота-манипулятора

Разработка роботизированной руки

Роборука – это многоцелевая роботизированная манипуляторная установка с захватом. Всё проектирую в САПР Компасе 3D [1, с.3], детали собираю в сборку, если всё ок, то печатаю на 3D принтере и собираю.

Изучил учебное пособие по роботизированным системам [2, с.8] и спроектировал роборуку с тремя степенями свободы №1, №2, №3 (см.рис.2) и редукторной клешней захвата №4 (см.рис.6), это позволяет использовать маломощный сервопривод для удерживания больших предметов. Роборука представлена на рис.6. До изготовления мною было произведено математическое моделирование в ПО ARTAS SAM механики роборуки, для того чтобы оценить рабочие характеристики, результаты представлены на рис.1.

Максимальный вертикальный вылет получился 576мм, максимальный горизонтальный вылет 366мм. Весь рабочий диапазон изображен на рис.4 оранжевым цветом. Этих характеристик достаточно для выполнения задач по сбору, переключиванию и транспортированию крупных предметов.

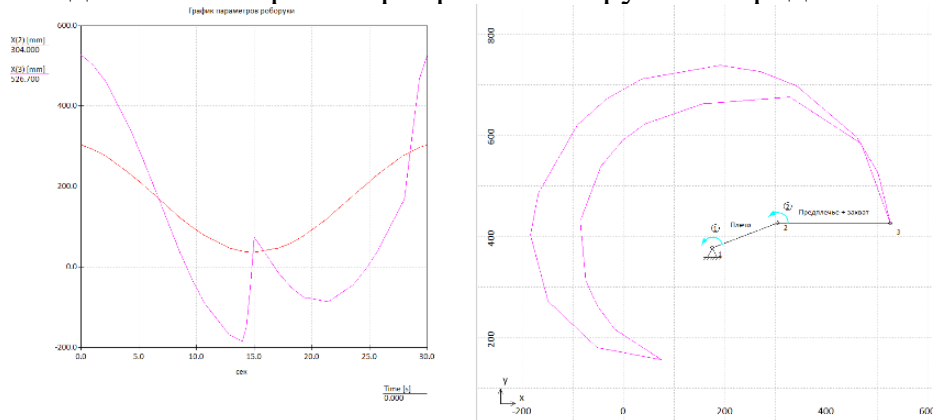


Рис.1

Также я предусмотрел поворотную ось №3 (см.рис.2) для захвата, что помогает захватывать предметы любой формы. Максимальный поворот составляет $\pm 100^\circ$

Редукторный захват №4 (см.рис.2) на роботизированной руке является съемным, его можно заменить на другой захват в зависимости от целей. На текущий момент я спроектировал редукторный захват в виде клешни.

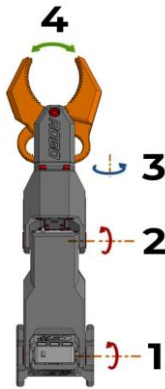


Рис.2

В качестве сервоприводов были использованы сервоприводы с двигателем постоянного тока и максимальным крутящим моментом 25кг*см. Такой сервопривод способен удерживать груз до 25кг на рычаге длиной 1см от вала сервопривода. Зная длины и все всех сегментов роботизированной руки, я провел расчет полезной нагрузки, которую робот может перемещать в пространстве.

Расчет максимальной полезной нагрузки роборуки

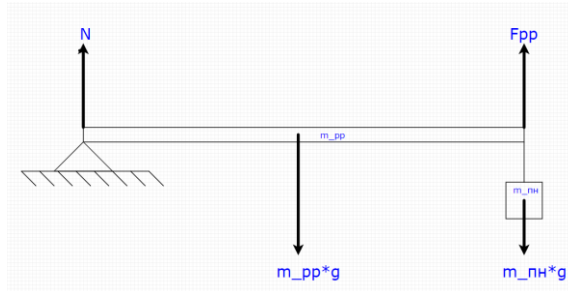
Так как я использую одинаковые сервоприводы, то максимальную нагрузку определяет сервопривод, который стоит на максимальном рычаге, это сервопривод плеча №1 (см.рис.5)

$$M_{\text{сервопривода}} = 2,45 \text{ Н} * \text{м} \text{ (момент на валу сервопривода)}$$

$$l = 0,36 \text{ м} \text{ (длина роборуки)}$$

F_{pp} – сила требуемая роборуке для поднятия груза массой $m_{\text{пн}}$

$$m_{\text{pp}} = 0,5 \text{ кг} \text{ (масса роборуки без полезной нагрузки)}$$



