

Объектно – ориентированное программирование на Scratch.

Разработка игры «Лабиринт в космосе»

Смирнова Галина Леонидовна, педагог
дополнительного образования Кировского
областного государственного образовательного
автономного учреждения дополнительного
образования «Центр технического творчества»
структурного подразделения «Детский
технопарк «Кванториум»

Scratch — визуально-блочная, объектно – ориентированная среда программирования, созданная для детей и подростков. Эту среду можно использовать в программах дополнительного образования для развития навыков программирования для детей разного возраста. Среда содержит набор библиотек объектов, удобный интерфейс, систему блочного программирования. Все это помогает освоить основы алгоритмизации, научить детей создавать анимированные истории и разрабатывать интерактивные игры.

Визуальное программирование позволяет школьникам быстрее освоить способы управления объектами, а блочное программирование упрощает написание программы.

Основными объектами в **Scratch** – являются спрайты. Слово "спрайт" часто используется в программировании игр для обозначения фигурок героев и предметов. Каждый спрайт может иметь несколько костюмов. Благодаря смене костюма можно задавать анимацию спрайта. Для визуализации событий в интерфейсе программы собраны рисунки различных сцен, что позволяет сразу начать создавать анимированную историю. Кроме этого есть возможность самостоятельного создания своих героев и рисунков для сцен в простом графическом редакторе, который быстро осваивают школьники. Можно использовать готовые рисунки и загрузить их в систему программирования или редактировать существующие. Для озвучивания истории в программе **Scratch** есть библиотека звуков и простой редактор звуковых фрагментов. Все это позволяет создавать свои анимированные истории со звуковыми сопровождениями. Особенно это важно при разработке игр.

Система программирования **Scratch** используется для раннего обучения программированию на территории России и известна во всем мире.

С использованием **Scratch** проводятся олимпиады на всех уровнях от муниципального до международного, организуются различные тематические конкурсы. Знание этого языка программирования позволяет школьникам создавать свои проекты на различные темы.

Общее описание практики

Цель: знакомство с основами программирования на **Scratch** через создание интерактивной анимированной игры «Лабиринт в космосе».

Задачи:

- получить знания об основных элементах программы Scratch;
- научиться строить алгоритм управления объектами с помощью пазлов;
- освоить способы построения сложных конструкций соединения пазлов в один алгоритмический блок;
- научиться пользоваться библиотекой спрайтов и сцен;
- научиться пользоваться графическим редактором и получить навыки работы в режимах векторной и растровой графики;
- получить навыки работы с координатной плоскостью основного рабочего поля;
- получить навыки построения алгоритмов программы для перемещения объекта с помощью клавиш;
- научиться создавать диалог между объектами в игре;
- получить навыки работы с переменными для подсчета баллов и работы таймера;
- научиться создавать сообщения в игре для управления сюжетом;
- научиться использовать и применять алгоритмы условных операторов(ветвления) и операций повторений(циклов);
- научиться управлять клонами объекта;
- научиться сохранять программу и загружать её с компьютера.

Для создания интерактивной анимированной игры «Лабиринт в космосе» разработана подробная инструкция (см. Приложение).

Целевая аудитория

Данная методическая разработка предназначена для педагогов дополнительного образования, ведущих занятия технической направленности у школьников разных возрастных групп: 8-9 лет и 10-11 лет.

Для возрастной группы детей 8-9 лет желательно использовать два занятия по 2 академических часа, для детей 10-11 лет можно использовать одно занятие на 2 академических часа.

Все зависит от возможностей школьников и навыков работы за компьютером.

Основные этапы работы:

1. Введение в программирование – описание самого интерфейса.
2. Выбор сцены и рисование самого лабиринта.
3. Выбор объектов для игры из библиотеки Scratch.
4. Программирование объектов:
 - 4.1. Программирование движения объекта с помощью клавиш.
 - 4.2. Программирование простого диалога (начало истории).
 - 4.3. Программирование «клонов» (объектов с управляемым появлением в разных точках сцены)
 - 4.4. Создание переменных и их использование для системы расчетов
 - 4.5. Программирование счета переменной по условию сценария игры.
 - 4.6. Создание условий перемещения по лабиринту.
 - 4.7. Программирование начала истории по условию сценария игры
 - 4.8. Использование переменной таймера (счет времени).
 - 4.9. Программирование финала игры

Все этапы работы должны быть выполнены последовательно на основании приложенной инструкции к разработке этой игры «Лабиринт в космосе». На первом этапе желательно не отходить от основных рекомендаций и не менять сценарий по своему усмотрению. Но после завершения работы над программированием всех этапов возможны изменения, самостоятельная доработка, изменение сцены, объектов и самого сценария игры.

Методическое обеспечение представляет из себя 2-4 урока (для разных возрастных групп) при начальном изучении работы с программой Scratch.

Если школьники свободно владеют навыками работы с компьютером (умеют просматривать инструкцию на экране и одновременно программировать в программе Scratch), то для создания игры достаточно двух уроков по 40 минут.

Для детей младшего школьного возраста необходимо разбить всю работу на отдельные этапы и изучать в более медленном темпе с постоянным контролем и объяснением каждого раздела. После двух занятий необходимо сохранить проект и на следующем занятии продолжить работу, загружая проект с компьютера.

Если школьники плохо владеют навыками работы с файловой системой компьютера и допускают ошибки при сохранении и загрузке файлов, то для таких сложных ситуаций лучше всего иметь заготовки разработанного проекта на разных этапах работы, чтобы помочь восстановить проект, случайно потерянный школьником. Такие ситуации возникают, когда ребята путают две команды между собой («Сохранить на компьютер» свой проект» и «Загрузить с компьютера» ранее сохраненную работу). Кроме этого в школах или учебных центрах на разных компьютерах могут быть доступны для сохранения совершенно разные дисковые пространства. В этом самая большая сложность, которая может возникнуть для учеников с низким уровнем знаний работы за компьютерами. В остальном школьники быстро осваивают систему программирования Scratch и свободно создают свои проекты.

Сценарий занятия

№ п/п	Наименование этапа	Описание	Тайминг(мин)
	Теоретическая часть		
1	Введение	Идея разработки игры. Знакомство с интерфейсом программы	5 мин
	Практическая часть		
2	Разработка игры	Практическая работа в программе Scratch по	70 мин

		методическим рекомендациям	
3	Самостоятельная работа	Изменение сценария или доработка	10 мин
4	рефлексия	Выявление сложностей учащихся, появившихся на занятии	5 мин
Всего			90 мин

Ресурсы:

- ✓ Персональный компьютер или ноутбук
- ✓ Подключение к сети Интернет
- ✓ Интерактивная доска или экран с проектором

Программное обеспечение:

- ✓ Scratch Desktop – бесплатная версия, скачивается и устанавливается на компьютер или ноутбук (<https://scratch2.ru/>)
- ✓ On-line версия (только с доступом в интернет) - <https://scratch.mit.edu/projects/editor/?tutorial=getStarted>
- ✓ Любой браузер для скачивания рисунков с общедоступных файлов

Список литературы:

- ✓ Учебник по программированию на Scratch скачать бесплатно http://dvboyarkin.ru/wp-content/uploads/2015/05/Scratch_2014_1.pdf

- ✓ Scratch 3 для юных программистов – скачать бесплатно
<https://fb2-epub.com/23942-scratch-3-dlja-junyh-programmistov.html>
 - ✓ Основы программирования в Scratch 3.0. Пошаговое решение стандартных задач – скачать бесплатно
<https://avidreaders.ru/book/osnovy-programmirovaniya-v-scratch-3-poshagovoe.html>
 - ✓ Scratch-программирование – on-line учебное пособие с примерами
<https://bosova.ru/metodist/authors/informatika/3/scratch.php>
-

Приложение

Инструкция по изучению возможностей объектно – ориентированного языка программирования Scratch на примере создания игры «Лабиринт в космосе».

