

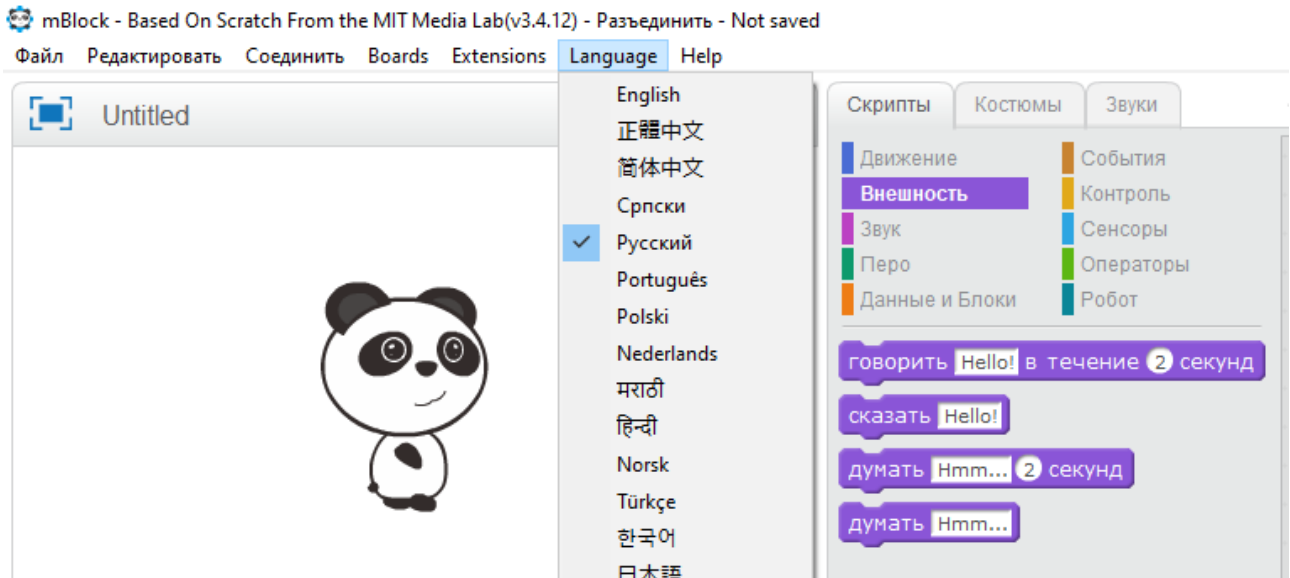
Программирование в среде mBlock

Выполнять лабораторные работы «Конструктора программируемых моделей инженерных систем» будем в среде программирования mBlock3.

1. **Официальная страница загрузки mBlock** —

<https://www.mblock.cc/en-us/download>

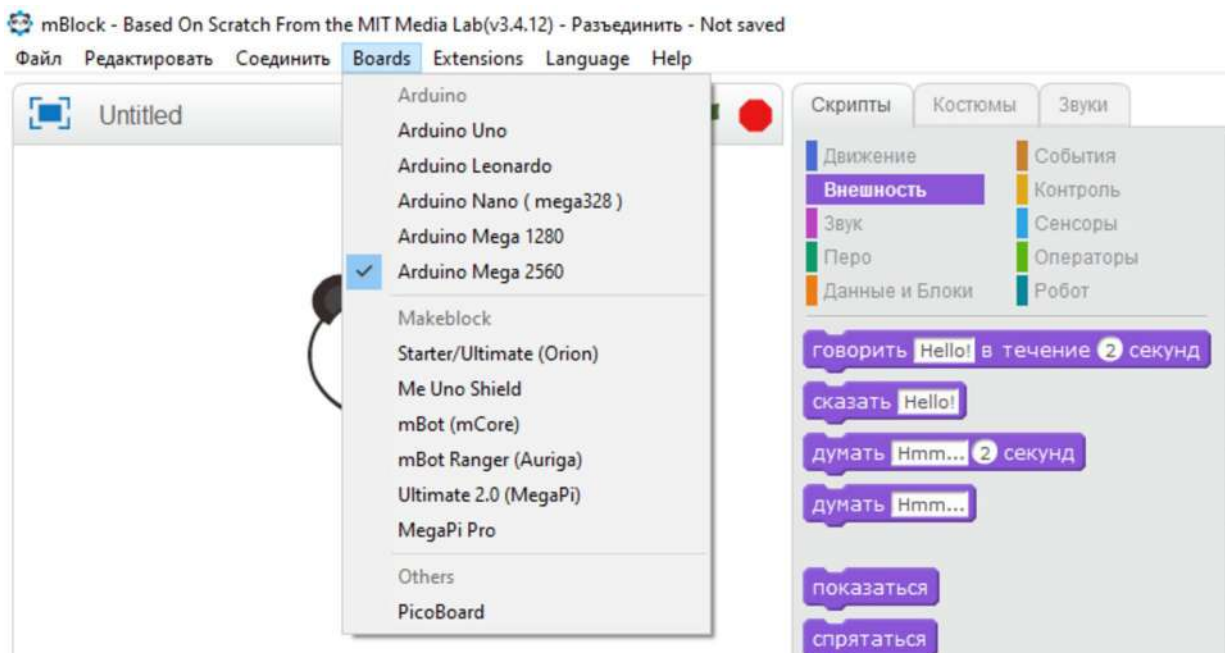
2. Для изменения языка интерфейса служит команда главного меню **Language**



