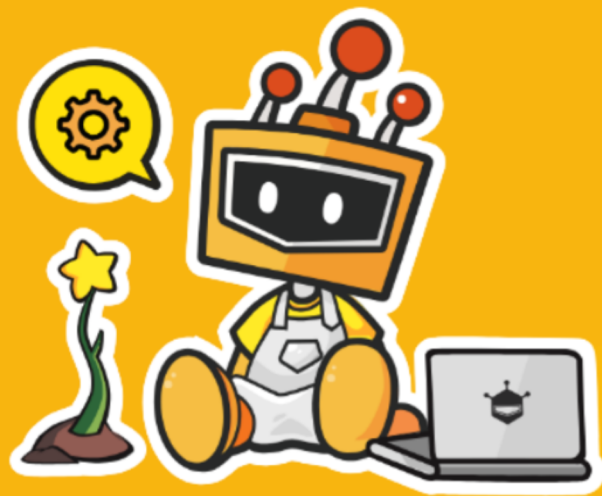
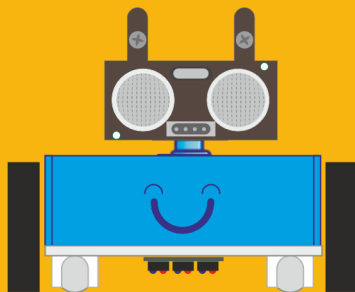


РУКОВОДСТВО ПО РАБОТЕ С НАБОРОМ «Я – РОБОТ»

FREE

MIND+

Часть 1
Знакомство

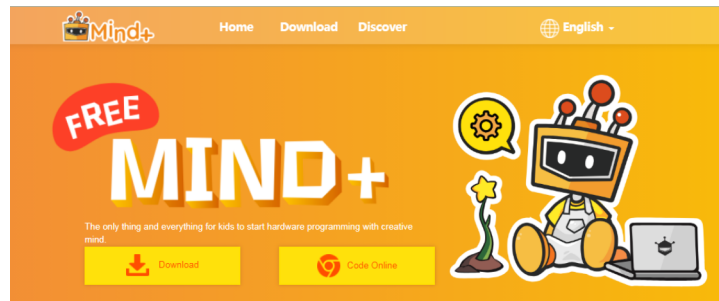


Начало. Установка. Режимы

Mind+ — это графическая платформа визуального программирования, которая поддерживает все виды аппаратного обеспечения с открытым исходным кодом, такие как Arduino, ESP32, micro:bit.

Платформа не только предназначена для младших школьников, но и предоставляет удобную и интуитивно понятную среду обучения для студентов и взрослых энтузиастов, которые изучают языки программирования высокого уровня, такие как Arduino, Python, C, C++ и др.

С помощью Mind+ вы сможете легко создавать программы, перетаскивая и комбинируя командные блоки, быстро ощутите радость творчества и уверенность в собственных силах.

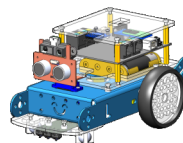


Для начала работы:

- Загрузите дистрибутив приложения: <http://www.mindplus.cc> (для Windows это файл Mind+ Link Setup 0.2.0.exe)
- Установите Mind+, запустив дистрибутив на выполнение.

Для управления Arduino-совместимым роботом в Mind+ существуют два режима: **автономный** и **интерактивный**.

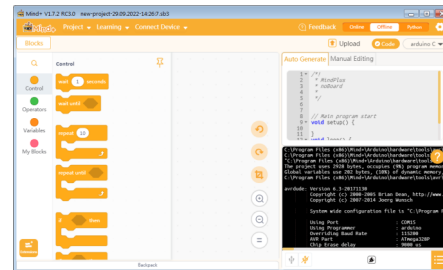
АВТОНОМНЫЙ РЕЖИМ



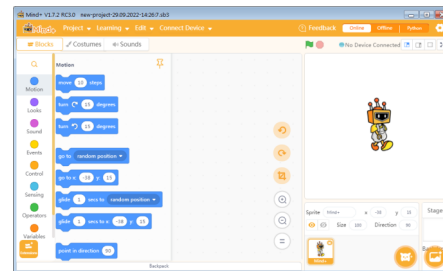
ИНТЕРАКТИВНЫЙ РЕЖИМ



В *автономном режиме* вы загружаете в «мозги» Робота программу (скетч) с помощью кабеля USB. Затем можете отключить кабель, и Робот будет выполнять загруженную программу.