Цель: создание интерактивной анимированной игры.

Задачи:

- получить знания об основных элементах программы Scratch;
- научиться строить алгоритм управления объектами с помощью пазлов;
- освоить способы построения сложных конструкций соединения пазлов в один алгоритмический блок;
- научиться пользоваться библиотекой спрайтов и сцен;
- научиться пользоваться графическим редактором и получить навыки работы в режимах векторной и растровой графики;
- получить навыки работы с координатной плоскостью основного рабочего поля;
- получить навыки построения алгоритмов программы для перемещения объекта с помощью клавиш;
- научиться создавать диалог между объектами в игре;
- получить навыки работы с переменными для подсчета баллов и работы таймера;
- научиться создавать сообщения в игре для управления сюжетом;
- научиться использовать и применять алгоритмы условных операторов (ветвления) и операций повторений(циклов);
- научиться управлять клонами объекта;
- научиться сохранять программу и загружать её с компьютера.

Основной сюжет игры

На Луне работает космическая станция, которую обслуживают роботы. Случилась внезапная поломка энергетического блока и на его восстановление необходим редкий лунный минерал. Станция подает сигнал SOS и вызывает помощь.

Роботу предстоит пройти лабиринт и собрать редкий минерал для станции. Время работы робота ограничено таймером. Проходя лабиринт, робот может коснуться опасных участков и потерять свою энергию, ему

придется возвратиться на базу и подзарядиться. Число подзарядок ограничено возможностью конструкции робота, и после 5 перезарядок он требует полного ремонта.

Кристаллы появляются на своих участках и могут пропадать от действия внешней среды, поэтому их надо успеть собрать. Число кристаллов, необходимое для ремонта устанавливается в алгоритме программы (например, минимум 4, достаточное количество – 5 для полного ремонта энергетической установки).

Задача у робота сложная, надо успеть за определенный период собрать не менее 4 кристаллов, которые появляются в разных участках лабиринта, при прохождении лабиринта нельзя коснуться его стен.

Этапы работы по созданию игры

1. Выбрать сцену для лабиринта (Рис 1.)

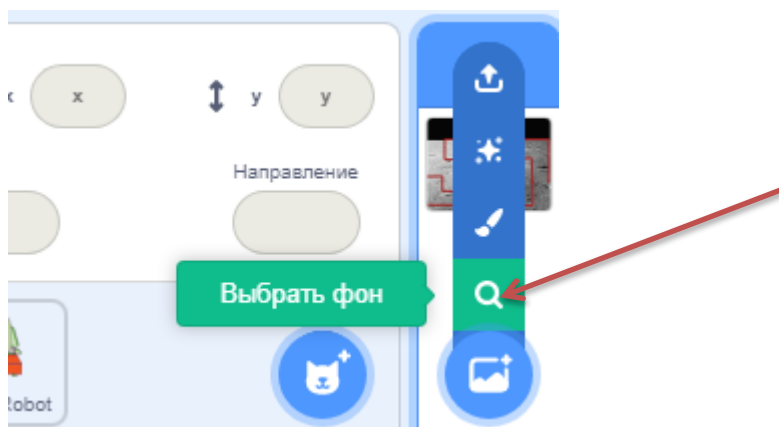


Рис 1. Выбор сцены.

2. Нарисовать сам лабиринт.

В режиме Сцена включить вкладку –Фон (Рис 2.).