3. **Выбираем плату контроллера**

Далее необходимо выбрать правильную плату контроллера, чтобы программа mBlock использовала соответствующую систему команд. Для выбора

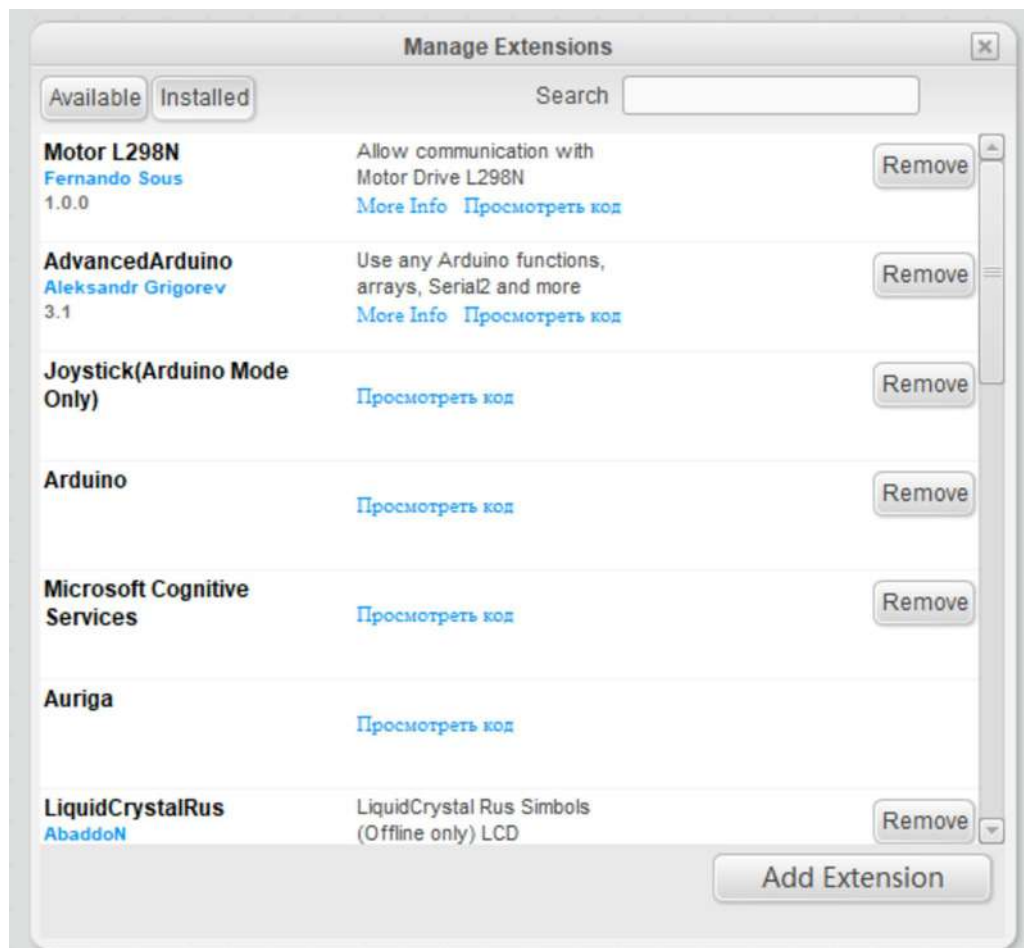
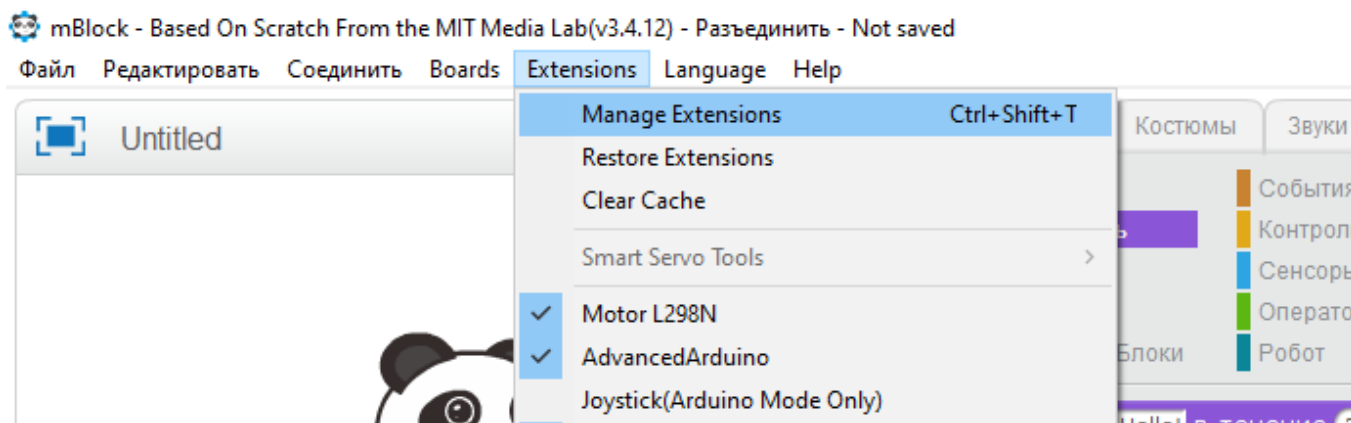
откройте в главном программном меню пункт **Boards**. В этом меню перечислены платы, с которыми умеет работать mBlock.



