
**ХІ Республіканская навучна-практычная канферэнцыя-конкурс
навучна-даследавальскіх работ учащихся средних,
средних специальных учебных заведений и студентов вузов
«От Альфа к Омеге...» (с международным участием)
Секция 3. Компьютерные науки и программирование
НАУЧНО-ИССЛЕДОВАТЕЛЬСКИЕ РАБОТЫ ШКОЛЬНИКОВ**

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

Государственное учреждение образования «Высоковская средняя школа» Каменецкого района

КВАДРОПОД НА ОСНОВЕ МИКРОКОМПЬЮТЕРА MICRO:BIT

Пашкевич Максим Владимирович,

Янчук Егор Витальевич,

учащиеся 7 «А» класса,

Туз Глеб Сергеевич,

учащийся 8 «В» класса,

Лешкевич Александр Николаевич,

учитель математики и информатики

ГУО «Высоковская средняя школа»

Каменецкого района,

высшая кв. категория учителя математики и

информатики

**ХІ Республіканская навучна-практычная канферэнцыя-конкурс
навучна-даследацкіх работ учасніц сярніх,
сярніх спецыяльных учебных заведений і студэнтаў вузав
«От Альфа к Омеге...» (с міжнародным удзелам)
Секцыя 3. Камп'ютэрныя навукі і праграмаванне
РЕФЕРАТЫ НАУЧНО-ИССЛЕДОВАТЕЛЬСКИХ РАБОТ ШКОЛЬНИКОВ**

КВАДРОПОД НА ОСНОВЕ МИКРОКОМПЬЮТЕРА MICRO:BIT

М. В. Пашкевич, Г. С. Туз, Е. В. Янчук

*ГУО «Высоковская средняя школа» Каменецкого района, 7 «А» класс, 8 «В» класс, 7 «А»
класс
Высокое, Беларусь*

Научный руководитель – А. Н. Лешкевич, учитель математики и информатики ГУО «Высоковская средняя школа» Каменецкого района, высшая кв. категория учителя математики и информатики.

Работа 13 с., 5 ч., 12 рис., 6 источников.

Ключевые слова: квадропод, модель, micro:bit.

В работе исследуется использование платы Micro:bit для создания роботов, а также программирование робота в среде MakeCode. При сборке модели робота используются детали из конструкторов Lego Mindstorms EV3 и Lego Wedo 2.0. Управление моделью организовано с помощью джойстика. Программирование робота и джойстика происходит с помощью блоков.

Объектом исследования являются возможности платы Micro:bit для создания роботов.

Цель исследования – создать модель квадропода на базе платы Micro:bit с управлением при помощи джойстика.

В результате исследования были получены следующие результаты:

собранный и запрограммированный модель робота квадропода, которая может осуществлять движение в четырех направлениях под управлением джойстика.

Данную разработку в перспективе можно использовать для организации движения в труднодоступных для человека местах на Земле (пустыня, горная поверхность) с определенной целью (доставка груза, исследование территории). Модель можно использовать в соревнованиях по робототехнике в категории «Гонки шагающих роботов».

ОГЛАВЛЕНИЕ

Введение.....	4
1 Микрокомпьютер Micro:bit.....	5
2 MakeCode как среда программирования.....	8
3 Сборка модели квадропода.....	10
4 Программирование квадропода.....	14
5 Программирование джойстика.....	16
Заключение.....	17
Список использованных источников	18

ВВЕДЕНИЕ

Люди, которые умеют писать, смотрят на мир несколько иначе, чем те, кто умеет только читать. Так и программирование позволяет по-новому посмотреть на мир, где технологии играют все большую роль. Сегодня важно не только уметь пользоваться компьютерными программами и мобильными приложениями, но и понимать, как все устроено, — потому что это открывает новые возможности.

Уже много лет мы читаем в интернете и книгах, слышим по телевизору, что скоро нас будут окружать умные, добрые и интересные роботы. Однако в реальной жизни роботов не так много, как хотелось бы. Работа с платой Micro:bit – это первый шаг к созданию «умных» устройств и машин.

Цель исследования – создать модель квадропода на базе платы Micro:bit с управлением при помощи джойстика.

Задачи работы:

- ознакомиться с возможностями среды программирования MakeCode;
- собрать модель квадропода на базе платы Micro:bit;
- запрограммировать платы Micro:bit.

Методы исследования: изучение среды программирования, анализ, алгоритмизация, моделирование.

