



Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение высшего образования  
«Московский государственный технический университет имени Н.Э. Баумана  
(национальный исследовательский университет)»  
(МГТУ им. Н.Э. Баумана)

# Проектирование программно- аппаратного комплекса «Куб для визуализации трехмерного изображения»

Студент: Шпаковский П. А. ИУ7-53Б

Руководитель: Строганов Ю. В.

Москва, 2023 г.

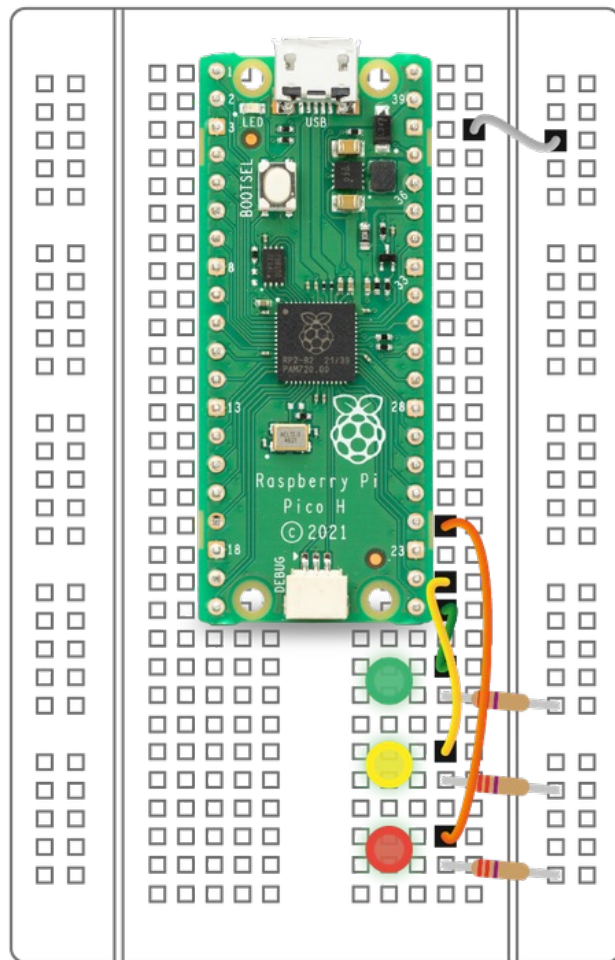
# Цели и задачи

Цель данной работы – спроектировать программно-аппаратный комплекс для построения моделей трехмерных объектов с использованием микроконтроллера, а также разработать макет устройства с шестью гранями, где каждая грань – экран, на котором отображается проекция выбранной пользователем трехмерной модели.

Чтобы достичь поставленной цели, требуется решить следующие задачи:

- описать структуру трехмерной сцены, включая объекты, из которых состоит сцена, и определить формат задания исходных данных;
- выбрать наиболее подходящий из существующих алгоритмов трехмерной графики, позволяющих синтезировать изображение трехмерной сцены;
- реализовать выбранные алгоритмы построения трехмерной сцены;
- исследовать возможности микроконтроллеров в области визуализации трехмерной графики.