В *интерактивном режиме* Робот постоянно «привязан» к ПК кабелем USB. Сама программа выполняется на компьютере, а Робот получает и выполняет команды от Mind+.



Начало. Автономный режим

Панель меню

Загрузка скетча на плату

Переключение режимов

Область скриптов
Здесь вы будете создавать программы (сценарии), которые будут управлять спрайтами. Блоки программ располагаются слева

Программный код

Панель блоков
Здесь располагаются разноцветные блоки для создания сценариев, сгруппированные по темам

Консоль последовательного порта

Работа с последовательным портом

```
1  /*  
2   * MindPlus  
3   * noBoard  
4   *  
5   */  
6  
7  
8  // Main program start  
9  void setup() {  
10  
11  }  
12  void loop() {  
13  
14  }  
15
```

Начало. Интерактивный режим

The image shows the Mind+ V1.7.2 RC3.0 interface. The top menu bar includes Project, Learning, Edit, and Connect Device. The main workspace is divided into three sections: a left sidebar for block categories, a central script area, and a right stage area. The left sidebar is categorized into Motion, Looks, Sound, Events, Control, Sensing, Operators, and Variables. The central script area contains a sequence of motion blocks: 'move 10 steps', 'turn 15 degrees' (twice), 'go to random position', 'go to x: -38 y: 15', 'glide 1 secs to random position', 'glide 1 secs to x: -38 y: 15', and 'point in direction 90'. The right stage area displays a robot sprite with its parameters (x: -38, y: 15, size: 100, direction: 90) and a 'Stage' dropdown menu. Annotations with callouts identify key interface elements: the 'Панель меню' (Menu Panel) at the top; the 'Панель блоков' (Block Panel) on the left; the 'Область скриптов' (Script Area) in the center; the 'Сцена' (Stage) on the right; the 'Переключение режимов' (Mode Switching) button in the top right; the 'Выбор сцены' (Scene Selection) button at the bottom right; the 'Параметры текущего персонажа' (Current Character Parameters) section at the bottom left; and the 'Библиотека персонажей (спрайтов)' (Character Library (Sprites)) section at the bottom right.

Панель меню

Панель блоков
Здесь располагаются разноцветные блоки для создания сценариев, сгруппированные по темам

Область скриптов
Здесь вы будете создавать программы (сценарии), которые будут управлять спрайтами. Блоки программ располагаются слева

Переключение режимов

Сцена

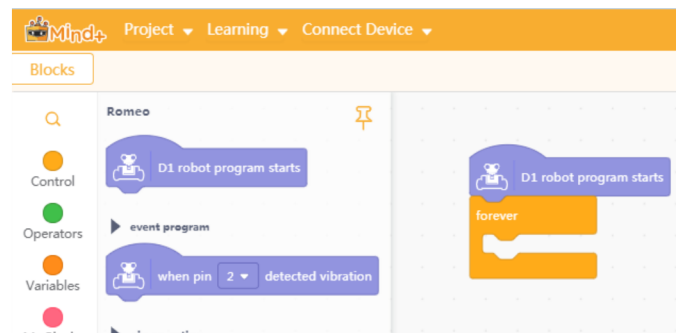
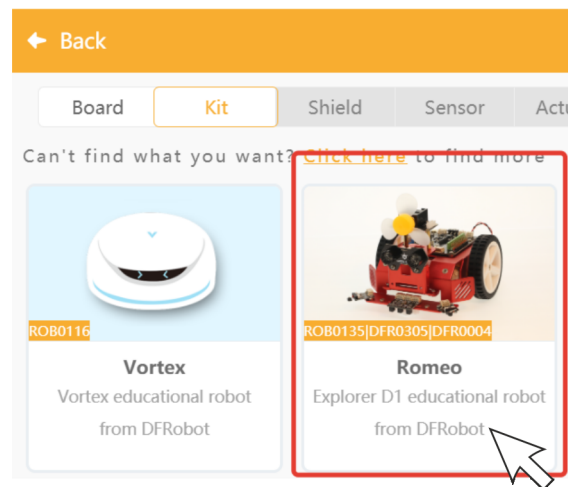
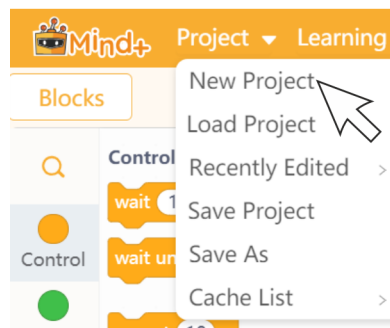
Выбор сцены

Параметры текущего персонажа

Библиотека персонажей (спрайтов)

Начало. Загрузка блоков

Нам нужно загрузить командные блоки, которые будет «понимать» Робот. В нашем случае выберем Arduino-совместимого робота Romeo.