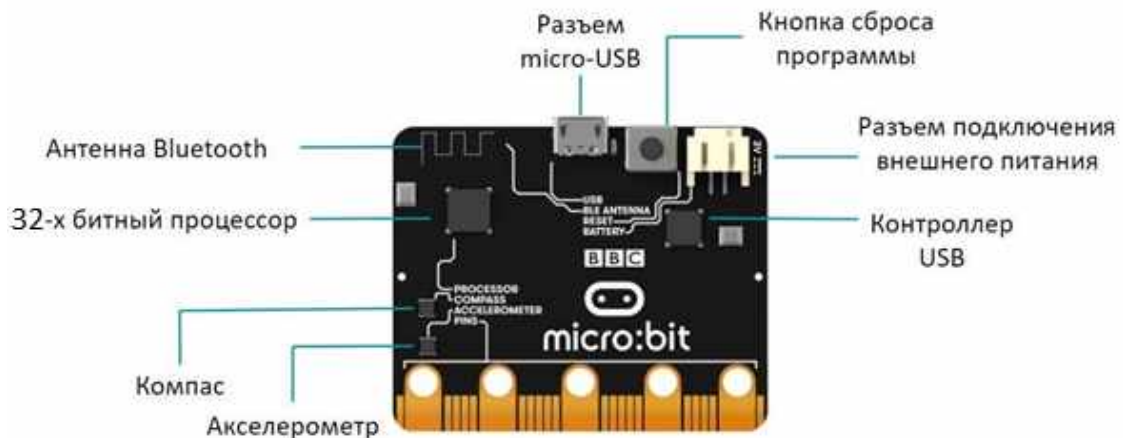
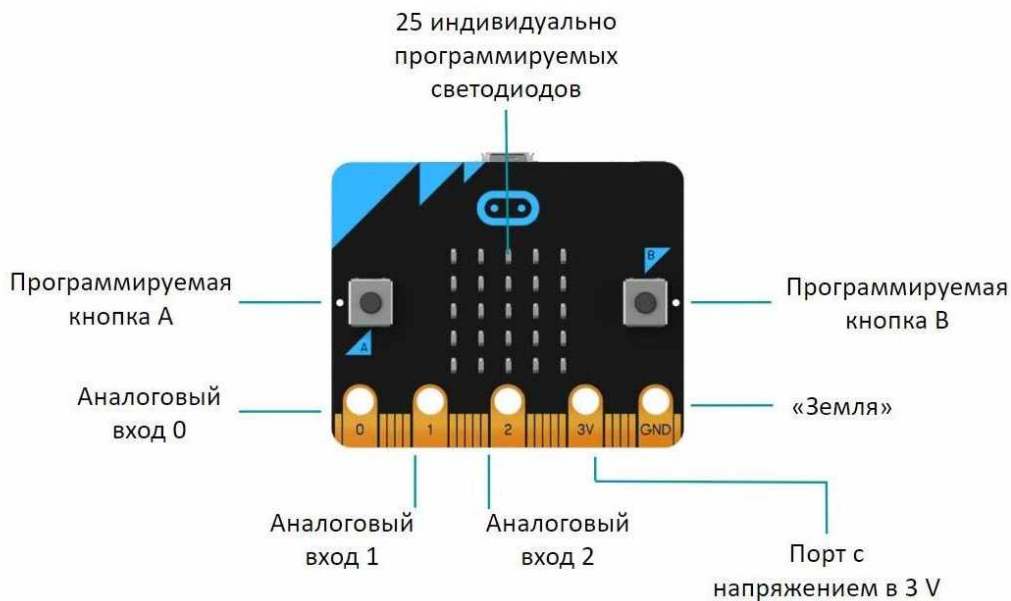
Необходимое оборудование: модель квадропода на базе платы Micro:bit, джойстик с платой Micro:bit.

1 МИКРОКОМПЬЮТЕР MICRO:BIT

Плата Micro:bit создана в 2015 году компанией BBC и ее партнерами, включая компанию Microsoft, размеры платы 4×5 см.

Плата Micro:bit включает:

- § дисплей в виде матрицы из 25 светодиодов (5×5);
- § программируемые кнопки A и B;
- § контакты ввода-вывода (0 – 2 – аналоговые, 3V, GND, 23 малых контакта) (см. рисунок 1.1);
- § порт micro-USB для прошивки программы на плате, питания от компьютера;
- § микроконтроллер (32-х битный процессор ARM Cortex-M0 с тактовой частотой 16 мегагерц, 256 КБ Flash-памяти и 16 КБ оперативной памяти);
- § кнопка сброса RESET;
- § компас;
- § акселерометр (позволяет отслеживать движения платы);
- § радиомодуль для беспроводной связи с другими устройствами (Bluetooth 4.1 BLE);
- § разъем для подключения внешнего источника питания (см. рисунок 1.2) [2].



Существует плата Nano:bit, которая является аналогом платы Micro:bit (см. рисунок 1.3). Она имеет небольшие размеры, внешне и по своим характеристикам напоминает плату Arduino Nano.

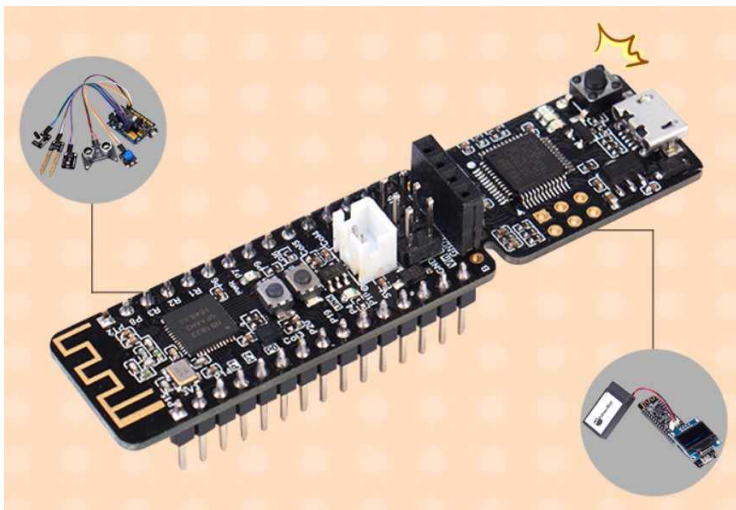


Рисунок 1.3 – Nano:bit

Внешние устройства подключаются к пронумерованным портам (пинам). Существует достаточно большое количество плат-расширений для платы Micro:bit, использование которых позволяет создавать различные интересные проекты:
§ базовая плата расширения (см. рисунок 1.4);



Рисунок 1.4 – Базовая плата расширения

§ плата расширения Super:Bit (см. рисунок 1.5);