# Выбор аппаратной платформы



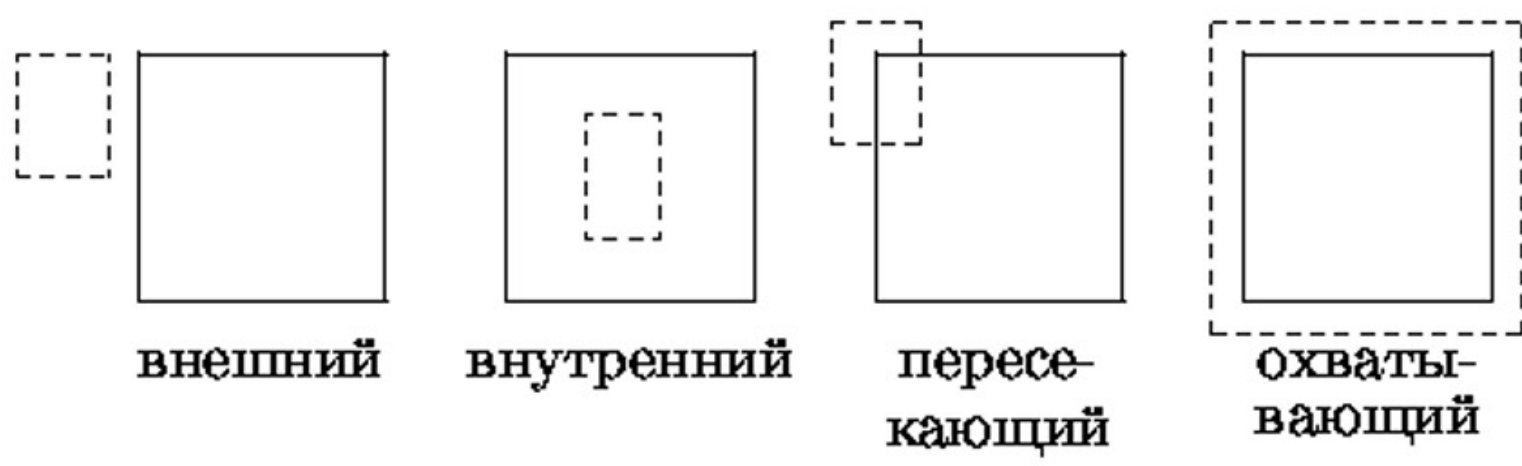
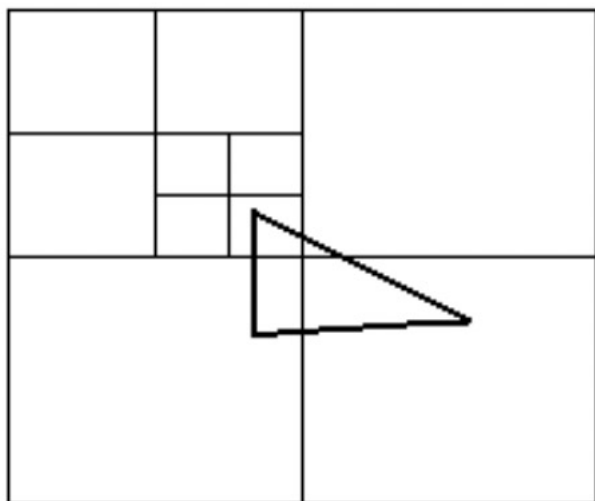
В данной работе используется микроконтроллер RP2040, оснащенный процессором семейства ARM Cortex-M0+ и отладочная плата Raspberry Pi Pico.

Характеристики:

- тактовая частота процессора 133 МГц;
- объем оперативной памяти 264 Кб;
- объем постоянной памяти 2 Мб;
- частота шины SPI 40 МГц;
- наличие встроенного контроллера DMA.

Также используются 2 SPI дисплея с разрешением 240x240 пикселей с контроллером дисплея ST7789.

# Выбранный алгоритм удаления невидимых поверхностей



## Алгоритм Варнока

Преимуществом данного алгоритма является использование меньшего объема памяти, чем другие алгоритмы. Таким образом, задача визуализации может быть решена с использованием микроконтроллера, для которого объем памяти ограничен 264 Кб.

Также с помощью аппаратных возможностей платформы данный алгоритм может быть модифицирован для наиболее эффективного использования ресурсов.