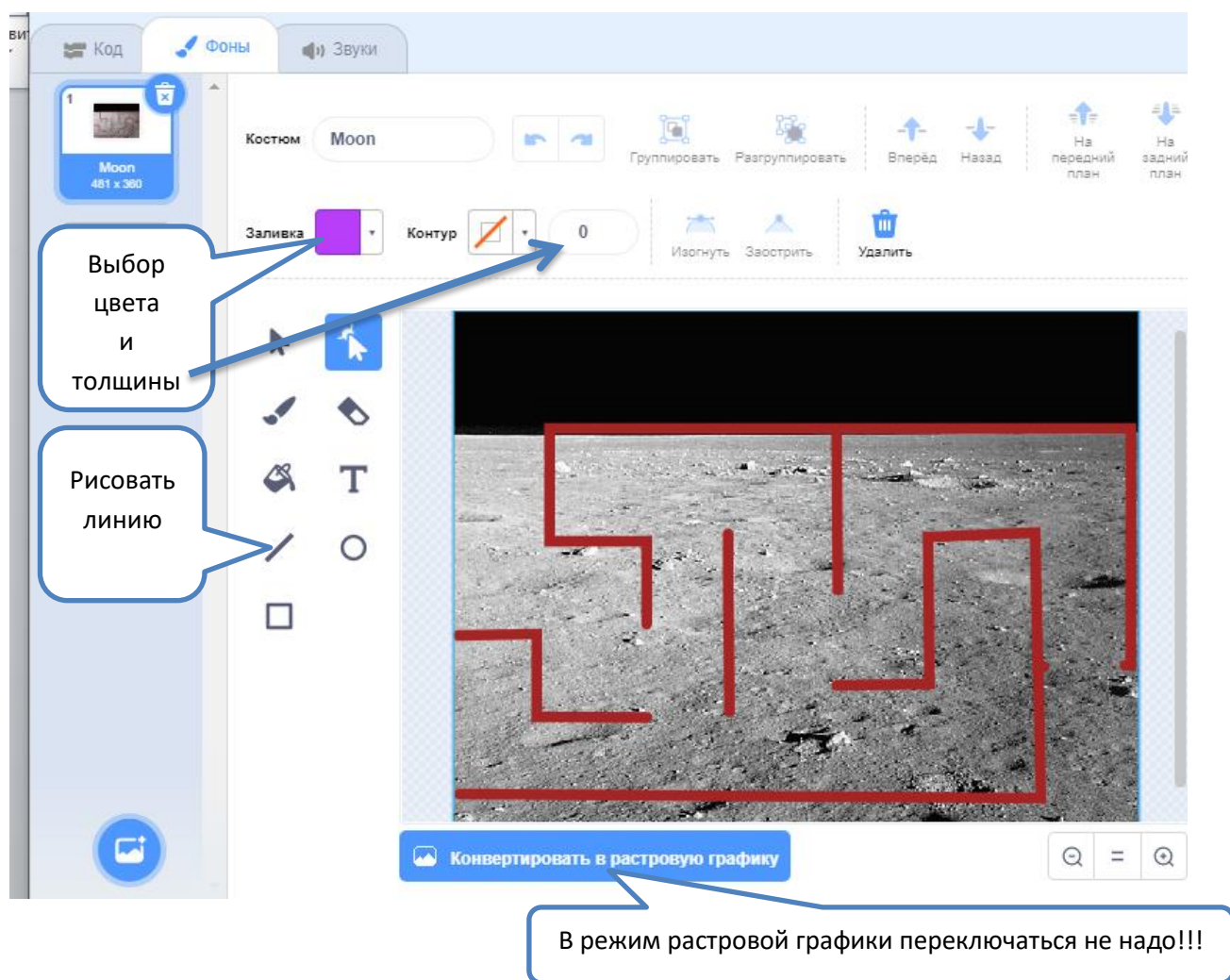


Рис 2. Рисование и редактирование фона для игры.

Рисовать лабиринт необходимо в режиме векторной графики, так как удаление и продолжение линий в этом режиме наиболее эффективно. При работе в растровом режиме удаление линий повреждает фон самой сцены. Необходимо научиться переключать эти режимы.

Для рисования основного лабиринта необходимо сначала выбрать цвет, желательно насыщенный и однородный, толщину линии, режим рисования Линии (кисточкой неудобно рисовать стены лабиринта)

При рисовании прямых линий необходимо использовать клавишу Shift. Постараться при создании лабиринта предусмотреть расстояние между стенками, чтобы можно было пройти (все конечно можно потом исправить, но лучше продумать сразу).

Основные элементы рисования, как в обычных графических редакторах (Рис 3.).

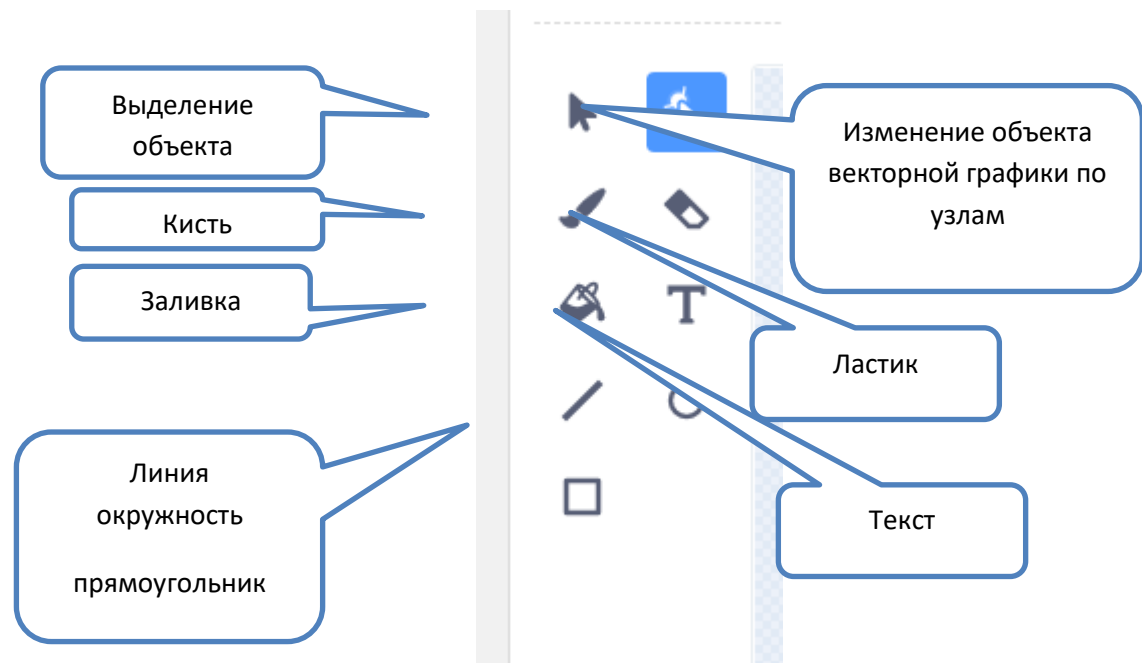


Рис 3. Панель инструментов графического редактора.

Примерный вид лабиринта (Рис 4.).

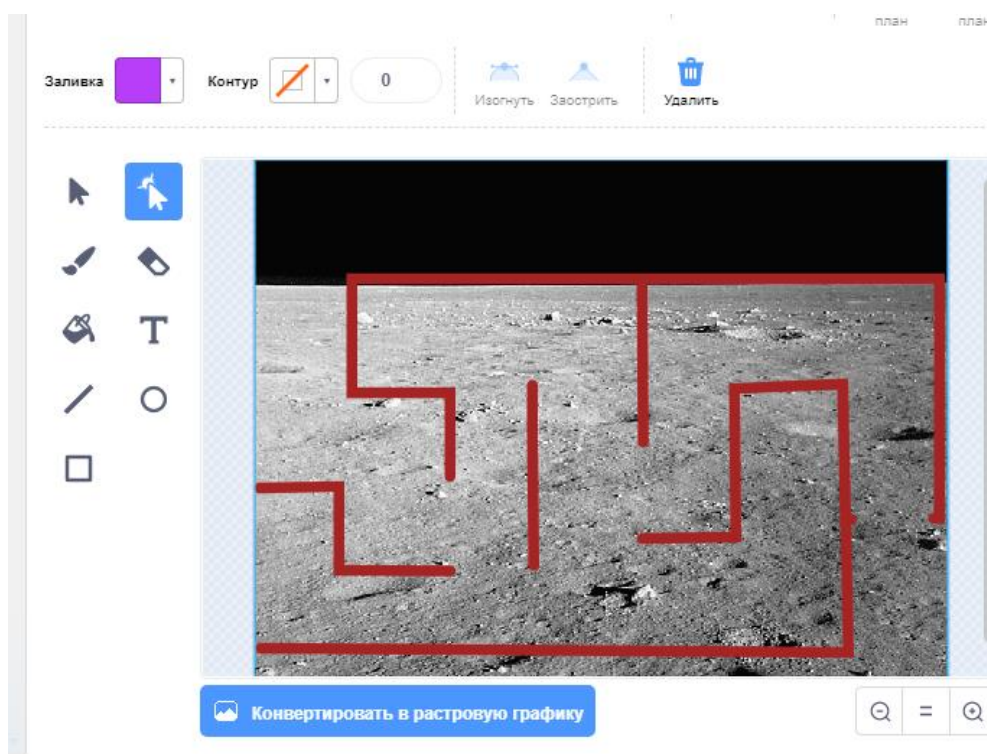


Рис 4. Готовый Фон для игры «Лабиринт»