Составим уравнение суммы моментов относительно точки крепления роборуки

$$m_{\text{pp}} * g * \frac{l}{2} + m_{\text{пн}} * g * l = N * 0 + F_{\text{pp}} * l$$

$$M_{\text{сервопривода}} = F_{\text{pp}} * l$$

$$m_{\text{пн}} = \frac{M_{\text{сервопривода}}}{g * l} - \frac{m_{\text{pp}}}{2}$$

$$m_{\text{пн}} = \frac{2,45}{9,8 * 0,36} - \frac{0,5}{2} = 0,44 \text{ кг}$$

Таким образом мы получили, что максимальная полезная нагрузка составляет 440 гр. Этот показатель легко можно увеличить в 2,9 раза заменив

*сервопривод плеча на более мощный, с максимальным крутящим моментом 5,36 Н*м (55кг*см)*

В ходе проведения испытаний выяснилось, что угловые ускорения и скорости работы сервоприводов слишком высокие. При захвате полезной нагрузки близкой к максимальной возникали сложности в управлении из-за инерции и перегреву сервоприводов. Самым высоконагруженным сервоприводом оказался самый нижний, он у меня сгорал дважды на испытаниях. Выходы из строя сервоприводов меня навели на мысль, что нужно как-то ограничивать ток или ограничивать ускорения и скорости. Идеальным решением проблемы стала бы установка дополнительных редукторов, но для этого нужно было бы менять сервоприводы на шаговые двигатели, т.к. максимальный ход у сервоприводов 180 градусов и переделываться концептуально всю роботу. Но для первой версии я счел достаточным использование специальной библиотеки для сервоприводов, в которой можно ограничить ускорение и скорость сервопривода. Результат меня устроил, роборука стала медленнее и плавнее перемещаться с грузом.

Еще одной особенностью роборуки является скрытая проводка, я очень много времени уделил кабель-менеджменту, в текущем исполнении роборука и выглядит красиво и защищена от механических воздействий при перевозке крупных грузов, см рис. 3.

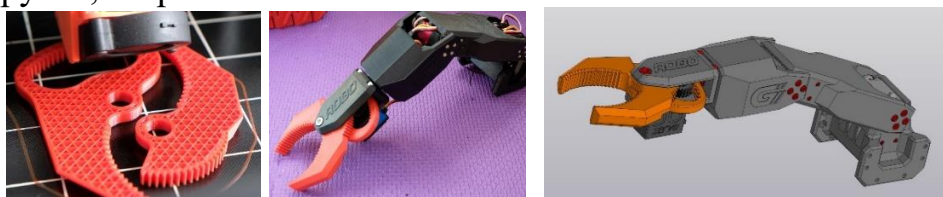


Рис.3

Дизайнерское решение

Промдизайн при проектировании занимает далеко не последнее значение, поэтому я переделываю эскиз по несколько раз, провожу опрос среди своих друзей и одноклассников, и только потом как начинаю изготавливать.

Проектирование схемы электроники и программирование

Схему собирал на Arduino, Raspberry pi, web-камере. Электрическая схема приведена в прил.2.

Скетч для Arduino кодил в Arduino IDE, сигналы на сервы идут как раз с Arduino, читал книжку [3, с.250], смотрел ролики в youtube, боролся с багами. Скрипт для Raspberry PI 4 кодил ещё дольше. С использованием библиотеки OpenCV [4] удалось реализовать распознавание цвета, выделение области контуром, поиск центра контура и передача этих координат на Arduino для дальнейшего управления сервами.

Сборка, тестирование. Экспериментальная часть.