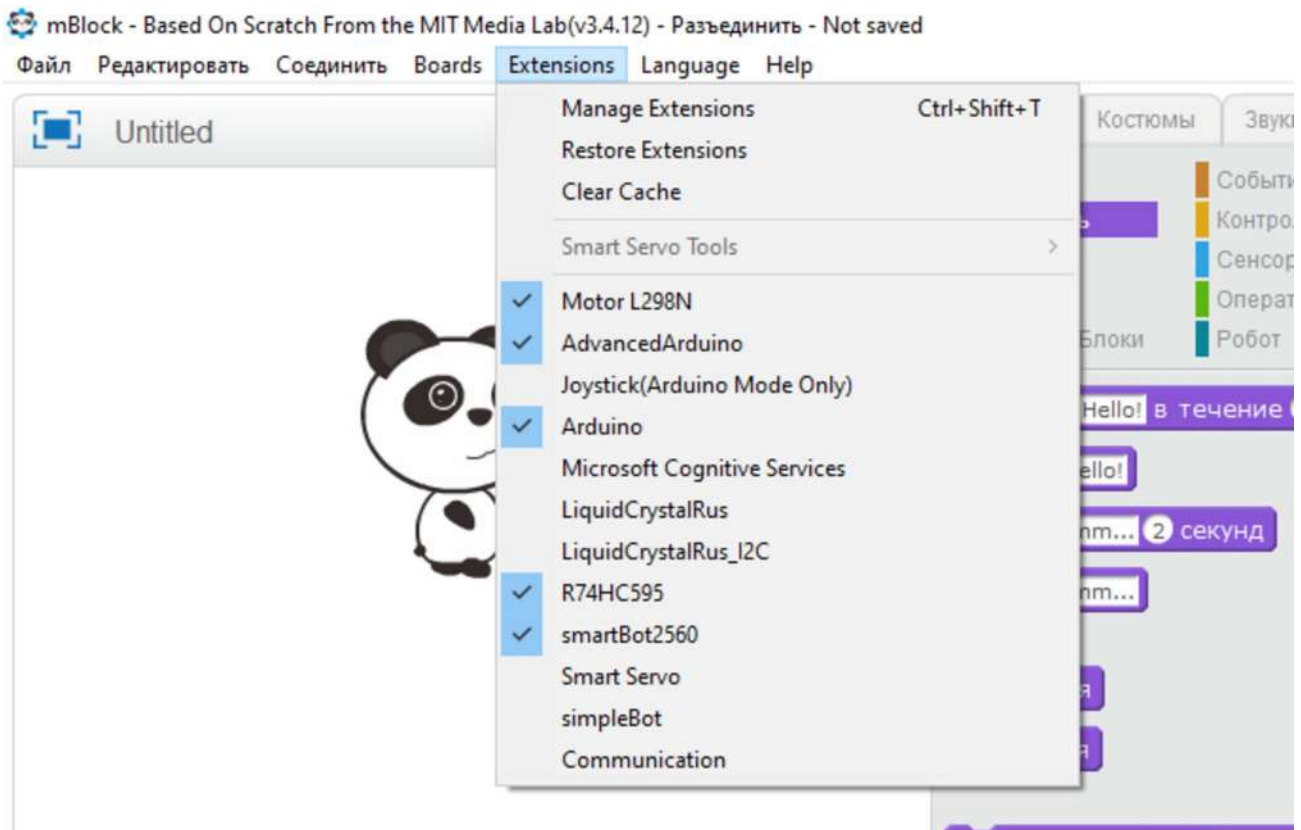
4. Добавление расширений в mBlock3

Программа mBlock имеет возможность добавления сторонних компонентов для управления роботизированными комплексами. Само дополнение должно быть представлено в ZIP формате.

Рассмотрим, как установить расширение из архивного файла, сохранённого на диске. Следует открыть в главном меню пункт **Extensions**, а в нем **Manage Extensions** и нажать расположенную внизу кнопку **Add Extension** (Добавить расширение). В появившемся окне необходимо выбрать тип загруженного файла (в нашем случае это **zip file (*.zip)**), указать его расположение и нажать кнопку **Открыть**. Готово

