

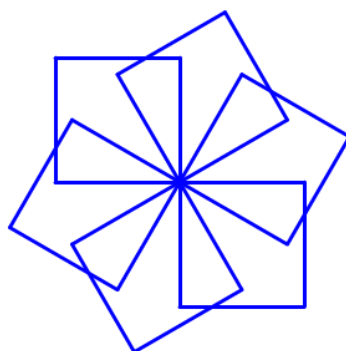
Методы распределённой разработки как учебный инструмент в робототехнике (план-конспект мастер-класса)

Автор: А.Т.Григорьев, методист, преподаватель робототехники

Введение

Любая робототехника, и прикладная, и соревновательная, и учебная, подразумевает решение сложных практических задач. При этом в образовательной робототехнике сложность задач напрямую сопряжена с возможностью удержания интереса обучающихся. Когда ребёнок видит, что по его команде робот выполняет действия, выглядящие разумными и рациональными, это неизменно вызывает восторг. Существуют ли методы и средства, которые позволяют детям решать сложные задачи легко, просто и с удовольствием?

Ответ кроется в самом понятии "сложность". "Сложный" это и "трудный", и "составной". Для того, чтобы решить сложную задачу, её следует разбить на простые части, этапы. Самый главный, основополагающий принцип структурного программирования, которому следует научить детей, это разбиение программы на куски. И прежде чем рассказывать о циклах и ветвлениях, надо освоить средства создания именованных блоков. Эти средства заложены в язык Scratch и во все программные продукты на его основе, а между тем о важности данного инструмента очень часто забывают. В частности, приходится наблюдать, как педагоги используют вложенные циклы, когда обучающиеся к этому ещё не готовы, а между тем с опорой на именованные блоки задача становится простой и понятной.



```
когда щелкнут по [флаг]
установить размер пера 3
опустить перо
повторить 6
  повторить 4
    идти 100 шагов
    повернуть на 90 градусов
  повернуть на 60 градусов
поднять перо
```

```
когда щелкнут по [флаг]
установить размер пера 3
опустить перо
повторить 6
  квадрат
  повернуть на 60 градусов
поднять перо
```

```
определить квадрат
повторить 4
  идти 100 шагов
  повернуть на 90 градусов
```

Практика показала, что одним из самых первых и любимых детьми заданий при знакомстве с робототехникой может быть общение азбукой Морзе. Ребёнок должен обучить робота произносить азбукой Морзе простые слова, например "да" или "нет", а также какое-нибудь придуманное имя. При этом нет нужды рассказывать о циклах или ветвлениях, и наглядно демонстрируется, насколько удобно опираться на блочную структуру программы.

```
установить ТЧК
play tone on note D5 beat 100
ждать 0.025 секунд

установить тиреее
play tone on note D5 beat 250
ждать 0.025 секунд

установить К
тиреее
тчк
тиреее
ПАУЗА

установить О
тиреее
тиреее
тиреее
ПАУЗА

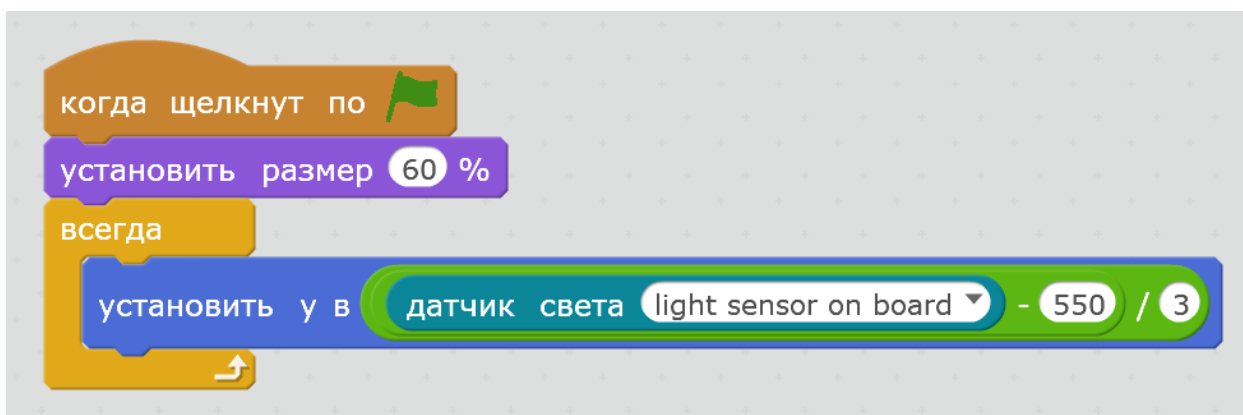
установить ПАУЗА
ждать 0.125 секунд
```

Структурное программирование, то есть разбиение программы на блоки, позволяет реализовывать сложные проекты поэтапно, а кроме того, позволяет организовать групповую работу, распределённую разработку проектного задания. Если разбить задание на части, каждую часть можно реализовывать в разное время и силами разных людей: обучающихся, преподавателей либо профессиональных программистов. Давайте посмотрим на практике, как это может использоваться. В среде mBlock3 существует несколько удобных способов: файл с шаблоном проекта, файл готового или частично готового спрайта как объект интерактивной среды, файл спрайта как библиотека готовых блоков для робота, загружаемое программное расширение как библиотека готовых блоков.

1. Файл с шаблоном проекта, файлы с готовыми спрайтами

Создаём игру "Тера с мячом" (управление спрайтом Панды с использованием датчика освещённости робота mBot).