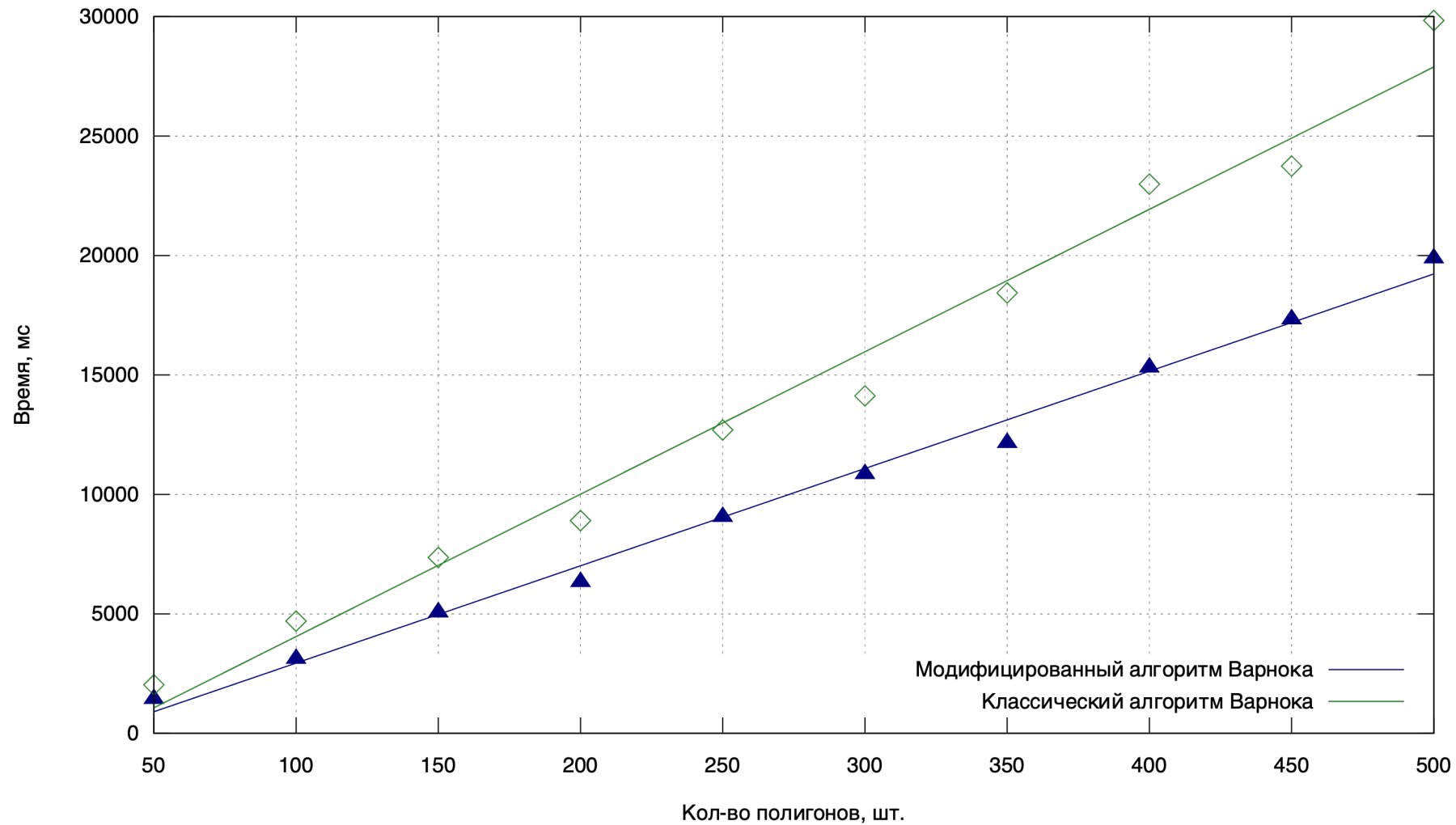
# Модифицированный алгоритм Варнока

В данной работы был модифицирован классический алгоритм Варнока для более эффективного использования вычислительных ресурсов микроконтроллера.