Все пластиковые элементы распечатал на своём 3D принтере пластиком ПЭТГ. Процесс печати деталей очень сильно растянулся во времени из-за медленной технологии FDM. Не все детали подходили с первого раза, даже если в сборке САПР Компас3D всё выглядело отлично. Т.к. пластик ПЭТГ имеет усадку после печати, то ее тоже нужно было учитывать, иначе либо зазоров не было, либо были большие. С опытом начинаешь понимать, как правильно проектировать детали для печати на 3D-принтере и становиться меньше брака и потерянного времени.

После того как был собрана роборука Collabaka я закрепил ее на фанере 300*400мм для того, чтобы протестировать и выявить слабые узлы. Самым большим моим просчетом является слабый сервопривод первой оси роборуки, который влияет на максимально подъемный вес. В ходе испытаний переработал поворотная ось роботизированной руки, которая оказалось слабой и не выдерживала расчетную нагрузку.

Мне нравится то, чем я занимаюсь, и я делюсь этим со зрителями через свой youtube канал «ТехноМейкер». На канале у меня выложены видеоролики со сборкой и тестированием роборуки Collabaka [5].

Анализ полученных результатов

Коллаборативный робот-манипулятора Collabaka получился функциональным и красивым. Collabaka уже сейчас имеет автономные функции по распознаванию, захвату и перемещению объектов в пространстве. Полные технические характеристики представлены в Прил.1. Виртуальный тур с обзором 3D-модели моего робота см. по ссылке [6], там его можно посмотреть со всех сторон.

Мне еще очень много предстоит сделать. Я ищу программиста в команду, чтобы ускорить разработку проекта, т.к. сам больше люблю моделировать и создавать новые механизмы и машины. Хочу сделать робота-манипулятора автономным и безопасным, для этого нужно реализовать целый комплекс мер, начиная от установки вращающегося лазерного лидара для построения цифрового пространства до внедрения искусственного интеллекта для анализа информации с датчиков и принятия решений.

Коммерциализация

Я провел экономический анализ, сравнение с конкурентами и подсчитал емкость рынка. Если коллаборативного робота-помощника Collabaka вывести на рынок, то можно заработать 5,1 млрд.руб, см.Табл.1.

Таблица 1. Оценка объема рынка

Сверху вниз	Снизу вверх
TAM - 102,0 млрд руб. Рынок автоматизации предприятий в России	SOM – 1.08 млрд руб. В конкурентной среде продукт займет 5%.
SAM – 51,0 млрд руб. 10% услуг оказывается для ВТГ (государства)	SAM – 10.8млрд руб. 10% услуг оказывается для ВТГ (государства)
50% услуг оказывается для ВТС (бизнес)	50% услуг оказывается для ВТС (бизнес)
SOM – 5.1 млрд руб. В конкурентной среде продукт займет 5%	TAM – 21.6 млрд руб. Рынок автоматизации предприятий в России.

Выводы

Подводя итоги создания коллаборативного робота-манипулятора Collabaka, я вижу огромный потенциал использования его автоматизации многих производственных процессов. Робот Collabaka сможет делать рутинную работу по сборке, сортировке, погрузке/разгрузке/загрузке и т.п., в итоге человек сможет заниматься более высококвалифицированным трудом.

Я уверен, что мой робот-манипулятора Collabaka будет приносить пользу людям!