3. Выбор объектов для игры

Найти кнопку выбора спрайтов из библиотеки и выбрать следующие объекты(Рис 5.):

- Робот
- Домик
- Кристалл

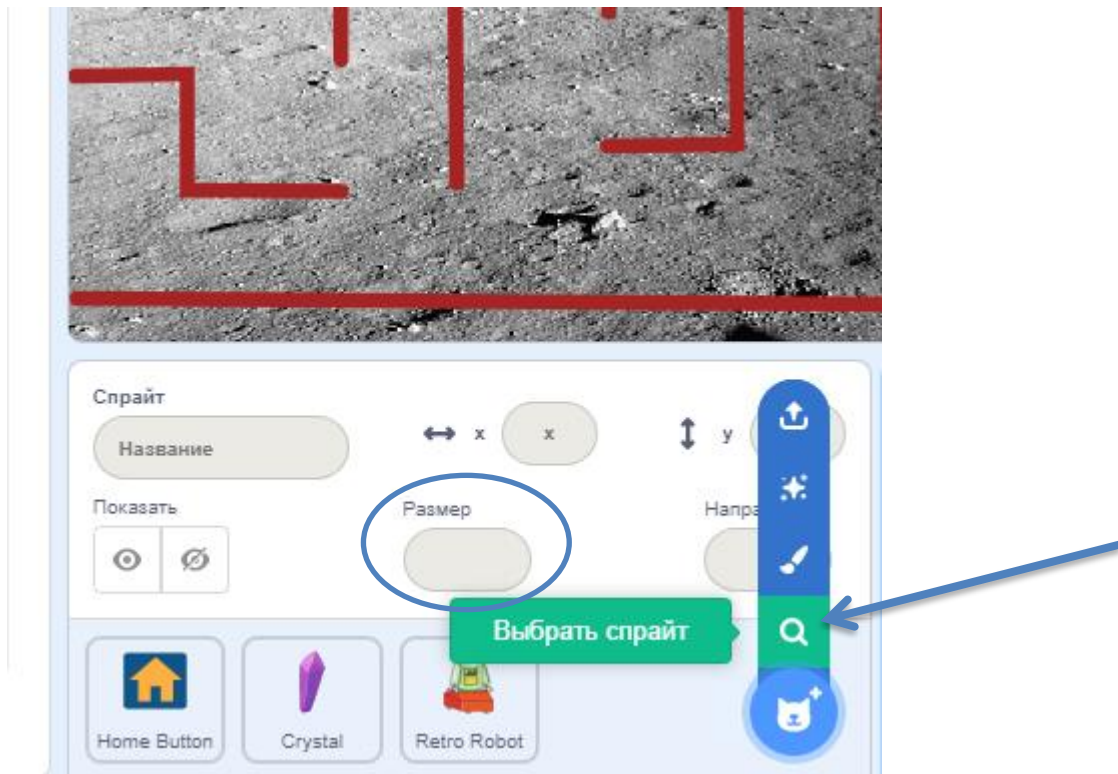


Рис 5. Кнопка выбора объектов из библиотеки спрайтов.

Всем объектам установить необходимый размер. Робот должен проходить мимо стен, домик должен вписываться в экран и лабиринт. Кристалл тоже нужно уменьшить до 50%, на (Рис 5.) указано поле изменения размера.

Примерный вид экрана с объектами(Рис 6.).

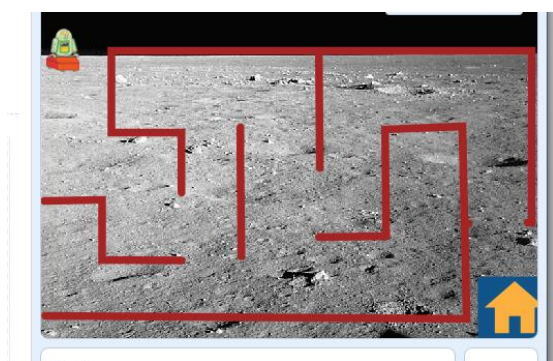


Рис 6. Фон игрового поля со спрайтами для игры «Лабиринт».

По желанию в режиме [Костюм] можно изменить цвет и форму любого спрайта, можно пользоваться любым удобным режимом работы (векторной или растровой графикой). Спрайты можно загрузить с компьютера, для этого надо найти режим - Загрузить спрайт. Чаще всего картинки загружаются с белым фоном, который нужно аккуратно удалить ластиком. Кропотливая это работа. Спрайт можно рисовать в режиме – Спрайт – Кисть (Нарисовать)

4. Программирование

4.1. Программирование движения робота с помощью клавиш.

Выберите спрайт – Робота и включите режим – Код.

Собрать программу из блоков. Блоки удобно запомнить по цветам. Они собраны в группы(Рис 7.).

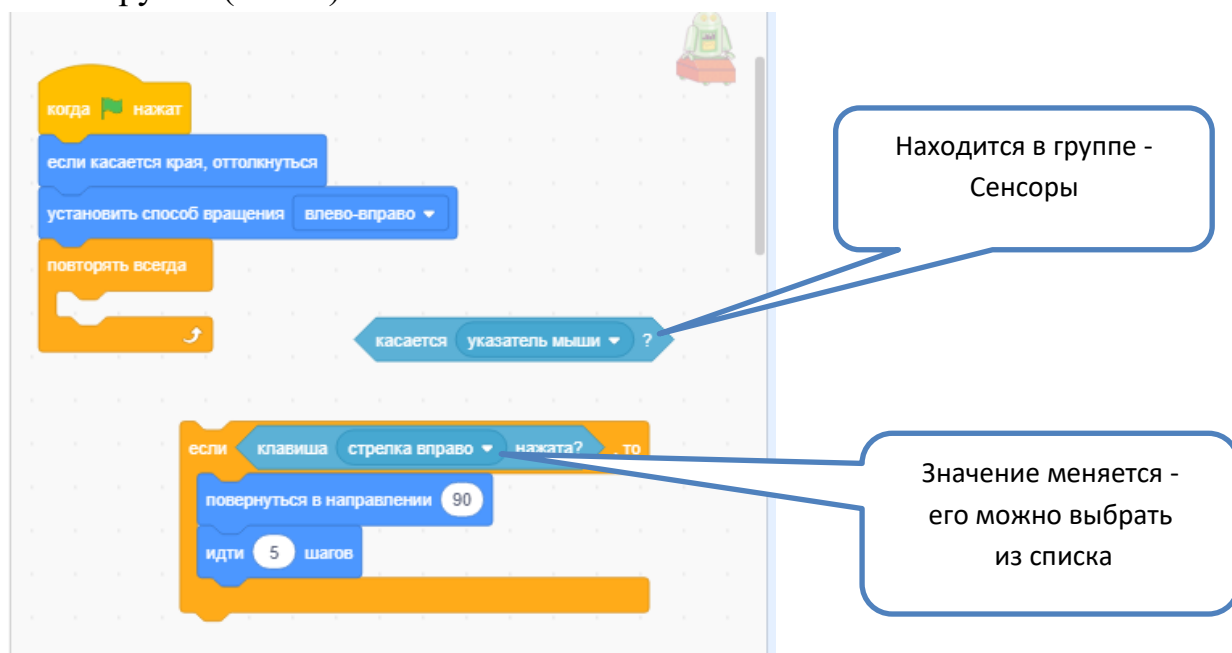


Рис 7. Схема сборки блоков.