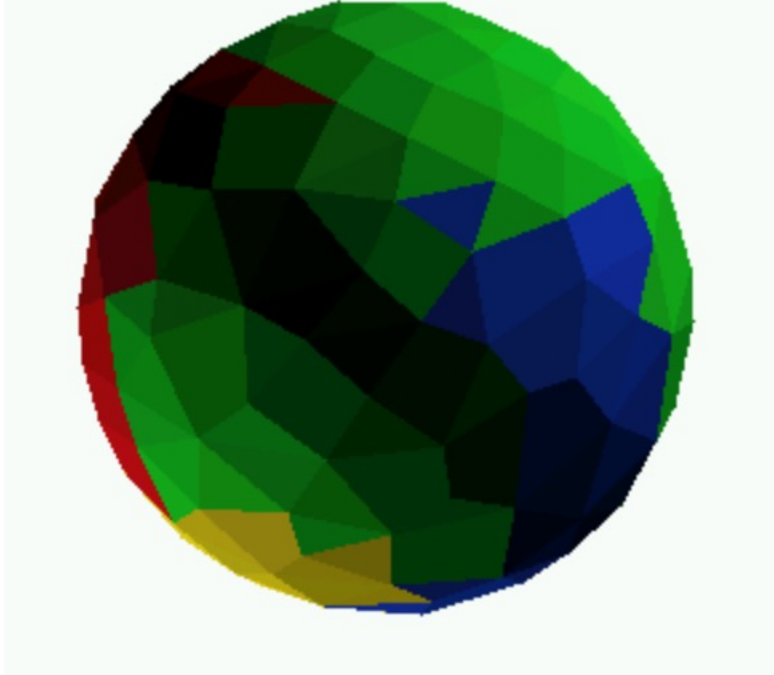
Это возможно за счет механизма прямого доступа к памяти, при котором процессор не участвует в пересылке данных между микроконтроллером и дисплеем. За это отвечает отдельный аппаратный модуль – контроллер DMA.

Также была модифицирована логика работы самого алгоритма Варнока для решения данной задачи.