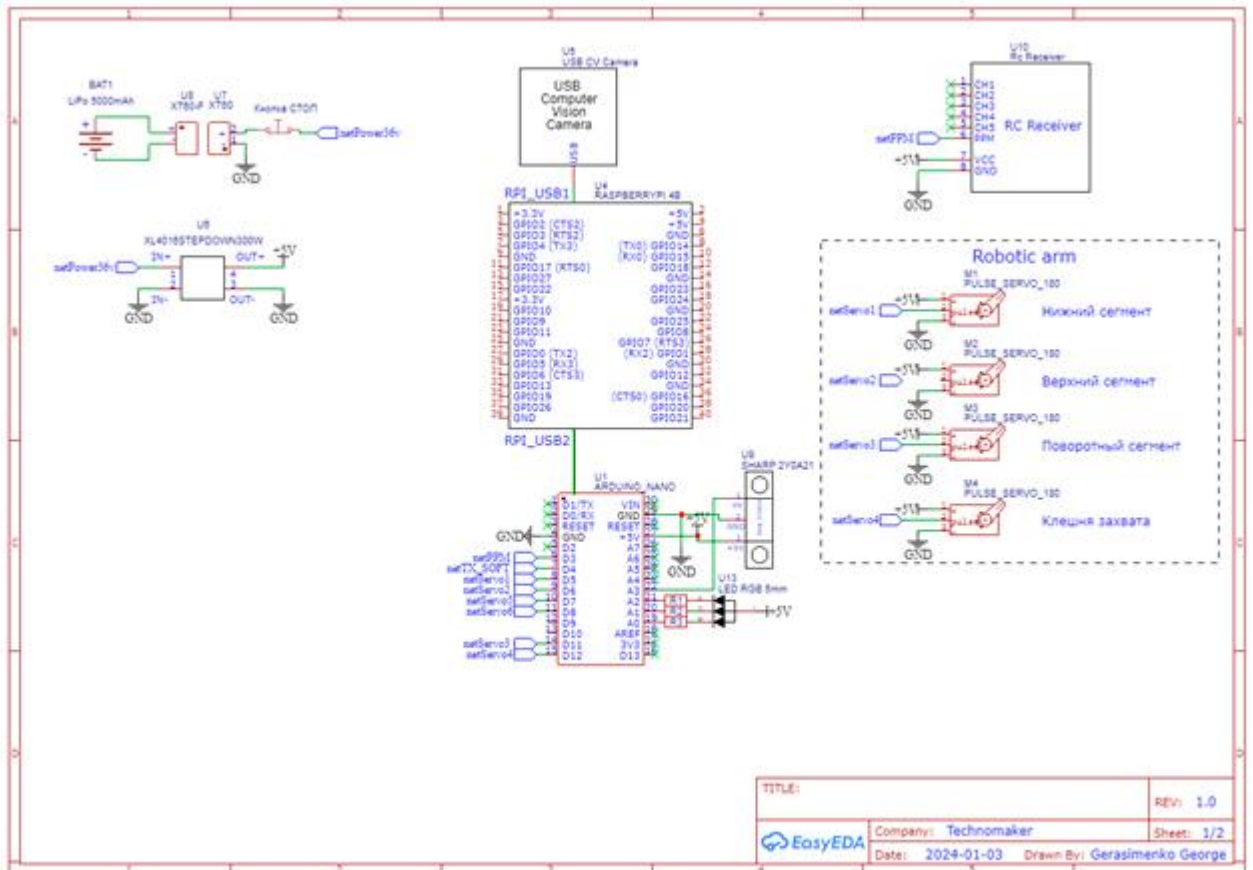
Список литературы:

1. Никонов Вячеслав. КОМПАС-3D: создание моделей и 3D-печать./В.В.Никонов — СПб.: Питер, 2020. — 208 с.: ил. — (Серия «Учебное пособие»).
2. Робототехнические системы: Учебное пособие / В. Г. Хомченко — Омск: ОмГТУ – 195 стр., 2016.
3. Цикл статей OpenCV в Python /Sthitaprajna Mishra, MaxRokatansky / [Электронный ресурс]. – Режим доступа: <https://habr.com/ru/companies/otus/articles/558426/>
4. Петин В. А. Проекты с использованием контроллера Arduino./ В.А.Петин — СПб.: БХВ-Петербург, 2014. — 400 с.
5. Герасименко Георгий Андреевич. Youtube-channel “Техномейкер”. Обзор и создание робота-манипулятора Collabaka / Г.А. Герасименко / [Электронный ресурс]. – Режим доступа: https://www.youtube.com/watch?v=8H4Ghihb_5c&t=73s
6. Герасименко Георгий Андреевич. 3D тур с виртуальным обзором робота Collabaka на платформе sketchfab.com/Г.А. Герасименко / [Электронный ресурс]. – Режим доступа: <https://sketchfab.com/3d-models/robotic-arm-collabaka-00c79eeb1af74af3874fee336a52d5a2>