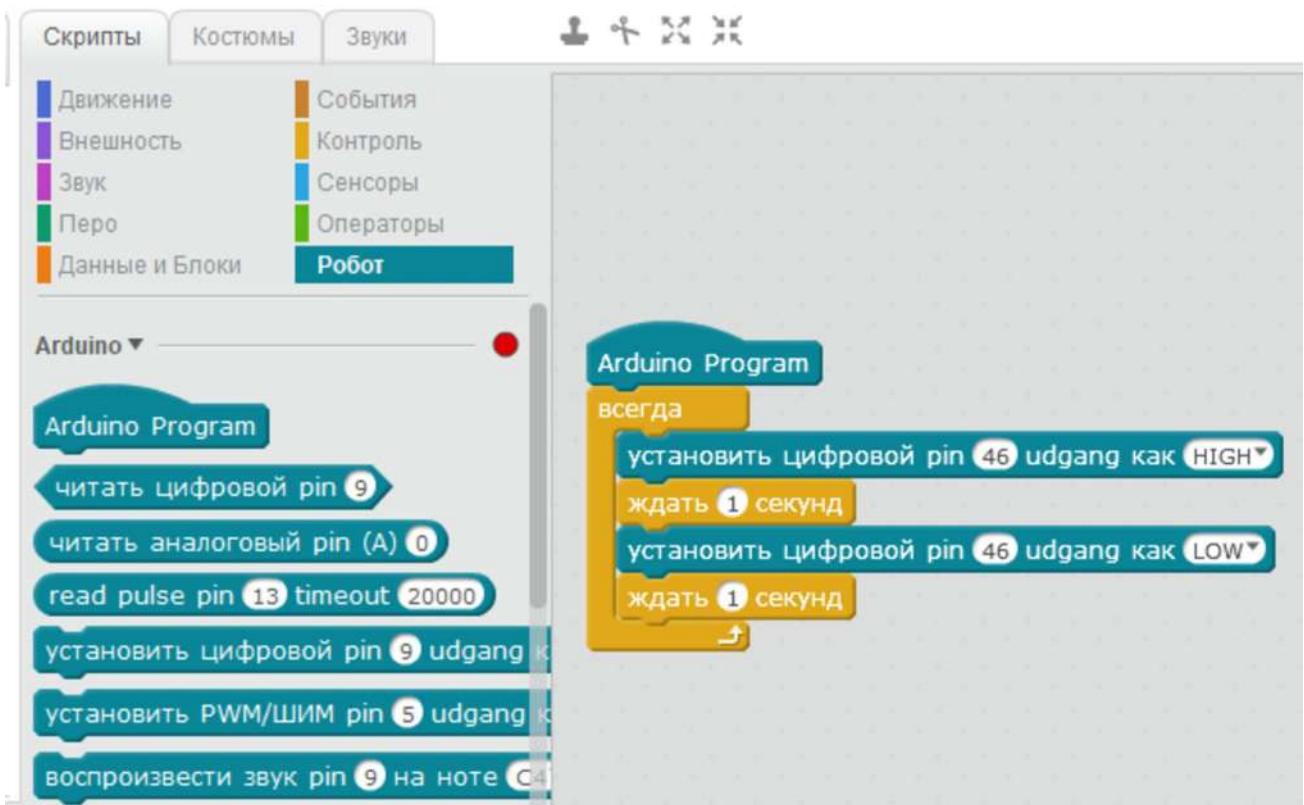


Установленное расширение появится в разделе **Manage Extensions** пункте **Extensions** главного меню.