# Сравнение классического и модифицированного алгоритмов Варнока



# Алгоритм закрашки



$$I_{\alpha} = I_0 \cos \alpha.$$

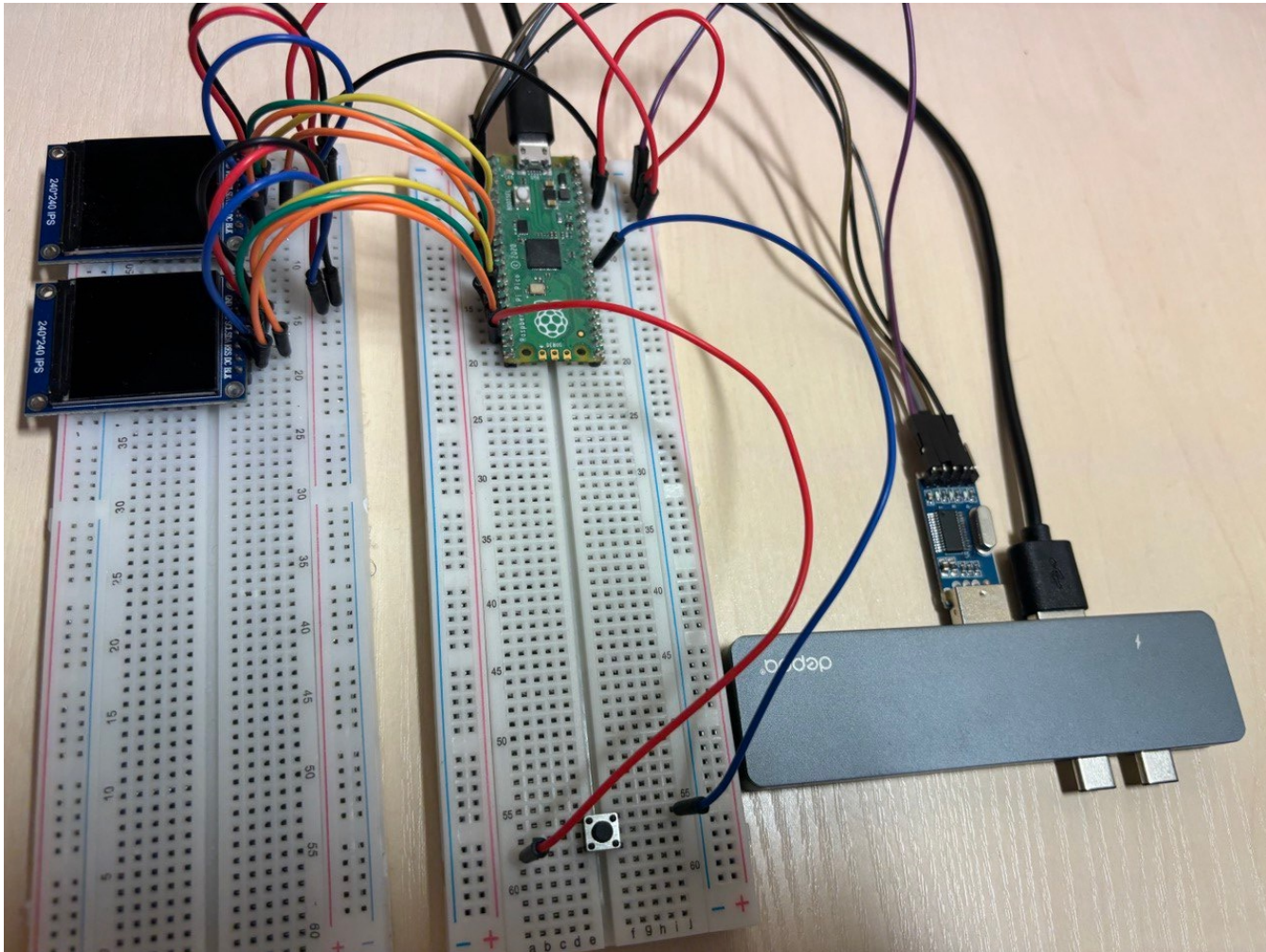
Применяется закрашивание, в котором каждая грань получает один уровень интенсивности, вычисляемый согласно закону Ламберта.