Приложение 1. Технические характеристики

Модификация	Collabaka CV
Эксплуатационная масса, кг	3
Сервоприводы	DC
Кол-во осей вращения, шт	3
Напряжение питания, В	5
Машинное зрение	Есть (web-camera + raspberry pi)
Роборука	
Степеней свободы	3
Крутящий момент сервопривода, кг*см	25
Полезная нагрузка, г	500
Длина роботизированной руки, мм	355
Захват	Клешневой, редукторный
Максимальное раскрытие захвата, мм	70
Удельное давление при захвате, кг/см ²	0,5
Рабочие характеристики	
Габаритные размеры (ДхШхВ) без отвала, мм	300x400x410
Материал изделия	Пластик PETG, фанера

Приложение 2. Схема электрическая принципиальная



Приложение 3. Фрагмент программного кода для Arduino

```
1.
2. // ##### УПРАВЛЕНИЕ РОБОРУКОЙ #####
3. // считываем данные положения роборукам с пульта и мапив в диапазон работы сервоприводов
4. if (millis() - myTimer >= 40) {
5.   myTimer = millis();
6.   rot_cur = map(ppm.latestValidChannelValue(6, 0), 980, 2030, 170, 10); // поворотная ось
7.   zah_cur = map(ppm.latestValidChannelValue(5, 0), 980, 2030, 10, 170); // захват
8.   arm2_cur = map(ppm.latestValidChannelValue(4, 0), 980, 2030, 160, 10); // предплечье
9.   arm1_cur = map(ppm.latestValidChannelValue(3, 0), 980, 2030, 160, 10); // плечо
10.
11. // управление поворотной осью роборуки
12. if (abs(rot_prev - rot_cur) > 2) { // зря не дергаем сервой
13.   servos[0].setTargetDeg(rot_prev + 0.1 * (rot_cur - rot_prev)); // если отклонение значительное, то плавно идем к значению
14.   rot_prev = rot_prev + 0.1 * (rot_cur - rot_prev); // сохраняем текущее положение
15. }
16. else {
17.   servos[0].setTargetDeg(rot_prev); // если меньше 2 градусов, возможно это помеха, ничего не делаем
18. }
19. // управление захватом роборуки
20. if (abs(zah_prev - zah_cur) > 2) { // зря не дергаем сервой
21.   servos[1].setTargetDeg(zah_prev + 0.1 * (zah_cur - zah_prev)); // если отклонение значительное, то плавно идем к значению
22.   zah_prev = zah_prev + 0.1 * (zah_cur - zah_prev); // сохраняем текущее положение
23. }
24. else {
25.   servos[1].setTargetDeg(zah_prev); // если меньше 2 градусов, возможно это помеха, ничего не делаем
26. }
27.
28. // управление предплечьем роборуки
29. if (abs(arm2_prev - arm2_cur) > 2) { // зря не дергаем сервой
30.   servos[2].setTargetDeg(arm2_prev + 0.1 * (arm2_cur - arm2_prev)); // если отклонение значительное, то плавно идем к значению
31.   arm2_prev = arm2_prev + 0.1 * (arm2_cur - arm2_prev); // сохраняем текущее положение
32. }
33. else {
34.   servos[2].setTargetDeg(arm2_prev); // если меньше 2 градусов, возможно это помеха, ничего не делаем
35. }
36.
37. // управление плечом роборуки
38. if (abs(arm1_prev - arm1_cur) > 2) { // зря не дергаем сервой
39.   servos[3].setTargetDeg(arm1_prev + 0.1 * (arm1_cur - arm1_prev)); // если отклонение значительное, то плавно идем к значению
40.   arm1_prev = arm1_prev + 0.1 * (arm1_cur - arm1_prev); // сохраняем текущее положение
41. }
42. else {
43.   servos[3].setTargetDeg(arm1_prev); // если меньше 2 градусов, возможно это помеха, ничего не делаем
44. }
45.
46.
47. // ##### END #####
```