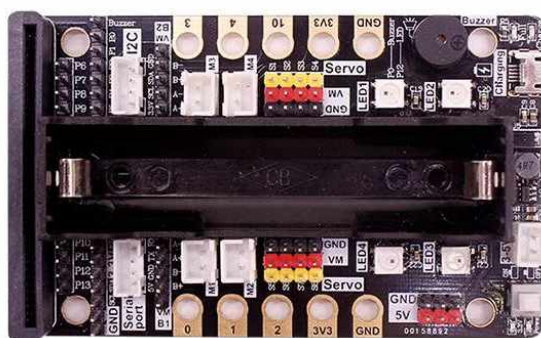


Рисунок 1.5 – Плата расширения Super:Bit

§ плата расширения «Крокодил» (см. рисунок 1.6);

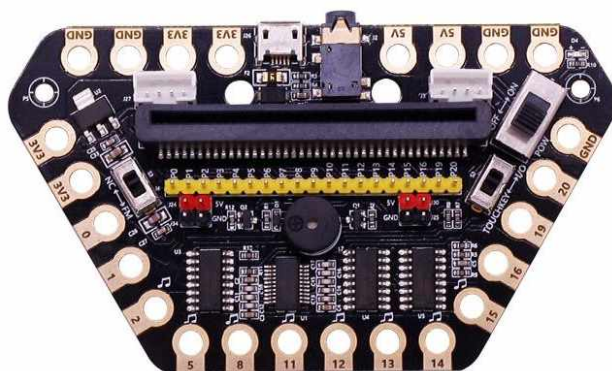


Рисунок 1.6 – Плата расширения «Крокодил»

§ плата расширения Robot:bit (см. рисунок 1.7).



Рисунок 1.7 – Плата расширения «Robot:bit»

2 MAKECODE КАК СРЕДА ПРОГРАММИРОВАНИЯ

Для программирования платы Micro:Bit необходимо подключение к сети интернет.

Для программирования платы можно применять различные среды: MakeCode, Scratch 3.0. Однако более удобной является среда разработки MakeCode, в которой есть возможность использовать блоки для программирования (см. рисунок 2.1) [2].

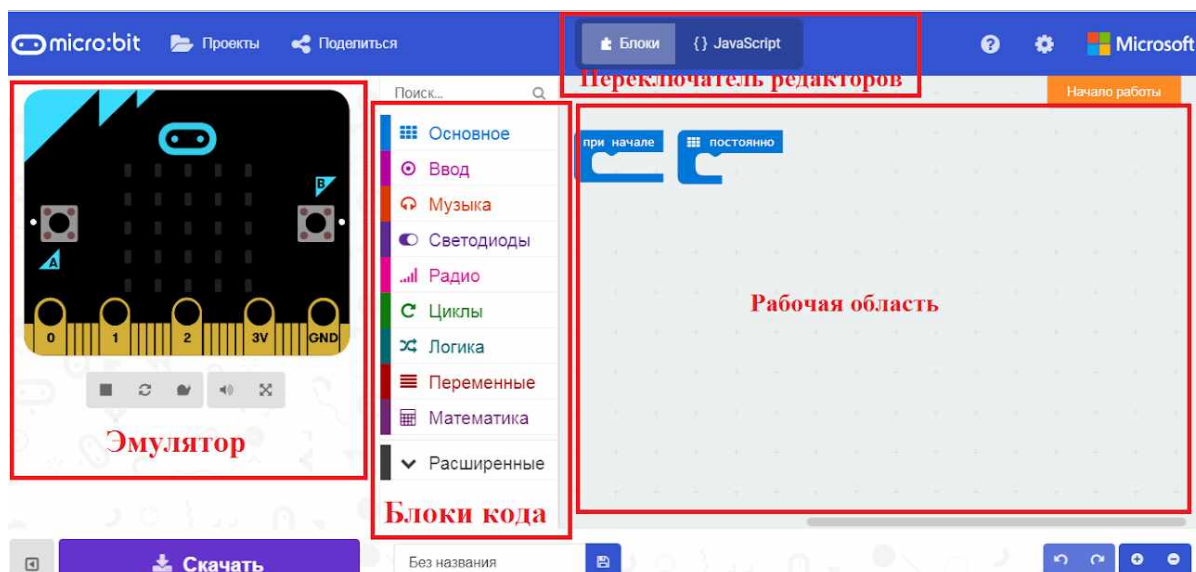


Рисунок 2.1 – Окно среды программирования MakeCode

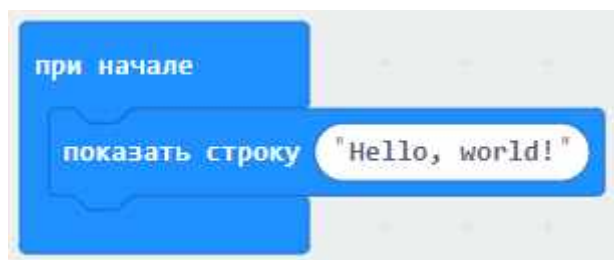
§ Переключатель редакторов. В MakeCode есть возможность программировать плату с помощью блоков (больше подходит для начинающих), «JavaScript» и «Python». В своей работе мы использовали редактор «Блоки».

§ Рабочая область, на которую переносятся блоки, или пишется код программы.

§ Блоки кода. Все блоки кодов упорядочены по разделам для более удобного поиска необходимого скрипта.

§ Эмулятор. Позволяет проверить работу программы без использования платы. Эмулятор платы Micro:bit, позволяет не только проверить работу программу, но и получить обратную связь о том, как работает программа. Эмулятор имеет полную поддержку всех возможностей платы Micro:bit.

При переключении режима «Блоки» на «JavaScript» или «Python» генерируется код программы на выбранный язык программирования (см. рисунок 2.2).



```
basic.show_string("Hello, world!")
```

Рисунок 2.2 – Режим редактора

Для программирования в MakeCode используется drag-and-drop-подход: блоки из палитры блоков перетаскиваются в рабочую область.