Направление выбирается мышкой по кругу(Рис 8.)

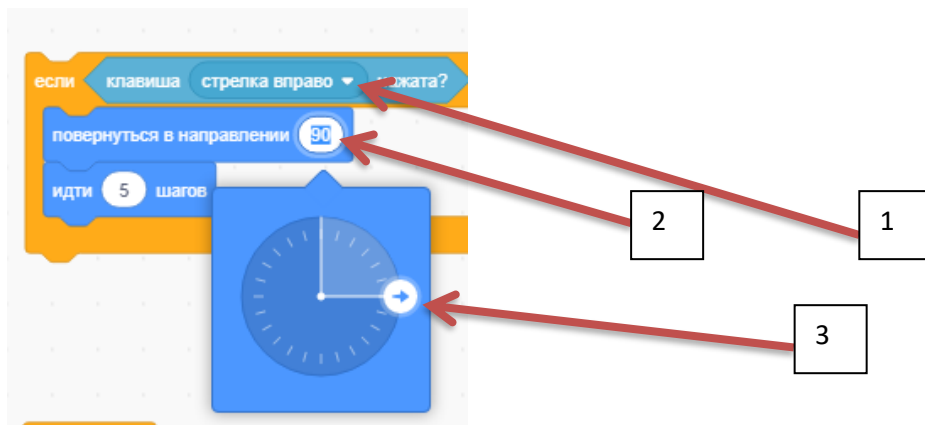


Рис 8. Выбор направления движения объекта с помощью мыши

Этот блок удобно дублировать правой кнопкой мыши на пазле(Рис 9.)

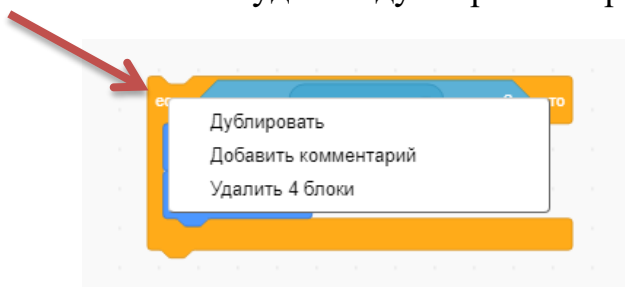


Рис9. Дублирование пазла с помощью правой кнопки мыши.

Далее нужно соединить и настроить все блоки (Рис 10.)

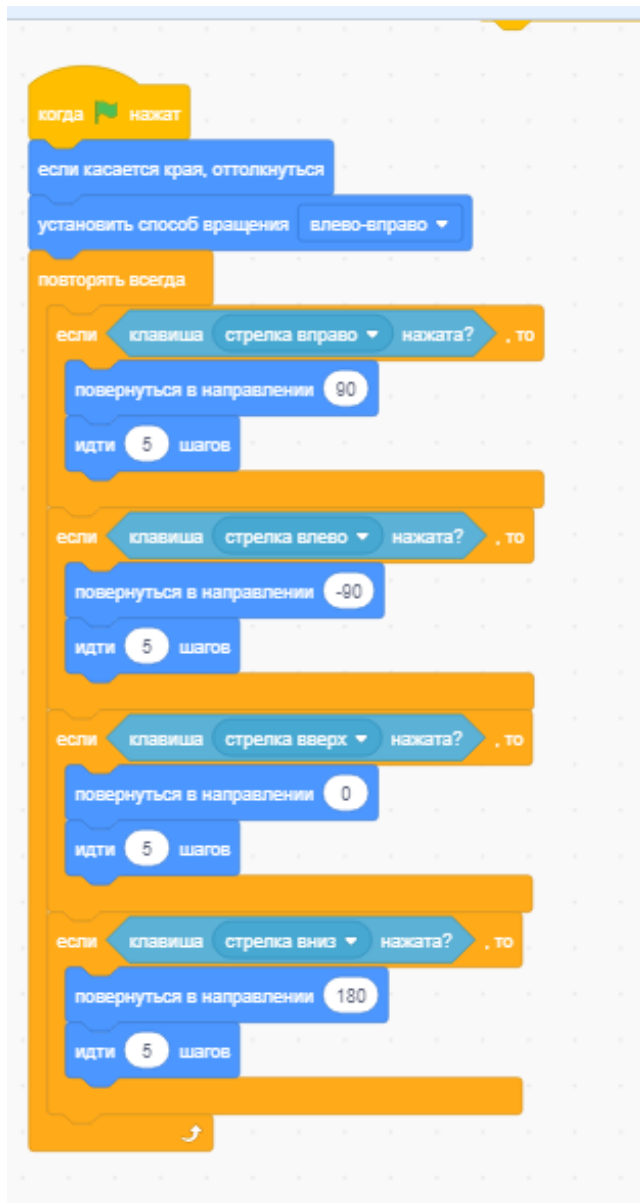


Рис 10. Пример собранной программы из пазлов для организации движения спрайта с помощью клавиш клавиатуры.

Проверка программы.

Включите режим Игры – Зеленым флажком(Рис 11.) и проверьте движение робота клавишами с клавиатуры. Красный кружок – остановка программы.



Рис 11. Основные кнопки запуска программы и её остановки, они находятся в верхней части окна интерфейса Scratch.

4.2. Программирование начала игры – переход к спрайту – базы (Домик)

Продолжение программирования объектов.

Программируем начала диалога. Вызов помощи от космической станции...

Включаем на спрайте – Домик – режим – Код.

Собираем следующий код программы (Рис 12.)



Рис 12. Код программы для организации диалога.

4.3. Программируем появление кристаллов.

Находим спрайт – Кристалл и собираем код (Рис 13. - 17.)

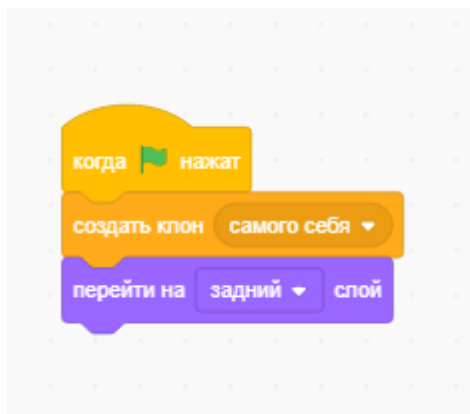


Рис 13. Создание клона.

