

ПРОЕКТ «УМНЫЙ ДОМ»



Подготовил: Исаков Ренат
Брянская область, г.Клинцы
ГАУ ДО «Клинцовский детский технопарк
«Кванториум»

ЦЕЛЬЮ МОЕГО ПРОЕКТА ЯВЛЯЕТСЯ
ИЗУЧЕНИЕ И ТЕСТИРОВАНИЕ
СИСТЕМЫ С ВОЗМОЖНОСТЯМИ
«УМНОГО ДОМА», РАССМОТРЕНИЕ
ПРИМЕНЕНИЙ ЭЛЕМЕНТНОЙ БАЗЫ,
РАЗЛИЧНЫХ ДАТЧИКОВ,
ИСПОЛНИТЕЛЬНЫХ МЕХАНИЗМОВ,
ПРОГРАММИРУЕМЫХ
МИКРОКОНТРОЛЛЕРОВ И
ВОЗМОЖНОСТЕЙ БЕСПРОВОДНОЙ
СВЯЗИ ДЛЯ УПРАВЛЕНИЯ И
ПОЛУЧЕНИЯ ИНФОРМАЦИИ.



ЗАДАЧИ:

1. Собрать и систематизировать доступную информацию.
2. Рассмотреть возможные примеры создания и реализации данной системы.
3. Определить для себя системы и элементную базу для реализации.
4. Разработать в графическом редакторе макет.
5. Создать макет из имеющихся материалов.
6. Запрограммировать микроконтроллер.
7. Выполнить тестирование и проверку на работоспособность.