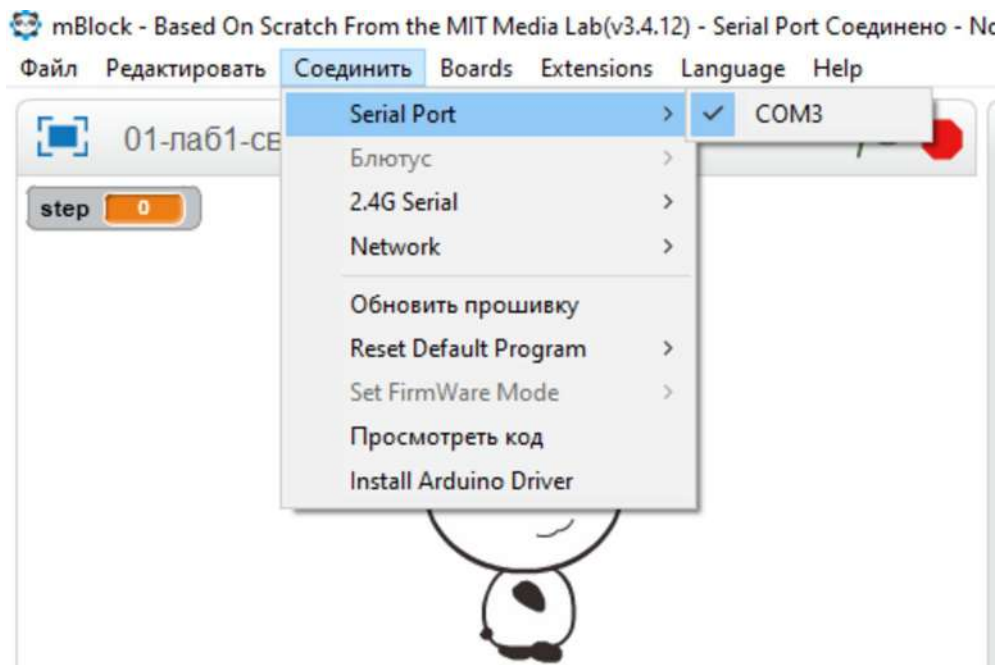
5. Пример реализации кода в mBlock3

а) С помощью блоков расширения и команд Scratch создаем программу решения задачи.