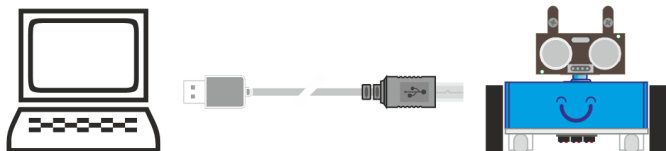


Нажмите кнопку **Back**.

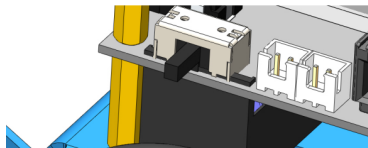
На панели блоков появились блоки для программирования робота.

Начало. Подключение Робота к ПК

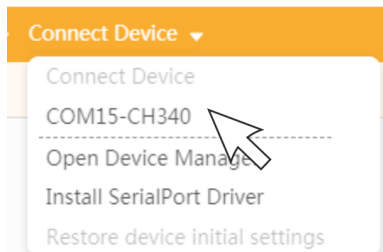
1. Подключите Робота к ПК с помощью кабеля USB.



2. Включите питание на плате Робота.



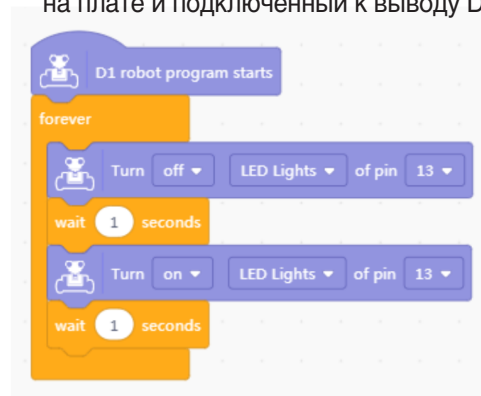
3. Выберите порт ПК, к которому подключаем Робота.



COM15-CH340 ▼

При удачном подключении имя канала появится в строке меню

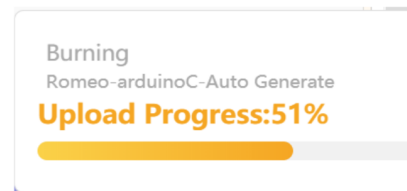
4. Соберите из блоков «стандартный» тестовый скетч Blynk, который включает и выключает светодиод, размещённый на плате и подключенный к выводу D13.



5. Загрузите скетч на плату.



Нажмите кнопку Upload



Когда индикатор загрузки покажет 100%, загрузка будет завершена.

Готово!

На плате с интервалом 1 секунда начнёт мигать встроенный синий светодиод

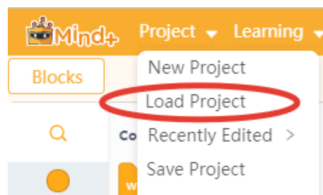


Проект 1. Мигаем встроенным светодиодом

Программа Blink стала де-факто первой программой для тестирования Arduino-совместимых контроллеров.

Мы уже применяли её, когда создавали новый скетч при тестировании подключения Робота к ПК.

1. Загрузите проект 01.Blynk.



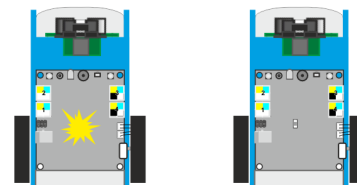
2. Подключите Робота к ПК, как показано на предыдущей странице.

COM15-CH340 ▾

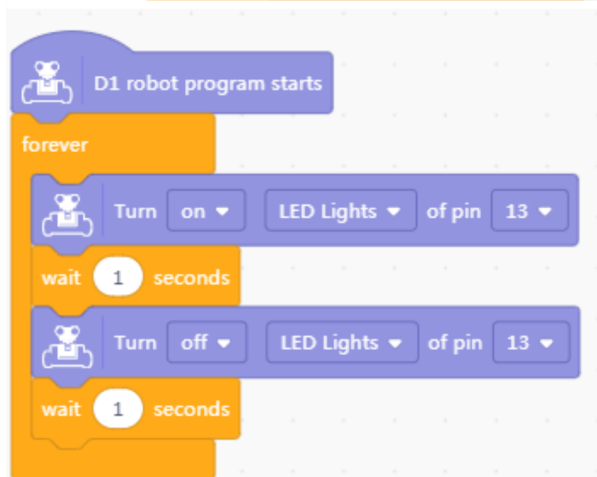
3. Загрузите скетч на плату.