В результате такой закрашки все плоские поверхности, включая те, что аппроксимируют фигуры, подвергаются однородному окрашиванию.

Этот метод обладает высокой производительностью, однако все пиксели на грани получают одинаковую интенсивность, что придает сцене нереалистичный вид.



# Макет устройства

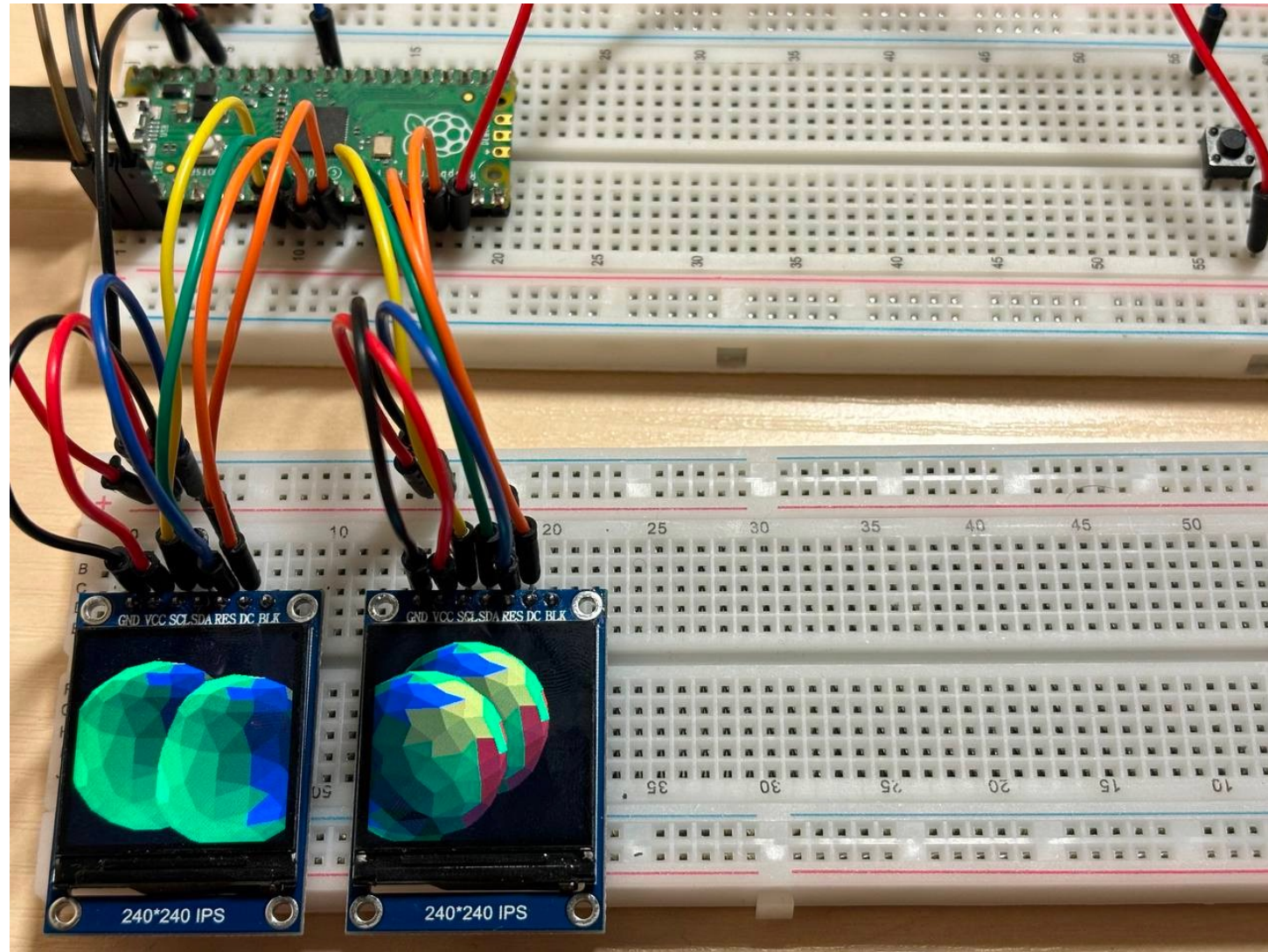


Макет устройства включает в себя следующие компоненты:

- Микроконтроллера Raspberry Pi Pico
- 2 SPI дисплея с разрешением 240x240 пикселей
- Кнопка сброса питания
- 2 макетные платы и провода
- Подключенный UART-USB преобразователь



# Пример работы программно-аппаратного комплекса



# Пример консольного интерфейса

Raspberry Pi Pico 3D

Инструкция:

help     Вывод информации о командах

models   Вывод списка названий доступных трехмерных сцен

load <название сцены> -- Загрузка сцены по ее названию

camera set cp <x, y, z> ct <x, y, z> cu <x, y, z> -- Установка камеры по трем векторам направлений

camera rotate <rx, ry, rz> -- Вращение камеры, где rx, ry, rz – углы поворота по осям в градусах

camera scale <k> -- Масштабирование камеры, где k – коэффициент масштабирования

camera reset -- Сброс настроек камеры к значению по умолчанию

Количество полигонов на сцене = 12

load sphee

Неправильное название сцены, проверьте список

load spheres

Количество полигонов на сцене = 536

camera rotate 0 30 0

camera rotate 90 90 90

camera scla

Неверное число аргументов

camera scale 0.5

camera reset

# Заключение

В ходе выполнения данной курсовой работы цель была достигнута и были решены следующие задачи:

- описана структура трехмерной сцены, включающая объекты, из которых состоит сцена, и определен формат задания исходных данных;
- выбран наиболее подходящий из существующих алгоритмов трехмерной графики, позволяющих синтезировать изображение трехмерной сцены;
- реализованы выбранные алгоритмы построения трехмерной сцены;
- исследованы возможности микроконтроллеров в области визуализации трехмерной графики.