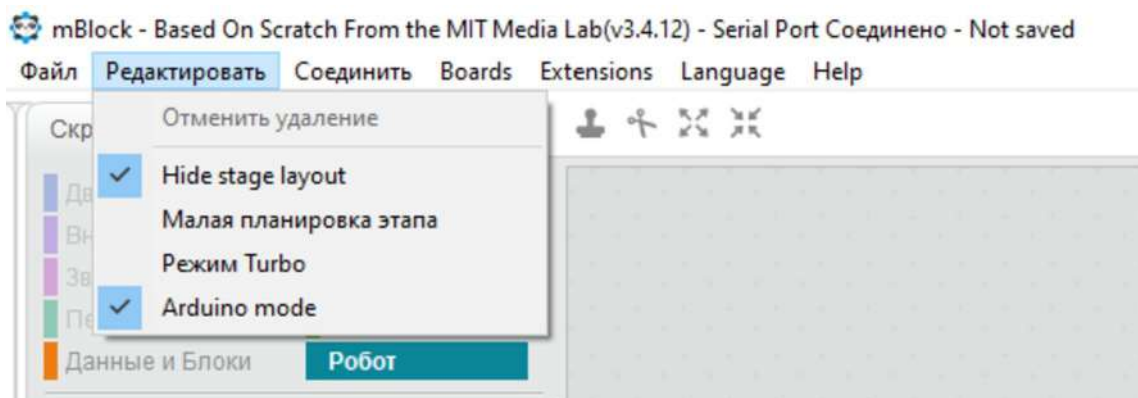


б) Подключите плату (робота) USB-кабелем к компьютеру. В списке доступных коммуникационных портов появится новый пункт . Выберите этот порт при помощи мыши или клавиатуры.

Откройте список коммуникационных портов снова, вы увидите, что установлен маркер напротив соответствующего пункта меню, а заголовок программы mBlock изменился и уведомляет нас о том, что соединение установлено.



в) Перейдите в режим **Arduino mode**, используя меню **Редактировать**



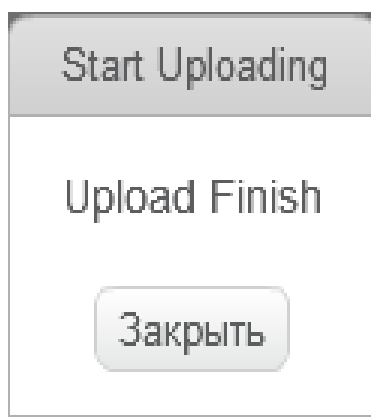
г) Справа, в окне, можно посмотреть вашу программу, но уже не в виде блоков, а в текстовом виде. Прокрутите программу, посмотрите, как записаны ваши блоки в виде текста.

Нажмите кнопку **Upload to Arduino** , понаблюдайте за процессом загрузки программы в память контроллера робота.

The image shows the Arduino IDE interface. On the left, a block-based program is visible, starting with 'Arduino Program' and a 'всегда' (always) loop containing: 'установить цифровой pin 46 udgang как HIGH', 'ждать 1 секунд', 'установить цифровой pin 46 udgang как LOW', and 'ждать 1 секунд'. On the right, the corresponding C++ code is displayed, including headers for Arduino, Wire, and SoftwareSerial, and functions for setup, loop, and a custom delay function.

```
1 #include <Arduino.h>
2 #include <Wire.h>
3 #include <SoftwareSerial.h>
4
5 double angle_rad = PI/180.0;
6 double angle_deg = 180.0/PI;
7
8 void setup() {
9     pinMode(46, OUTPUT);
10 }
11
12 void loop() {
13     digitalWrite(46, 1);
14     _delay(1);
15     digitalWrite(46, 0);
16     _delay(1);
17     _loop();
18 }
19
20 void _delay(float seconds) {
21     long endTime = millis() + seconds * 1000;
22     while (millis() < endTime) _loop();
23 }
24
25 void _loop() {
26 }
```

д) Если процесс загрузки прошел корректно, вы увидите сообщение о завершении загрузки



Всё, программа в памяти контроллера

Литература

Григорьев, А. Т. Робототехника в школе и дома. Книга проектов / А. Т. Григорьев, Ю. А. Винницкий. — СПб.: БХВ-Петербург, 2022. — 240 с.: ил.