Далее нужно установить кристалл на удобное место внутри лабиринта и собрать следующий блок кода

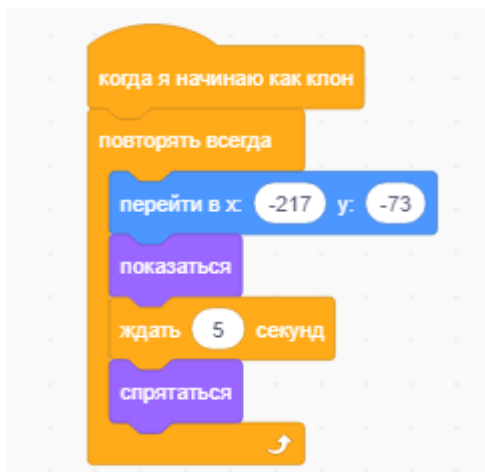
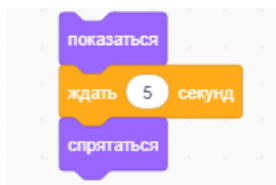


Рис 14. Управление клоном.

Группу кодов можно отсоединить и дублировать, например, часто будут повторяться три кода.



их можно отсоединить и продублировать несколько раз(4).

Собираем дальше код появления кристаллов в разных уголках лабиринта.

Для этого находим новое место для кристалла внутри лабиринта и находим в

группе – Движение Пазл -



, координаты

местоположения объекта программа сама запоминает.

Добавляем к нему продублированную группу пазлов и подсоединяем в блок – Повторять всегда

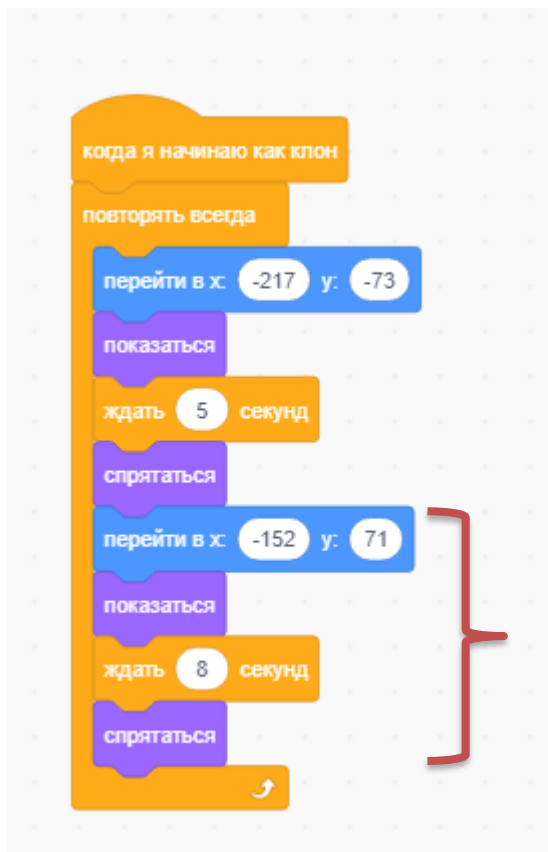


Рис 15. Управление клоном кристалла

Так повторяем еще 3 раза, каждый раз находим новое положение кристалла, выбираем пазл – (Перейти в x [] y[]) и добавляем продублированную группу пазлов. Время ожидания ставим разное, иногда оно потом подбирается индивидуально для каждого кристалла через многократную проверку в режиме игры.

(фрагмент объединения групп пазлов ...)

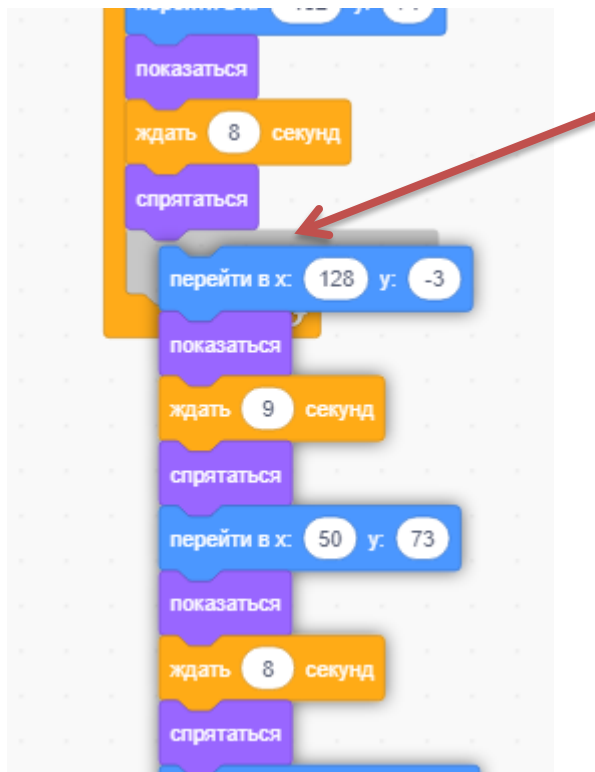


Рис 16. Объединение блоков клона кристалла

Это не последний блок....

Смотрите далее....

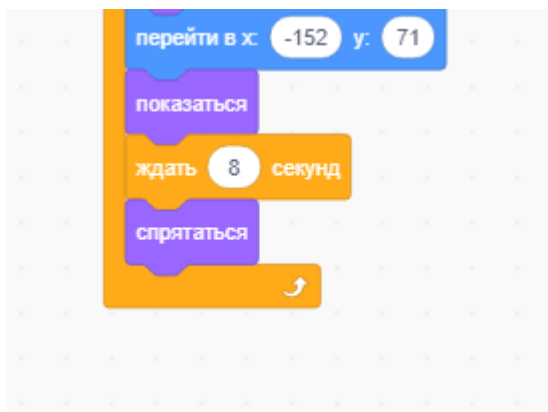


Рис 17. Заключительный блок собранного пазла для управления клона кристалла.

4.4. Програмируем управление переменными

Создаем переменные в группе Переменные - командой – Создать переменную (Рис 18.)

Нужно создать три переменных(кнопка создания переменной)

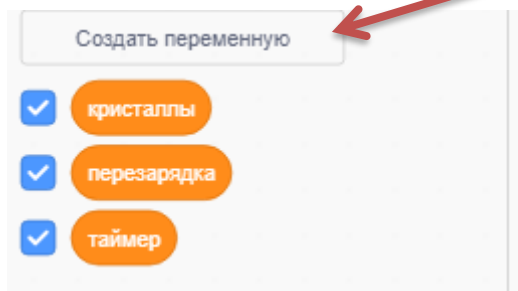


Рис 18. Пример создания переменных

(не перепутайте с командой – Создание списка, это будет неверная операция)

4.5. Программируем подсчет кристаллов

Обратите внимание – выбираем спрайт – Кристалл- это, тот объект, который собирает Робот(Рис 19.)