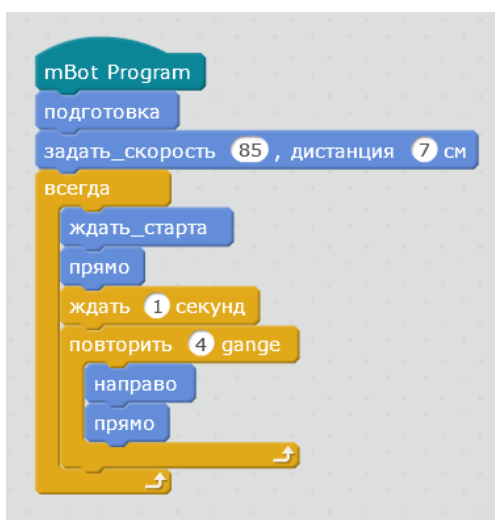
- Запустить программу mBlock, проверить настройки (демонстрация необходимых действий на интерактивной доске).
- Подключить робота к компьютеру посредством USB-кабеля, проверить возможность управления (демонстрация необходимых действий на интерактивной доске).
- Открыть шаблон "Панда и Тера играют с мячом.sb2" со всеми необходимыми блоками, расположенный на рабочем столе.
- Собрать скрипт для Панды по образцу на распечатке (внимание: вначале операция вычитания, потом операция деления!), проверить его работу.
- Добавить в проект два готовых объекта "Ball" и "Tera" при помощи кнопки "Добавить спрайт из файла" (файлы "Ball.sprite2" и "Tera.sprite2" расположены на рабочем столе).
- Активировать режим "Turbo" и проверить работу игры.



2. Файл спрайта как библиотека готовых блоков для робота

Решаем игровое задание: робот должен быть запрограммирован на самостоятельные действия так, чтобы после запуска (после включения питания и нажатия на кнопку, расположенную на корпусе) проехал по полю с квадратной разметкой, придерживаясь необходимого маршрута, а перед домом своего друга спел поздравительную песню. Используем режим "Arduino mode" (загрузка программы в память робота) и готовые блоки ("вперёд", "налево", "направо", "песня" и другие).

- Создать новый проект, удалить спрайт Панды.
- Открыть библиотеку готовых блоков, сохранённую как спрайт. Файл "mBot_on_marking.sprite2" расположен на рабочем столе.
- Переключиться в режим "Arduino mode", собрать скрипт тестовой программы для проверки скорости движения робота (см. рис. ниже).
- Проверить наличие соединения с роботом по последовательному порту (меню "Соединить", "Serial Port"), загрузить программу в робота нажатием на кнопку "Upload to Arduino".
- Отсоединить USB-кабель, поставить робота на поле с квадратной разметкой, запустить движение, нажатием на кнопку, расположенную на корпусе.
- При необходимости откорректировать скорость движения в блоке "задать скорость (85), дистанция (7) см", заново подключить робота кабелем, установить соединение по последовательному порту, загрузить программу в робота, проверить работу.
- Заново подключить робота, откорректировать программу для решения основной задачи, проверить её работу.



3. Ответы на вопросы, выполнение дополнительных усложнённых заданий по желанию (оставшееся время).

Вариант 1. Проект с несколькими библиотеками блоков, оформленными как спрайты. Задача: робот возле дома своего друга вместо исполнения песни должен передать сообщение азбукой Морзе.

- Выйти из режима "Arduino mode" (кнопка "Back" или меню "Редактировать", "Arduino mode").
- Добавить в проект спрайт с готовыми блоками символов азбуки Морзе (файл "Morse-blocks.sprite2" на рабочем столе).
- Выбрав спрайт "Morse-blocks" составить необходимое сообщение. Затем сдублировать это сообщение в спрайт, содержащий основной программный скрипт (методом Drag-and-Drop). Замечание: "бросив" блок или набор блоков на нужный спрайт необходимо этот спрайт выбрать и убедиться, что дублирование произошло. Внимание! Определения блоков дублировать не следует!

- Выбрав спрайт, содержащий основной программный скрипт, отредактировать его с использованием добавленных блоков. Переключиться в режим "Arduino mode", установить соединение, загрузить программу в робота, проверить работу.

Вариант 2. Проект с использованием расширения Advanced Arduino Extension. Интерактивное управление движением робота по квадратной разметке с помощью беспроводного канала связи.

- Составить программный скрипт с использованием распечатки, загрузить программу.
- Отсоединить USB-кабель, подключить робота с помощью комплекта беспроводной связи.
- Проверить работу.
- Самостоятельно модифицировать программу для запоминания роботом последовательности команд и самостоятельного возвращения в исходную точку (с использованием массивов).

```
mBot Program
подготовка
Serial begin 115200
declare GLOBAL byte var_D = 0
здать_скорость 85 , дистанция 7 см
всегда
  ждать_старта
  повторять до errorID > 0
    если Serial available > 0 то
      set var_D to Serial read
      если var_D = 'F' то
        прямо
      иначе
        если var_D = 'L' то
          налево
        иначе
          если var_D = 'R' то
            направо
          иначе
            если var_D = 'S' то
              песня
            иначе
              
```

```
повторять до errorID > 0
```

```
подготовка
```

```
ждать_старта
```

```
здать_скорость 85 , дистанция 7 см
```

```
прямо
```

```
налево
```

```
направо
```

```
песня
```