Подбор компонентов

1) Микро
сервопривод



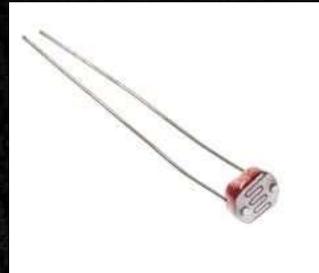
2) Светодиод



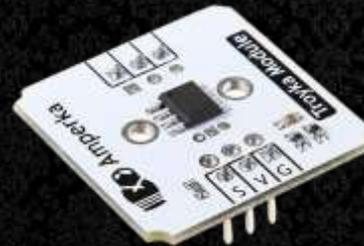
3) RGB светодиод



4) Фоторезистор

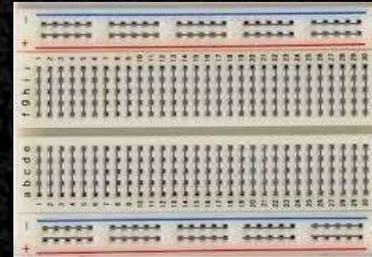


5) Резисторы: 220 Ом,
10 кОм



6) Bluetooth – модуль

7) Аналоговый
термометр



8) ARDUINO

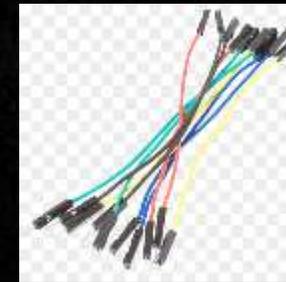


9) Малая макетная
плата

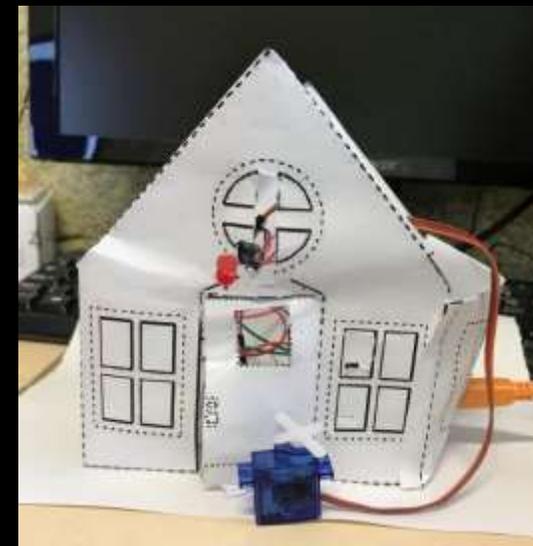


10) Провода

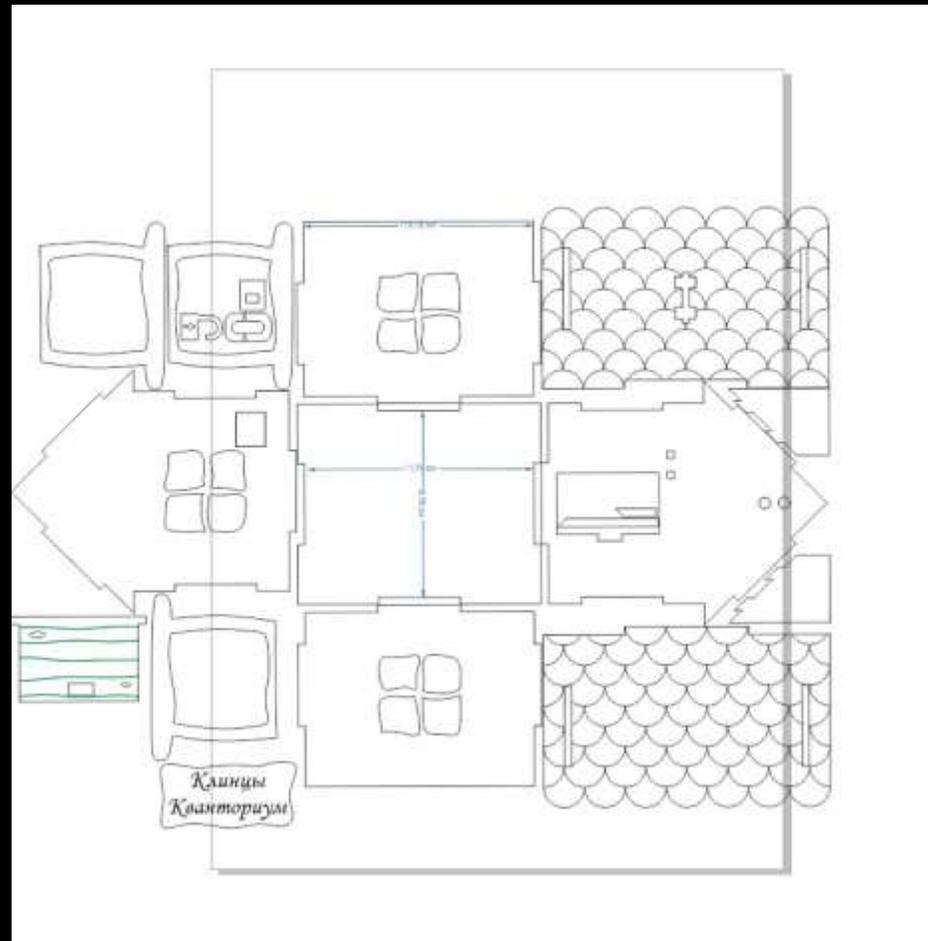
11) Соединительные
провода



Изготовлен первоначального макета из бумаги, для проверки расположения комплектующих, датчиков и исполнительных механизмов.