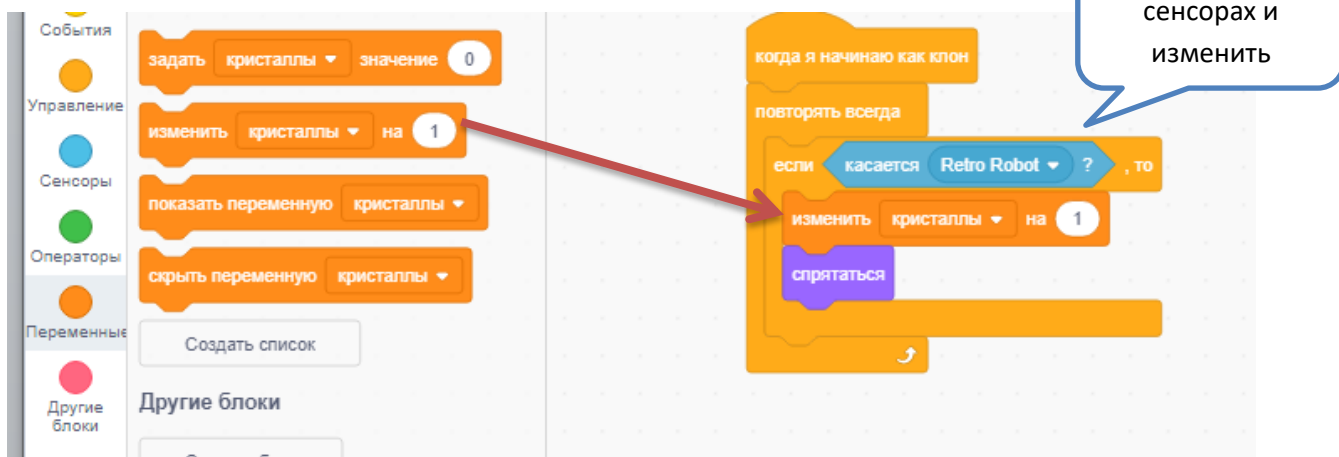


Рис 19. Пример сборки блока с управлением переменной.

Эту группу пазлов создаем дополнительно к тем, которые уже построены (Рис20.).

Чтобы собрать пазл – Касается – цвета.....

Нужно перенести этот пазл из группы Сенсоры, подсоединить к пазлу – Если.....

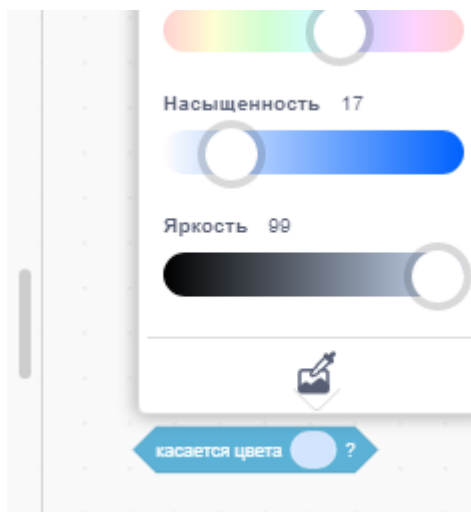


Рис 22. Фрагмент настройки цвета для выбора кнопки – «Пипетка».

Далее нажать левой кнопкой на поле Цвета, затем выбрать картинку – пипетка(Рис 22.)-(как в графических редакторах), далее перенести курсор на границу линии лабиринта и зафиксировать самый центр линии – цвет перенесется в пазл(Рис 23.)

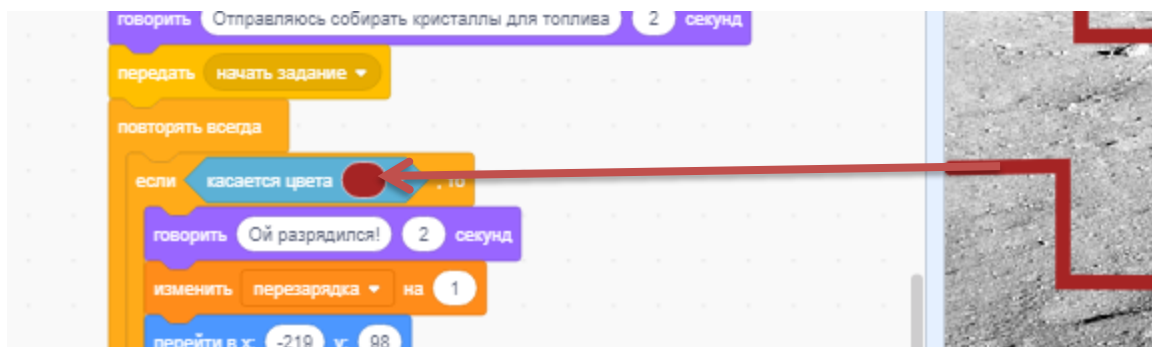


Рис 23. Фрагмент формирования цвета с помощью «Пипетки»

Продолжение группы пазлов в блоке – Повторять всегда (Рис 24.)



Рис 24. Фрагмент блоков для программирования Робота

Для формирования сложных условий используется группа – Операторы

Можно собирать несколько блоков в одно сложное выражение (Рис 25.)

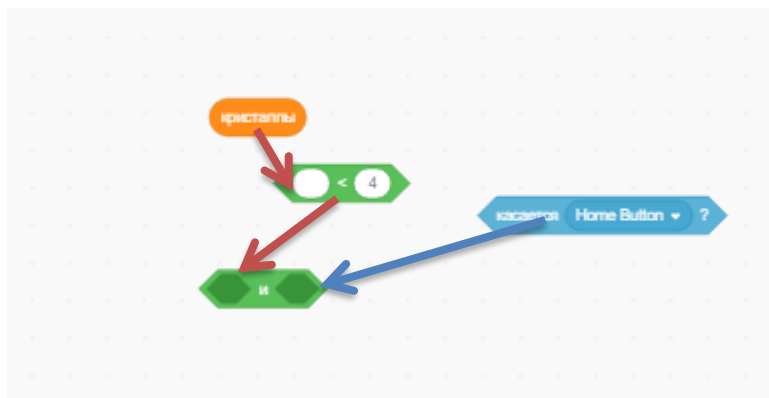


Рис 25. Схема сборки пазлов сложного условия.

4.7. Перепрограммируем начало игры так, чтобы Робот начинал движение только после диалога с базой

Переместим курсор на самую первую группу пазлов, отвечающих за движение Робота (Рис 26.)