По функциональному назначению блоки делятся на 8 групп. Для удобства принадлежность блока к той или иной группе обозначается его цветом (см. рисунок 2.3) [2].

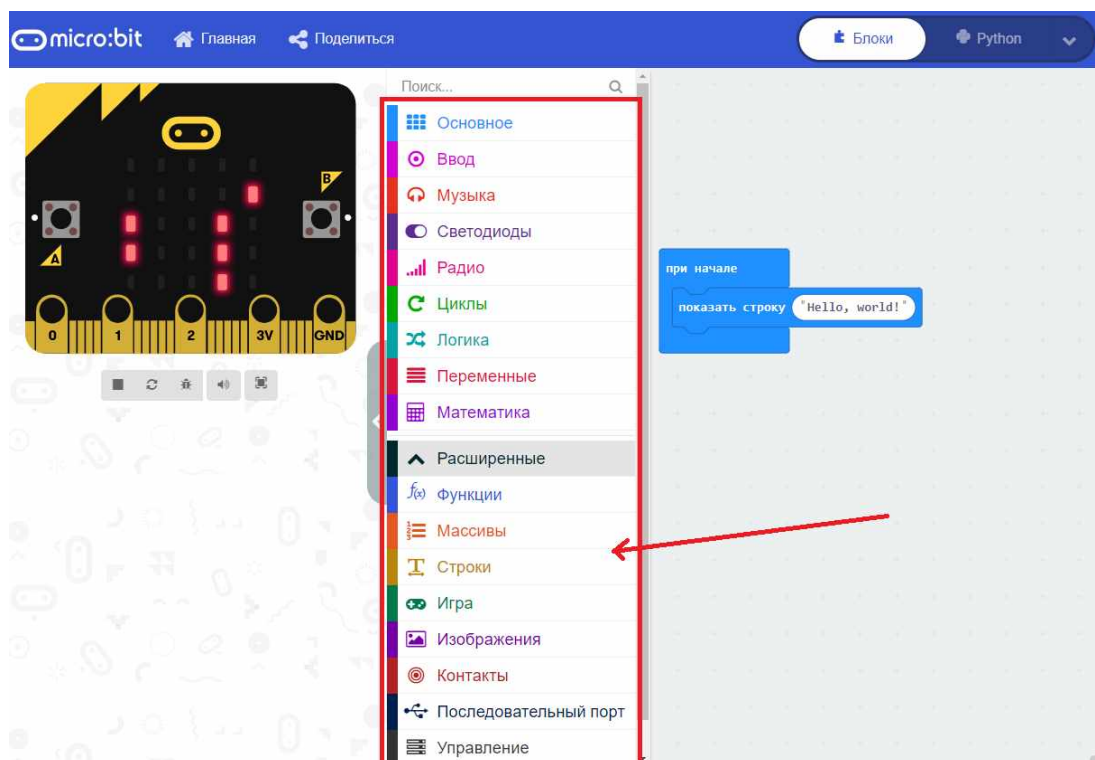


Рисунок 2.3 – Функциональные блоки

У многих блоков имеется редактируемое поле для вводимых программистом параметров.

Блоки бывают трёх видов: блоки стека, блоки заголовков и блоки ссылок.

Блоки стека сверху имеют выемку, а снизу — выступ, с их помощью они объединяются в группу блоков, называемую стеком. Стеки можно копировать и перемещать как единый блок. Особой разновидностью блоков стека являются управляющие конструкции, такие как циклы — они имеют С-образную форму, и могут охватывать собой вложенный стек блоков.

Блоки заголовков имеют выпуклый верхний край и выступ для объединения снизу — они образуют заголовки стеков. К блокам-заголовкам относятся блоки из группы «Ввод», позволяющие организовать обработчики сообщений.

Наконец, блоки ссылок предназначены для заполнения внутренних полей других блоков.

В среде MakeCode можно оперировать числами, текстовыми строками, логическими значениями, а также списками, играющими роль динамических массивов.

3 СБОРКА МОДЕЛИ КВАДРОПОДА

Сборка модели квадропода осуществлялась с использованием комплектов: Lego Wedo 2.0, Lego Mindstorms EV3 (см. рисунок 3.1).

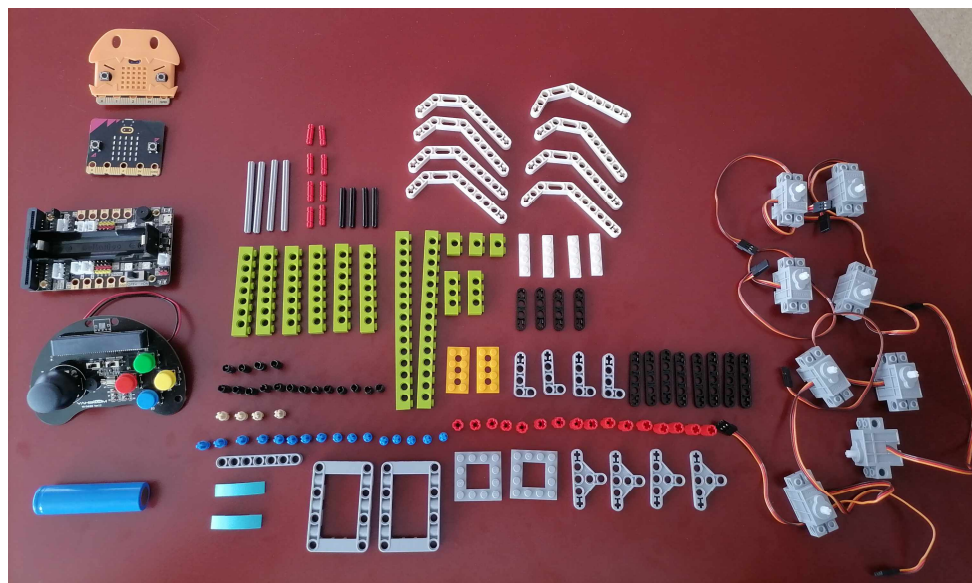


Рисунок 3.1 – Детали для сборки квадропода

Для модели квадропода использовались детали конструкторов Lego, плата расширения Super:Bit (см. рисунок 3.2), восемь lego-совместимых серводвигателей Robot:bit (см. рисунок 3.3). Выбор платы расширения Super:Bit был обусловлен наличием на ней 8 выходов для подключения сервоприводов и возможностью установки источника питания робота.