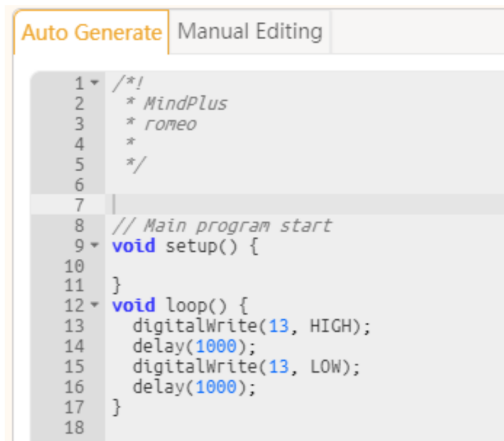
4. Начинает мигать встроенный светодиод с интервалом 1 секунда.



Скрипт 01.Blynk.sb3

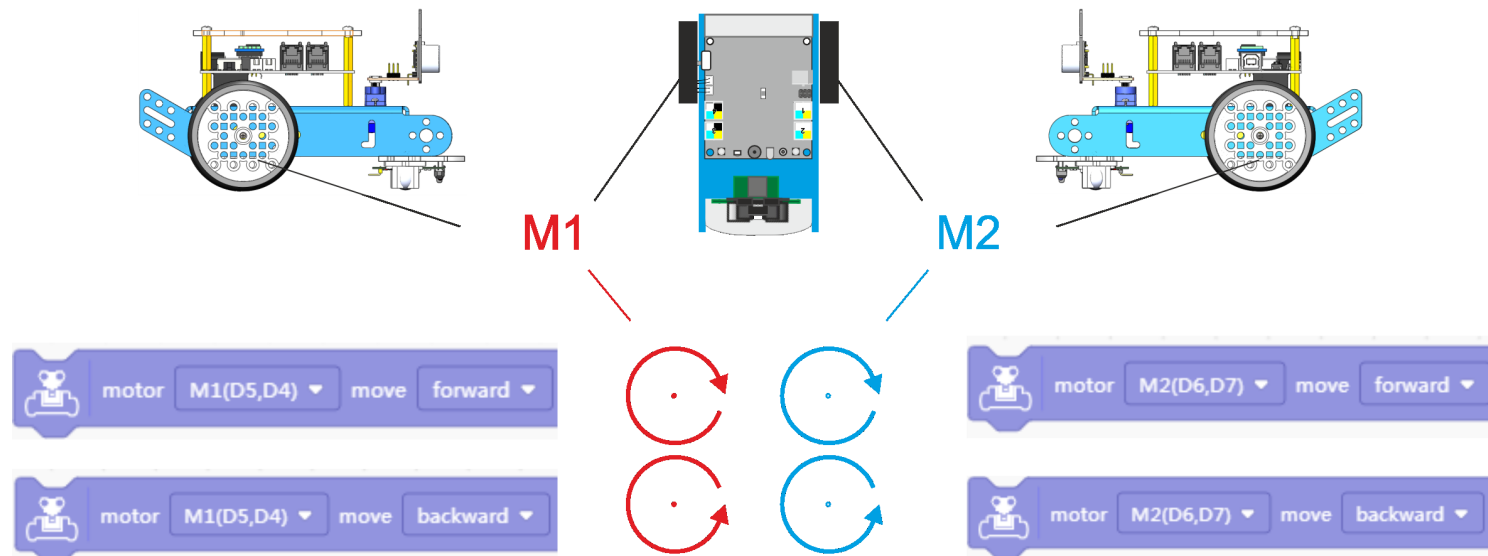


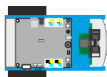
Код скрипта генерируется автоматически.



Проект 2. Управление моторами Робота

Схема коммутации на нашем Роботе несколько отличается от Romeo. Поэтому, если в модулях обоих моторов (M1 и M2) выбрать одновременно режим **forward** (или **backward**), колеса будут вращаться в разные стороны, а Робот начнёт кружиться на месте.