ПРОЕКТИРОВАНИЕ ЧЕРТЕЖА



ИЗГОТОВЛЕНИЕ ГОТОВОГО МАКЕТА



ПРОГРАММИРОВАНИЕ МИКРОКОНТРОЛЛЕРА

```
#define PIN_LED 11 //фотодатчик
#define PIN_PHOTO_SENSOR A0 //фотодатчик

#include <TroykaThermometer.h> // подключаем библиотеку термометр
TroykaThermometer thermometer(A2); //подключаем его во
//2 аналоговый пин
#include <Servo.h> // подключаем библиотеку сервопривод
Servo servo; // обозначаем переменную servo типа servo

int val; // целое число

void setup() {
  Serial.begin(9600); // скорость передачи данных
  pinMode(PIN_LED, OUTPUT); // устанавливает режим работы
  // пина на выход
  servo.attach(9); // сервопривод

  pinMode(4, OUTPUT); // пин для RGB светодиода
  pinMode(5, OUTPUT);
  pinMode(6, OUTPUT);
}

void loop() { // цикл
  thermometer.read(); // считывает показатели термометра
  Serial.println("Temperature is "); // выводим надпись
  Serial.println
  (thermometer.getTemperatureC()); // получаем температуру
  Serial.println(" C"); // выводим надпись
  delay(1000); // задержка = (мл)
  if (Serial.available())

  if (Serial.available ()) {
    val = Serial.read();
```

```
if (val=='1') { servo.write(10);} // дверь закрыть
if (val=='2') { servo.write(100);} // открыть

if (val == '4') { digitalWrite(4, HIGH);} // RGB светодиод
if (val == '0') { digitalWrite(4, LOW);}
if (val == '5') { digitalWrite(5, HIGH);}
if (val == '0') { digitalWrite(5, LOW);}
if (val == '6') { digitalWrite(6, HIGH);}
if (val == '0') { digitalWrite(6, LOW);}

}

int val = analogRead(PIN_PHOTO_SENSOR); // получаем данные
// с фотодатчика

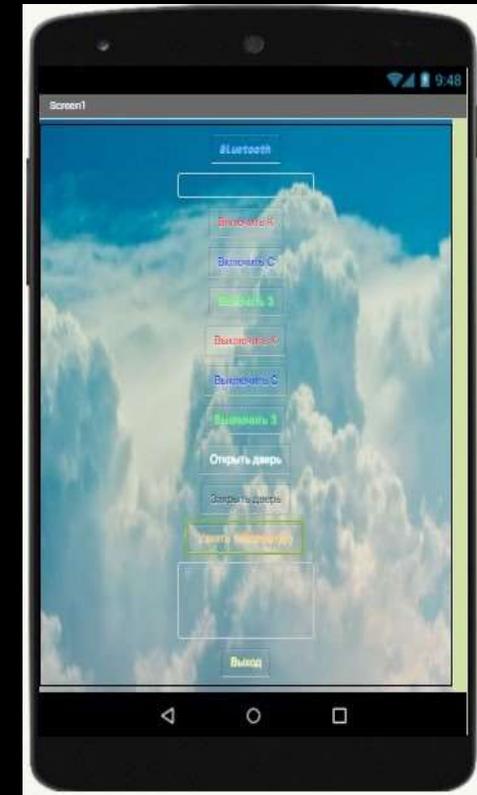
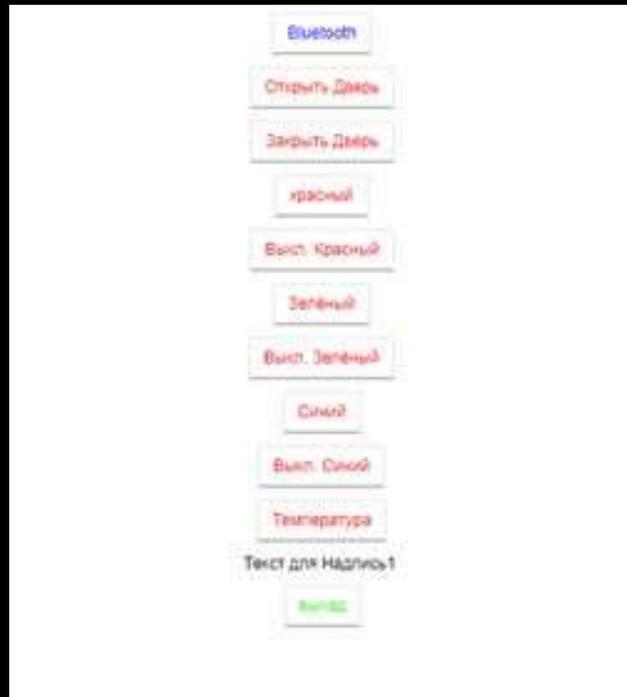
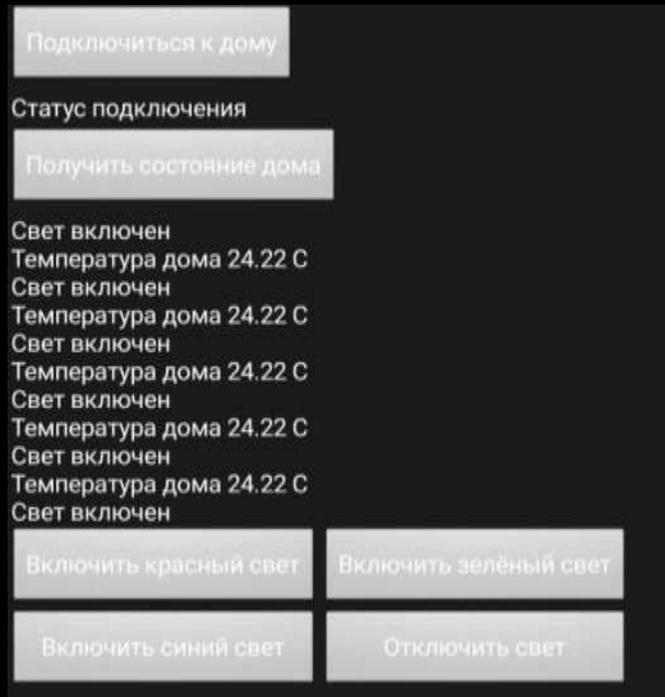
if (val < 1000) {
  digitalWrite(PIN_LED, LOW); //низкий уровень сигнала на пине
} else {
  digitalWrite(PIN_LED, HIGH); //высокий уровень сигнала
if (val < 1000) {
  Serial.println("off"); //если низкий то выводим off
else {Serial.println("on");} //если высокий то выводим on
}
```