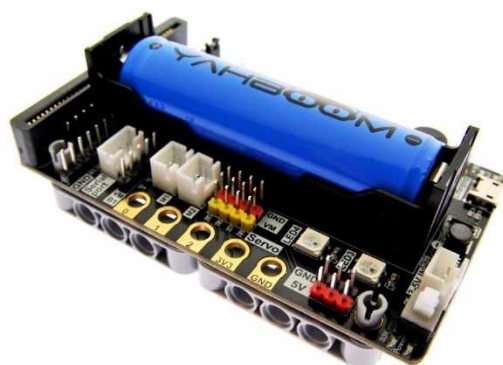


Рисунок 3.2 – Плата расширения Super:Bit

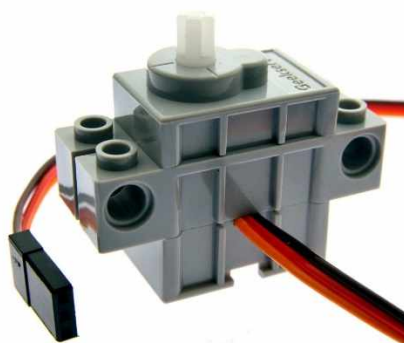


Рисунок 3.3 – Сервопривод Robot:Bit

На первом шаге были собраны четыре ноги робота (см. рисунок 3.4).

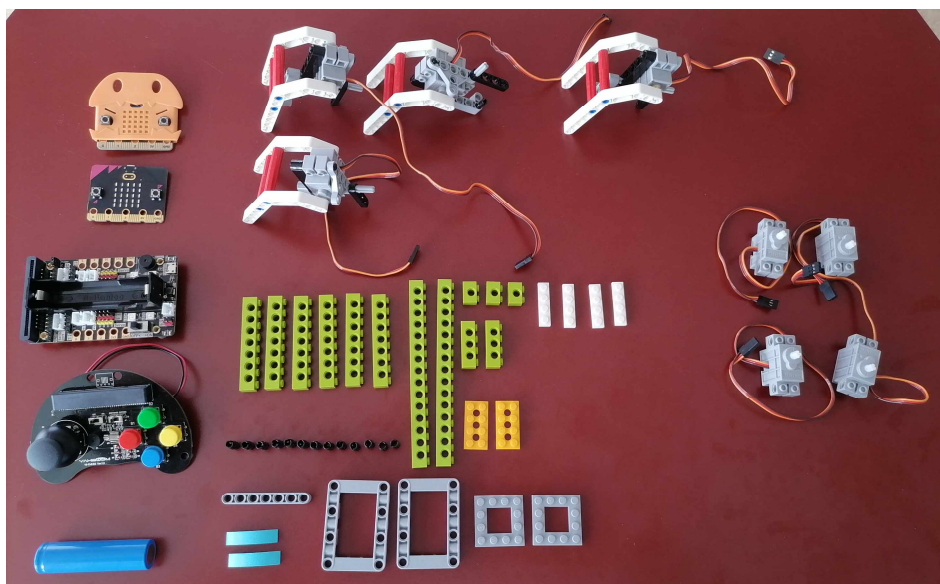


Рисунок 3.4 – Ноги робота в собранном состоянии

На следующем шаге была собрана платформа для крепления платы Super:Bit (см. рисунки 3.5, 3.6).