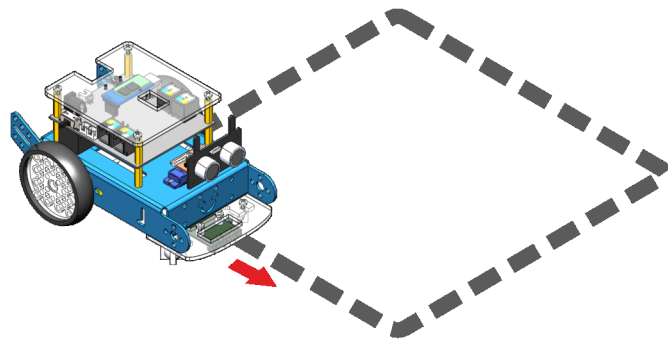
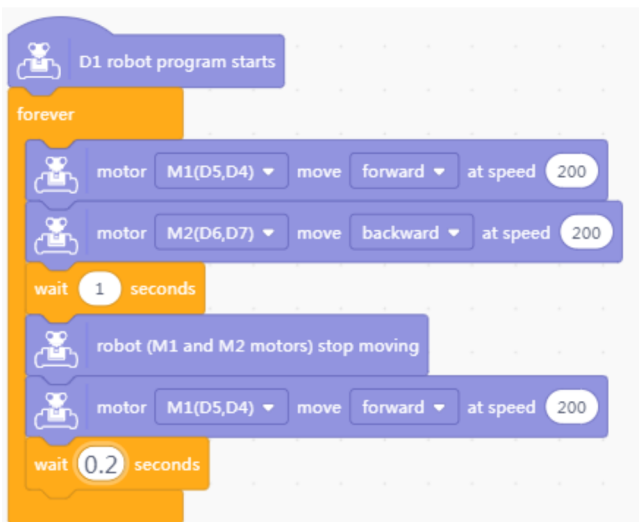
ВПЕРЕД



НАЗАД



Скрипт 02.Motor.sb3



Проект 3. Музыкальный Робот

С помощью зуммера, установленного на плате, Робот может воспроизводить различные музыкальные композиции.

В качестве примера рассмотрим известную композицию неизвестного автора «Чижик-пыжик».

Нажмите здесь, и появится всплывающее окно с фортепианной клавиатурой.



Нажмите на клавишу, чтобы выбрать ноту.



Скрипт 03.Music.sb3

D1 robot program starts

forever

- pin 8 play trumpet tone High E/E5 for Half beat
- pin 8 play trumpet tone High C/C5 for Half beat
- pin 8 play trumpet tone High E/E5 for Half beat
- pin 8 play trumpet tone High C/C5 for Half beat
- pin 8 play trumpet tone High F/F5 for Half beat
- pin 8 play trumpet tone High E/E5 for Half beat
- pin 8 play trumpet tone High D/D5 for Half beat

wait 0.5 seconds

Чижик-пыжик



Чи_жик пы_жик, где ты был?

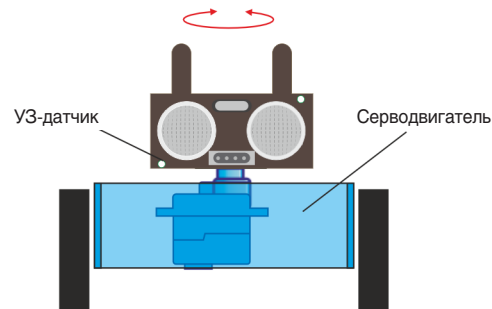
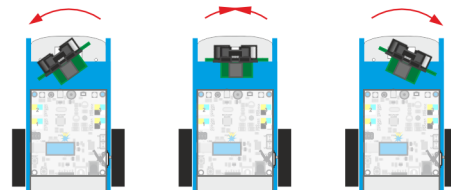
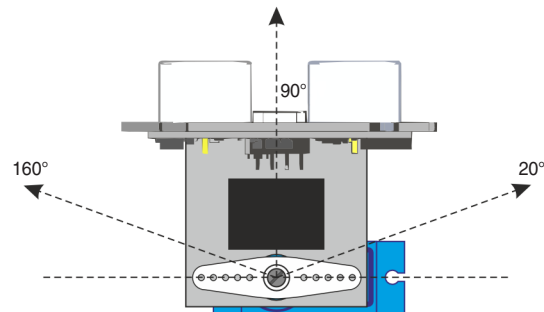
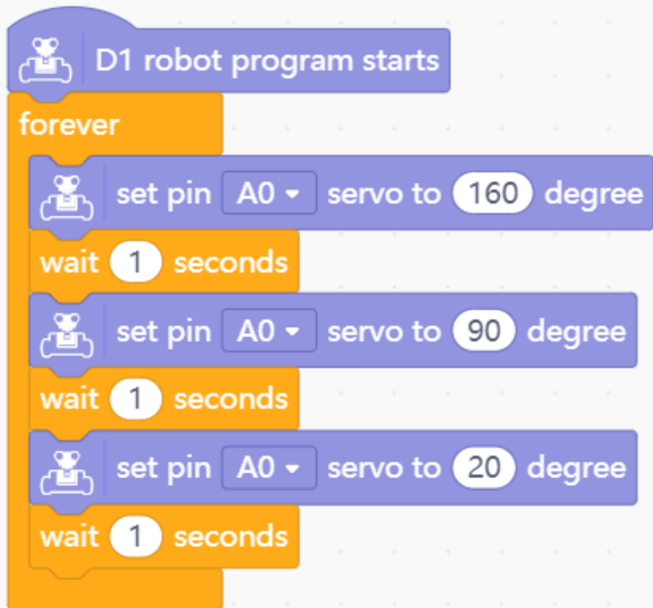