СОЗДАНИЕ ПРОГРАММЫ УПРАВЛЕНИЯ ДЛЯ СМАРТФОНА. [HTTPS://APPINVENTOR.MIT.EDU/](https://appinventor.mit.edu/)

The image displays a collection of visual programming blocks from MIT App Inventor, organized into several groups:

- Color Selection Events:** Six blocks for "Красный", "Зелёный", "Синий", "Выкл_Красный", "Выкл_Зелёный", and "Выкл_Синий". Each block has a "Щелчок" (click) event and a "вызов КлиентBluetooth1 .ПослатьТекст" (call ClientBluetooth1 .SendText) action with a specific text value in quotes (e.g., "4", "0", "2", "6", "1").
- Door Control Events:** Two blocks for "Открыть_Дверь" (Open Door) and "Закрыть_Дверь" (Close Door), both with "Щелчок" events and "вызов КлиентBluetooth1 .ПослатьТекст" actions with text values "2" and "1" respectively.
- Bluetooth Selection:** A "Bluetooth" event block for "ПередВыбором" (Before Selection) with a "присвоить Bluetooth .Элементы в КлиентBluetooth1 .АдресаИИмена" (assign Bluetooth .Elements to ClientBluetooth1 .AddressesAndNames) action.
- Bluetooth Connection:** A "Bluetooth" event block for "ПослеВыбора" (After Selection) with a "присвоить Bluetooth .Выбор в вызов КлиентBluetooth1 .Подключить адрес Bluetooth .Выбор" (assign Bluetooth .Selection to call ClientBluetooth1 .Connect address Bluetooth .Selection) action.
- Exit Event:** A "Выход" (Exit) event block with a "Щелчок" event and a "закреть приложение" (close application) action.
- Temperature Event:** A "Температура" (Temperature) event block with a "Щелчок" event and a "присвоить Надпись1 .Текст в вызов КлиентBluetooth1 .ПолучитьТекст числоБайтов вызов КлиентBluetooth1 .БайтовДоступноДляПринятия" (assign Label1 .Text to call ClientBluetooth1 .GetText numberBytes call ClientBluetooth1 .BytesAvailableForReception) action.

СОЗДАЕМ ИНТЕРФЕЙС ПРОГРАММЫ. ПОДБИРАЕМ И НАСТРАИВАЕМ РАЗЛИЧНЫЕ ВАРИАНТЫ.



ЗАКЛЮЧЕНИЕ.

Подводя итог, я могу сказать, что мне удалось достичь поставленной цели и создать действующий прототип «Умного дома», управляемый дистанционно при помощи приложения на смартфоне.

С самим результатом можно ознакомиться на видео по указанной ссылке:

<https://youtu.be/4dgmcrKB2wU>



СПАСИБО ЗА ВНИМАНИЕ!