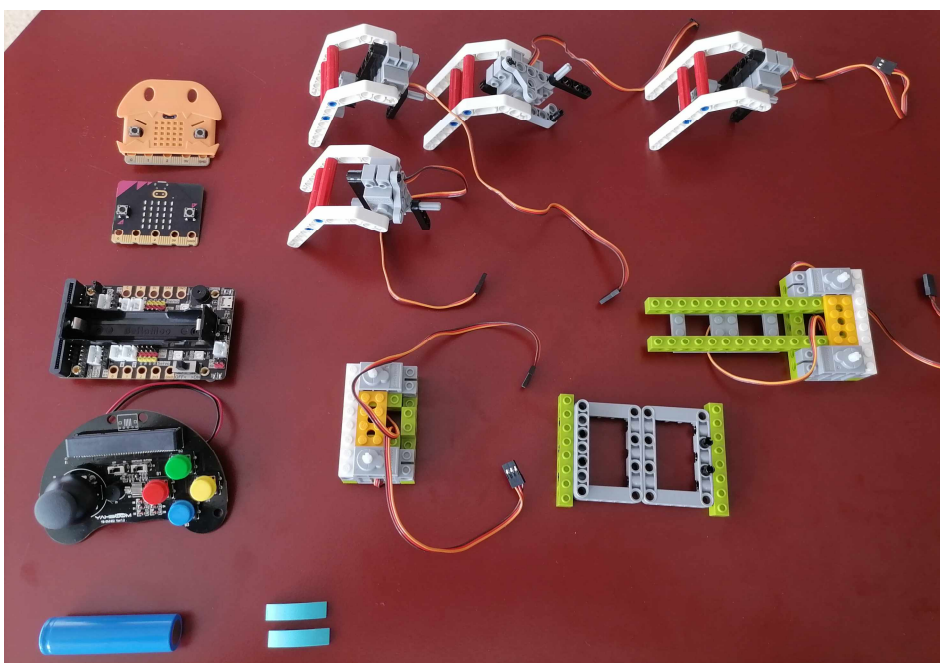


Рисунок 3.5 – Сборка платформы для крепления платы

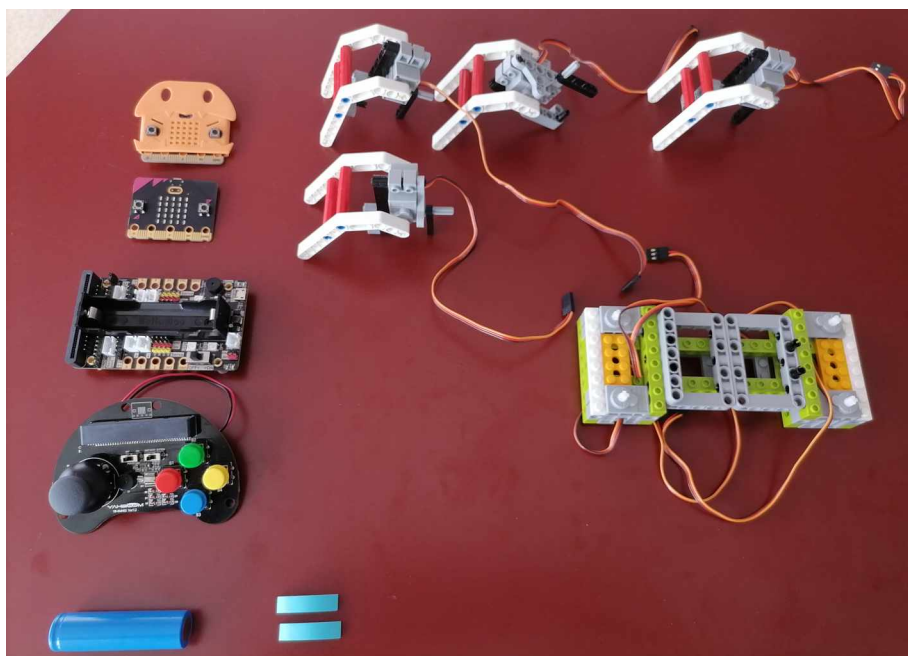


Рисунок 3.6 – Платформа для крепления платы

В верхней части платформы закрепили Micro:bit (см. рисунок 3.7).

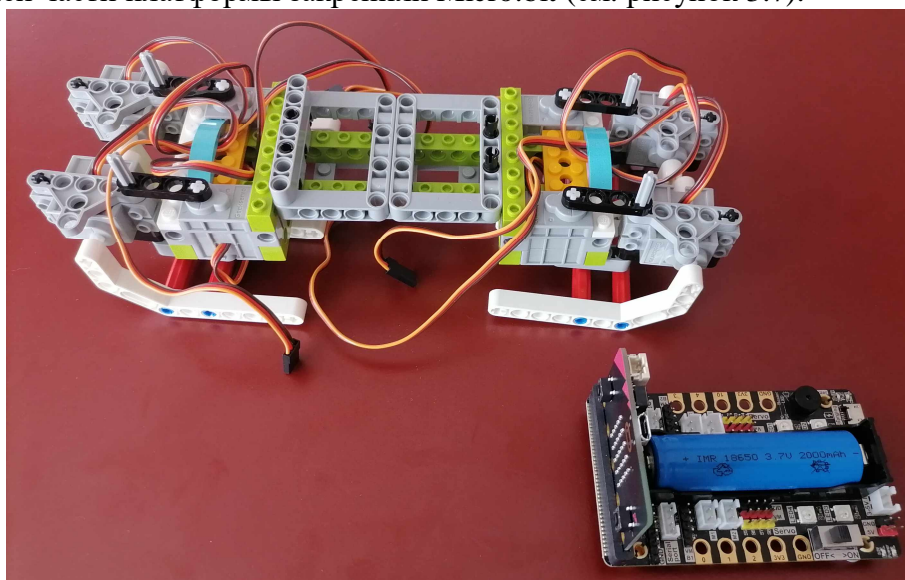


Рисунок 3.7 – Корпус робота

Все провода присоединялись к плате расширения Super:Bit.

Для управления двигателями на плате расширения использовались пины S1, S2, S3, S4 (правые двигатели) и S5, S6, S7, S8 (левые двигатели). Питание двигателям идет от платы (см. рисунок 3.8).