Проект 4. Вращаем головой

Голова Робота установлена на серводвигателе, который обеспечивает её вращение на заданный угол.



Скрипт 04.Servo.sb3



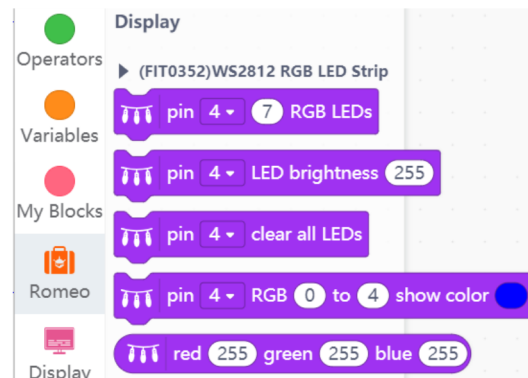
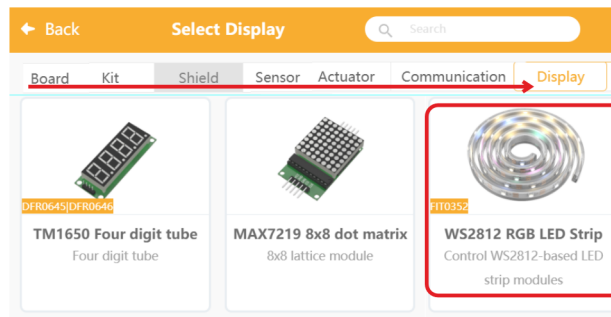
Проект 5. Включаем RGB-светодиоды

Для управления RGB-светодиодами на панель **Blocks** необходимо добавить группу специальных блоков (**FIT0352**) **WS2812 RGB LED Strip**.

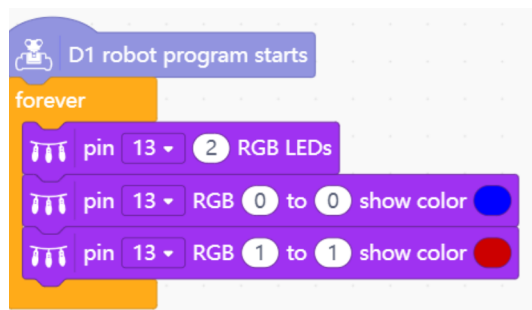
1. Щёлкните по кнопке выбора расширений **Extensions**.



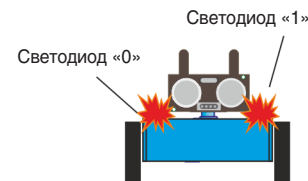
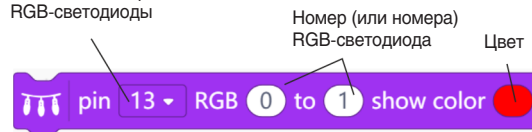
2. Выберите в меню **Display** блоки (FIT0352) WS2812 RGB LED Strip и нажмите **Back**



Скрипт 05.RGB_LED.sb3



Пин контроллера, к которому подключены адресные RGB-светодиоды



Проект 6. Передаём команды Роботу по последовательному порту

Связь с Роботом осуществляется по последовательному порту. Данные и команды могут поступать либо по кабелю USB, который «связывает» ПК и Робота, либо по каналу Bluetooth между смартфоном (Android)/ПК и модулем Bluetooth HC-05/06, установленным на Роботе.