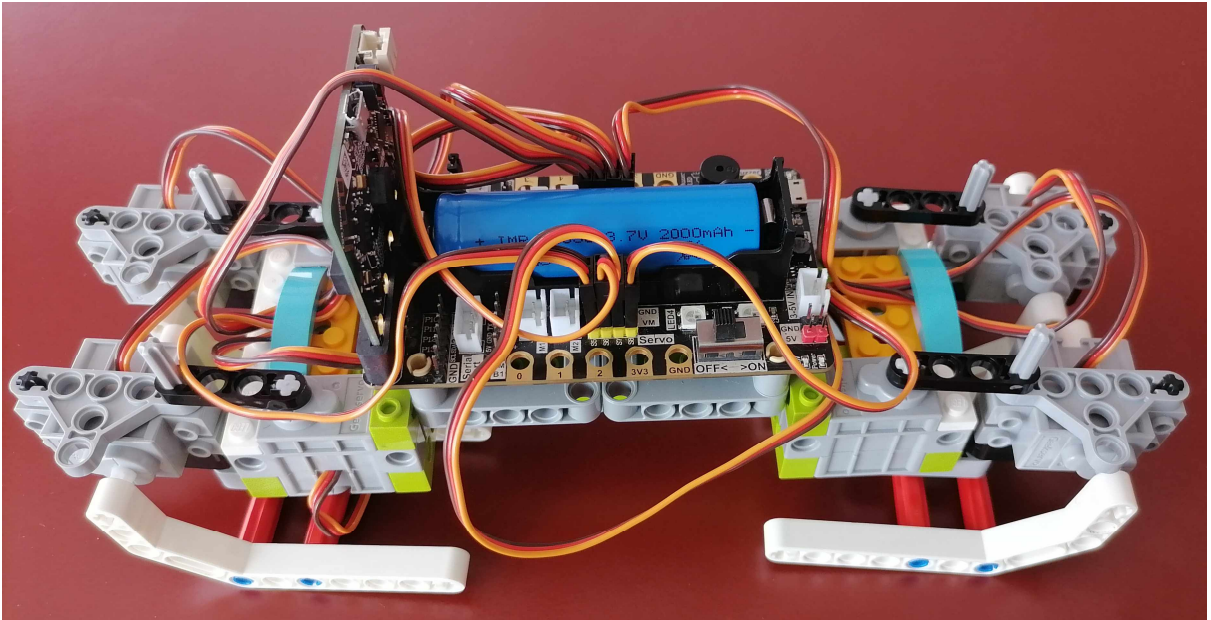


Рисунок 3.8 – Собранный модель квадропода

Для управления квадроподом использовался джойстик «Basic» с платой Micro:bit (см. рисунок 3.9).



Рисунок 3.9 – Джойстик «Basic»

На самом джойстике имеются кнопки, зуммер, вибрационный двигатель и другие модули.

4 ПРОГРАММИРОВАНИЕ КВАДРОПОДА

Для программирования квадропода использовалась среда программирования Makecode. Основные скрипты-команды в программе – это управление сервоприводами (см. рисунок 4.1).



Рисунок 4.1 – Код программы квадропода

После сборки робота осуществлялась настройка и синхронизация работы сервоприводов. Подбирались необходимые углы поворота сервоприводов (углы подбирались опытным путем). Целью настройки было достижение оптимального статического положения робота в разложенном и сложенном состояниях. Опытным путем для каждого серводвигателя были подобраны следующие значения:

Servo1 – значение +20 (передняя правая нога, поворот влево/вправо);

Servo2 – значение +55 (передняя правая нога, движение вверх/вниз);

Servo3 – значение -10 (задняя правая нога, движение вверх/вниз);

Servo4 – значение +20 (задняя правая нога, поворот влево/вправо);

Servo5 – значение +20 (передняя левая нога, поворот влево/вправо);

Servo6 – значение -10 (передняя левая нога, движение вверх/вниз);

Servo7 – значение +55 (задняя левая нога, движение вверх/вниз);

Servo8 – значение +20 (задняя левая нога, поворот влево/вправо);

Далее велась настройка движения робота в каждом из направлений.

Квадропод может двигаться в четырех направлениях (см. рисунок 4.2). Для движения робота в каждом направлении были написаны подпрограммы. Для более точного движения робота опытным путем подбирались углы сервоприводов. Для движения вперед программировалось плавное вытягивание и опускание ног для левой и правой стороны робота. Аналогично организовано движение назад.

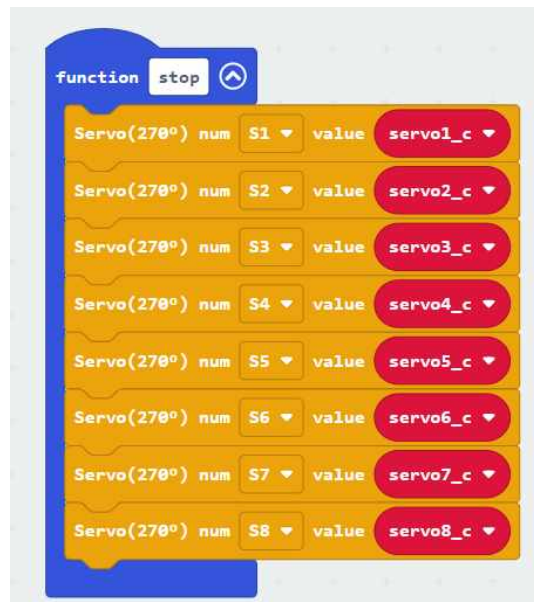


Рисунок 4.2 – Код программы квадропода (остановка)

Движение, остановка, поворот происходит через управление с джойстика.

Вызов каждой подпрограммы происходит из основной программы, после того как с джойстика поступила необходимая команда (см. рисунок 4.3).

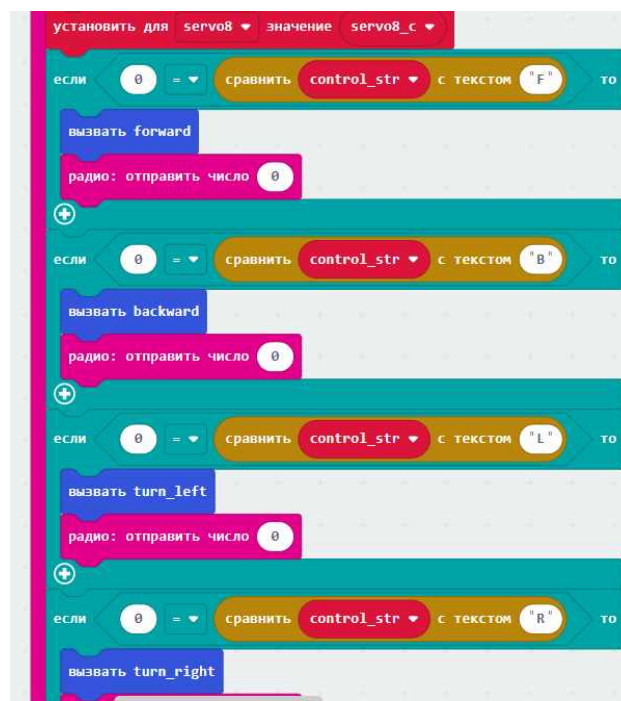


Рисунок 4.3 – Код программы квадропода (вызов подпрограмм)