Открыть последовательный порт



Скрипт 06.SerialEcho.sb3

```
D1 robot program starts
set serial-port baud rate to 9600
forever
  if Is there data to read on serial port? then
    serial output read serial data in origin , Wrap
```

upload success
A
B
C
0

Вводим символ для отправки через последовательный порт

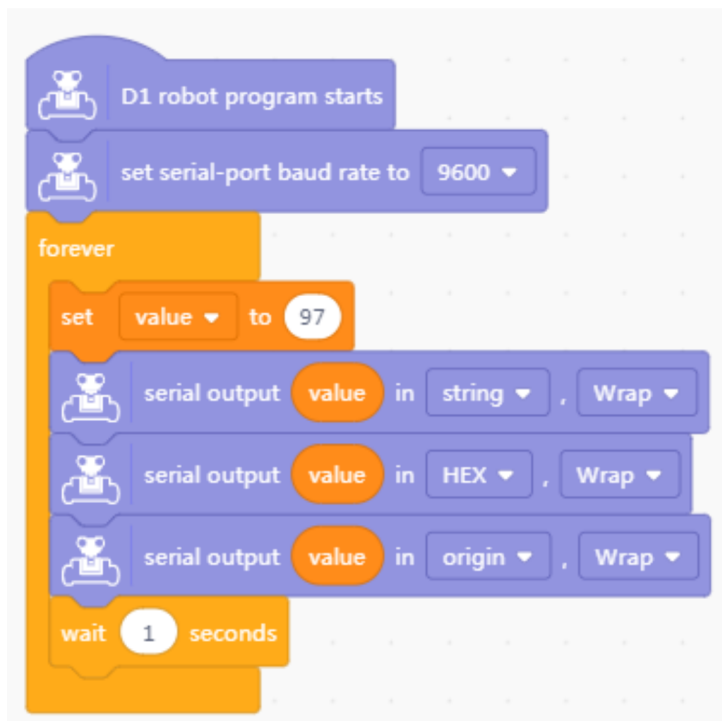
Кнопка для отправки

Проект 7. Передаём данные от Робота на ПК по последовательному порту

Робот может не только получать команды по последовательному порту, но и отправлять данные на ПК или смартфон.



Скрипт 07.SerialOutput.sb3



Открываем последовательный порт

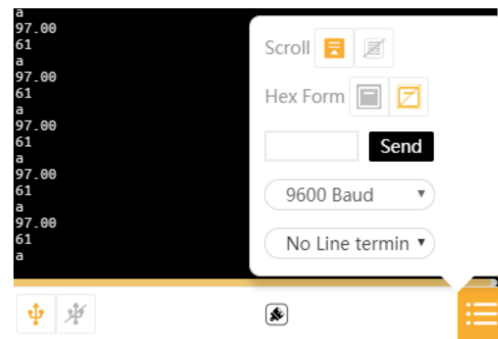


Получаем данные от Робота



Очистить вывод

Настройки

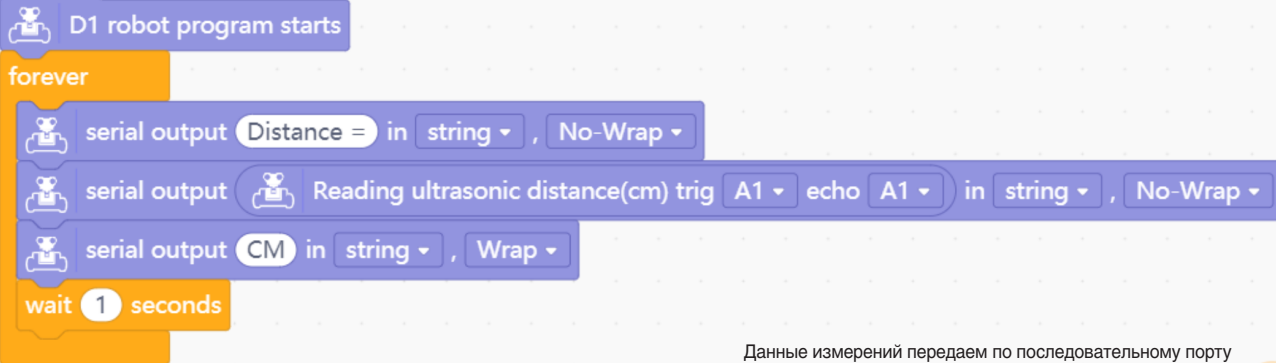


Проект 8. Измеряем расстояние

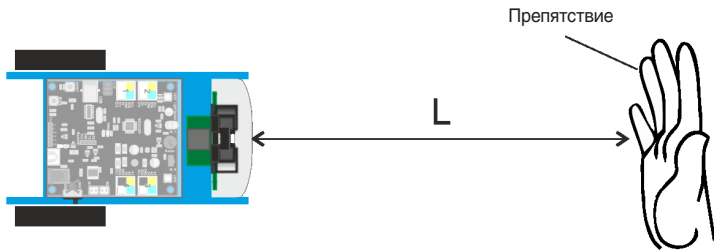
Для измерения расстояния используется УЗ-датчик HC-SR04.



Скрипт 08.ReadUltrasonic.sb3



Данные измерений передаем по последовательному порту (как в предыдущем проекте).

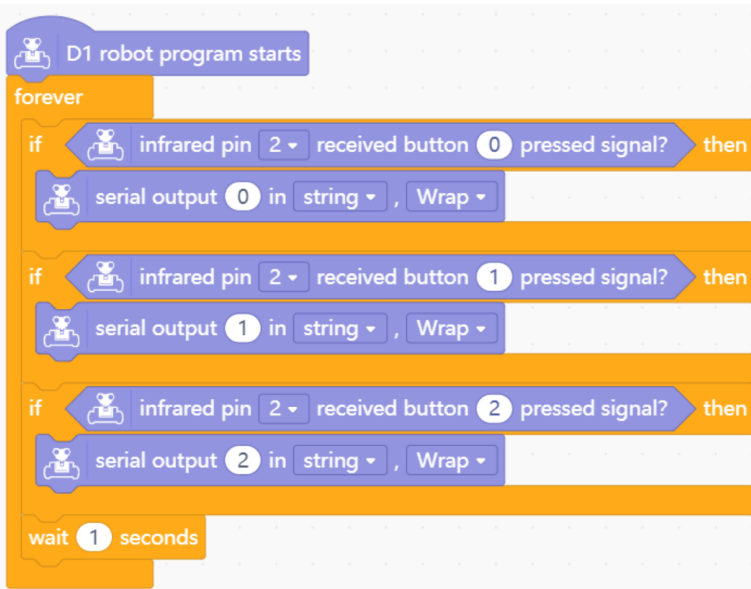


Открыть последовательный порт

Проект 9. Управляем Роботом с помощью ИК-пульта



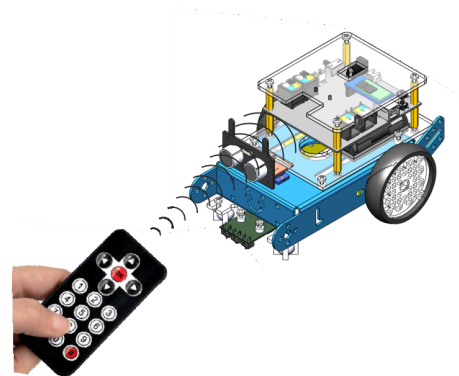
Скрипт 08.ReadUltrasonic.sb3



Auto Generate

Manual Editing

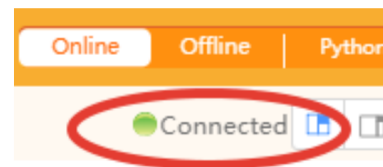
```
14 Serial.begin(9600);
15 remoteReceive_2.begin(2);
16 }
17 void loop() {
18
19   if (remoteReceive_2.keyPressed("FF4AB5")) {
20     Serial.println("0");
21   }
22   if (remoteReceive_2.keyPressed("FF6897")) {
23     Serial.println("1");
24   }
25   if (remoteReceive_2.keyPressed("FF9867")) {
26     Serial.println("2");
27   }
28   if (remoteReceive_2.keyPressed("FFB04F")) {
29     Serial.println("3");
30   }
31
32   delay(1000);
33 }
34
```



Проект 10. Осторожно, динозавр! Измеряем освещённость

Тестирование датчика освещенности Робота осуществим в интерактивном режиме. Напомним, что в этом режиме весь скрипт в «мозг» Робота не загружается.

Кнопки **Upload** нет. После подключения к порту (Connect Device) следует нажать кнопку **Connected**.



Скрипт.10. Lightsensor.sb3

when clicked

forever

set my variable to map read pin A6 Light Sensor from [0 , 1023] to [0 , 160]

say my variable

if my variable < 40 then

switch costume to Dinosaur4-d

switch costume to Dinosaur4-d

switch costume to Dinosaur4-d

pin 8 play trumpet tone Low D/D3 for Half beat

else

switch costume to Dinosaur4-a