С джойстика на плату управления квадроподом подаются команды:

- § «F» – движение вперед (направление джойстика вперед);
- § «B» – движение назад (направление джойстика назад);
- § «L» – движение влево (направление джойстика влево);
- § «R» – движение вправо (направление джойстика вправо);
- § «S» – разложить робота (зеленая кнопка на джойстике);
- § «X» – сложить робота (красная кнопка на джойстике);
- § «H» – подать звуковой сигнал приветствия, движение передней правой ногой (синяя кнопка на джойстике).

5 ПРОГРАММИРОВАНИЕ ДЖОЙСТИКА

Программирование джойстика также происходило в среда Makecode.

Обмен командами между квадроподом и джойстиком происходит по Bluetooth (см. рисунок 5.1).

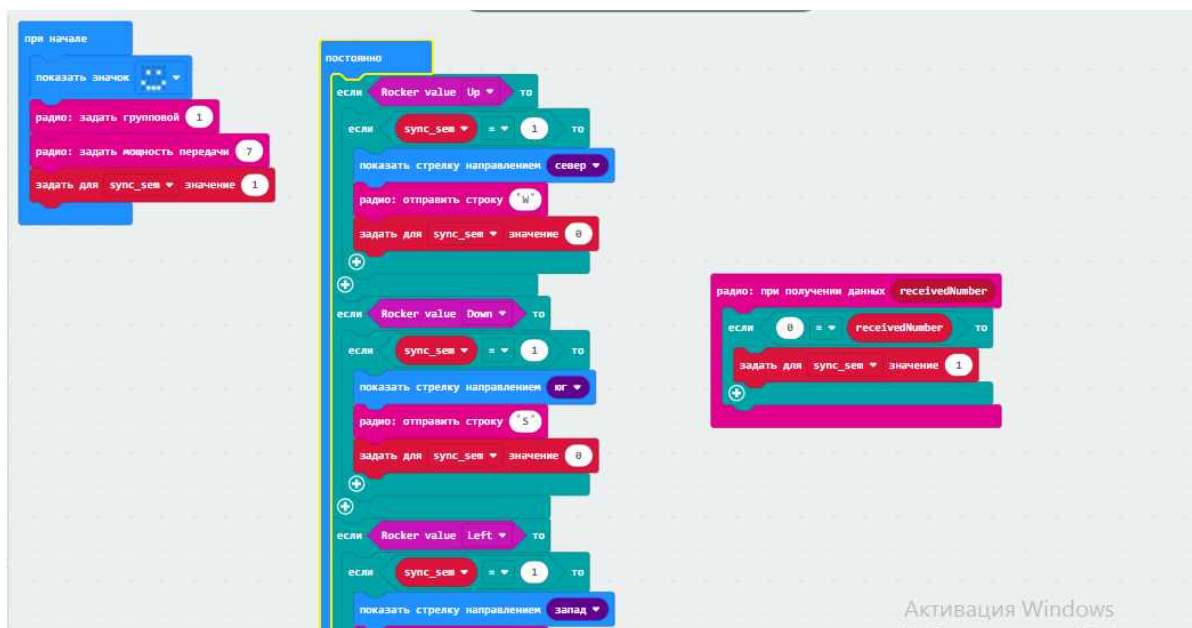


Рисунок 5.1 – Код программы джойстика

Выключенный робот находится в сложенном состоянии. После включения квадропод переходит в рабочее состояние: четыре сервопривода расходятся на углы по 45°, а еще четыре сервопривода поднимают робот. Сервоприводы работают как суставы ног.

ЗАКЛЮЧЕНИЕ

Таким образом, собранная и запрограммированная модель робота квадропода может осуществлять движение в четырех направлениях под управлением джойстика.

Данную разработку в перспективе можно использовать для организации движения в труднодоступных для человека местах на Земле (пустыня, горная поверхность) с определенной целью (доставка груза, исследование территории), по поверхности Луны. Модель можно использовать в соревнованиях по робототехнике в категории «Гонки шагающих роботов».

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. BBC micro:bit [Электронный ресурс]. – 2020. – Режим доступа: <https://microbit.org/>. – Дата доступа: 20.10.2020
2. Microbit: уроки [Электронный ресурс]. – 2020. – Режим доступа: <https://sites.google.com/view/p-s-p/>. – Дата доступа: 20.10.2020.
3. Давайте код [Электронный ресурс]. – 2020. – Режим доступа: <https://microbit.org/code/>. – Дата доступа: 20.10.2020.
4. Основные блоки программы [Электронный ресурс]. – 2020. – Режим доступа: <https://telegra.ph/nabor-instrumentov-Blocks-08-14>. – Дата доступа: 20.21.2020.
5. Робот на Micro:bit BBC [Электронный ресурс]. – 2020. – Режим доступа: <https://robotclass.ru/projects/microbit-bbc-robot/>. – Дата доступа: 20.10.2020.
6. Четвероногий робот на базе Arduino [Электронный ресурс]. – 2020. – Режим доступа: <http://mozgochiny.ru/electronics-2/chetveronogiy-robot-na-baze-arduino/>. – Дата доступа: 22.10.2020.