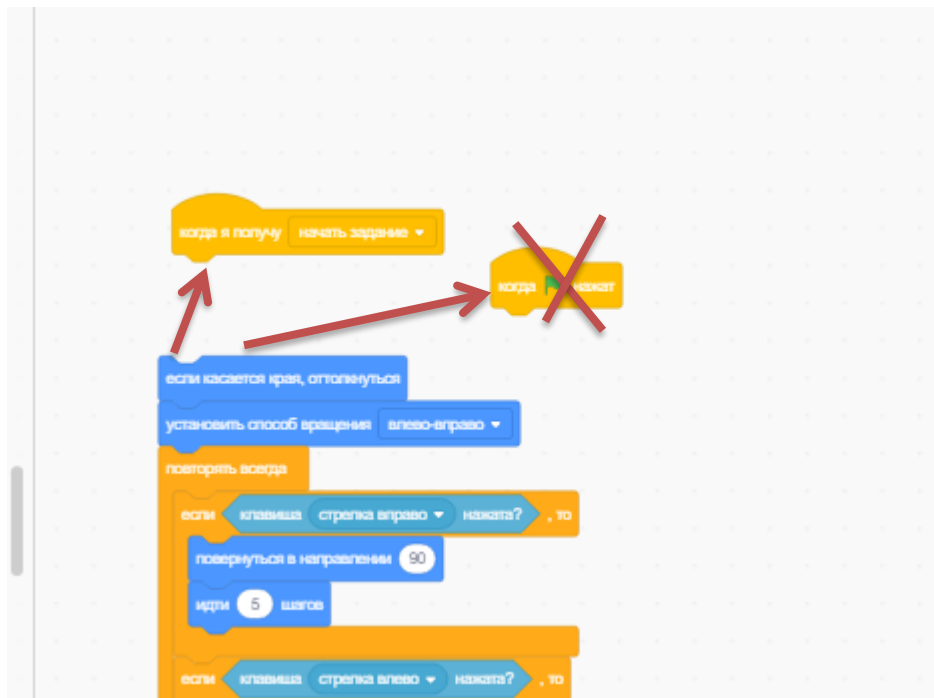


Рис 26. Схема сборки пазлов для изменения управления Роботом

Отсоедините пазлы сначала вниз, потом найдите нужный пазл и прикрепите нижнюю группу пазлов к новому управляющему пазлу – Когда получу....(Рис 26.).

4.8. Программируем блок зависимости от таймера. По условию игры – время у Робота ограничено таймером. Добавляем ему код (Рис 27.).

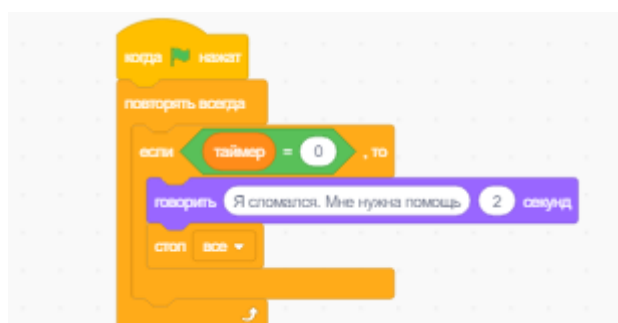


Рис 27. Фрагмент блока управления переменной «Таймер».

Начало работы счетчика программируем на самом фоне игры

Для этого, будьте внимательны и включите активно фон игры -
Сцену
И создайте группу пазлов на коде сцены (Рис 28.)



Рис 28. Фрагмент программы для сцены с работой переменной
«Таймер»

4.9 Програмируем финал игры.

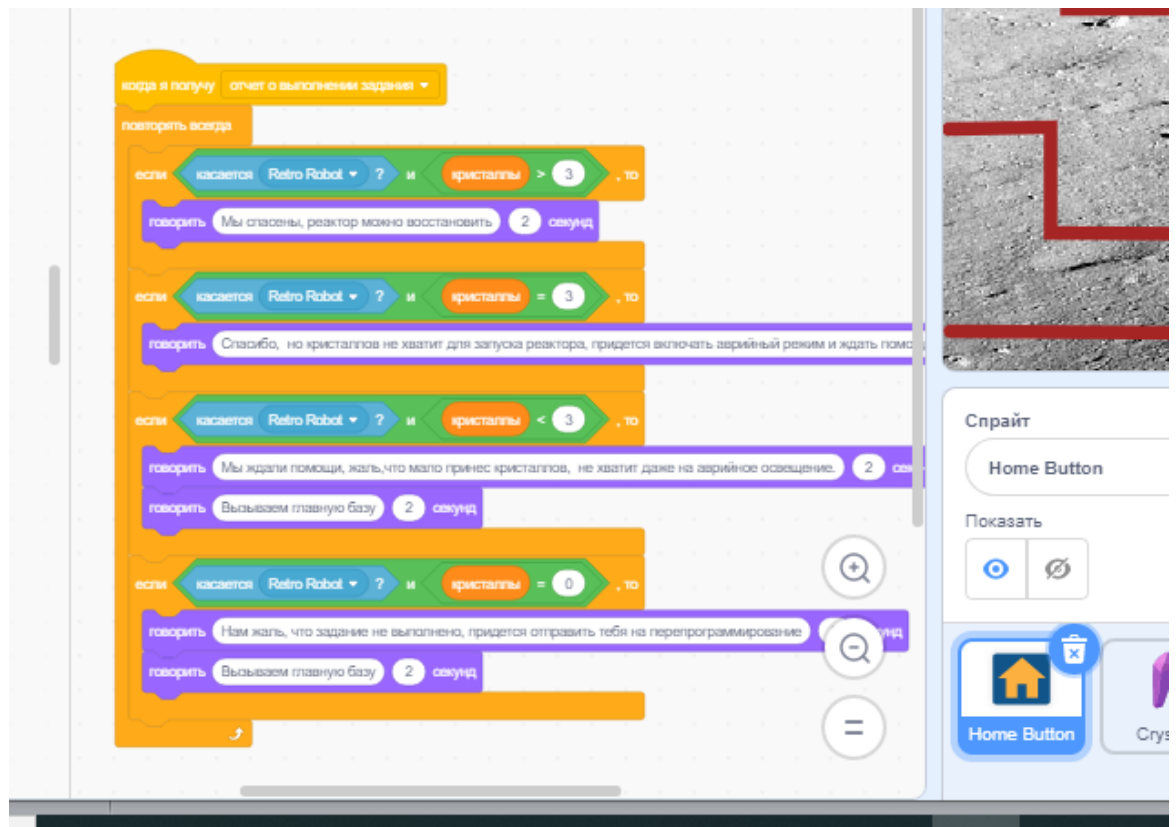
Если Робот дошел до базы, то должна быть проверка:
Сколько он собрал кристаллов?

И успел ли он коснуться базы до установленного таймером времени?

Переходим на пазл – Домика

И програмируем несколько условий

Тексты можно придумать произвольные, главное, чтобы они
соответствовали выполнению задания, например (Рис 29.)



Последнее условие обязательно

Рис 29. Фрагмент блоков для управления итогом игры.

Можно тестировать игру.

Проверять выполнение всех разработанных условий

Самостоятельная работа

Желательно придумать переход на новую сцену в разных ситуациях.

Как дополнение к разработке самостоятельного решения предлагается добавить следующие шаги игры:

- если Робот не успел по времени и вызвал основную базу – можно перейти на фон с космическим кораблем и придумать нового героя, который будет давать команду двигаться к Луне для спасения космической лунной базы;
- если Робот успел, но принес недостаточно кристаллов, то тоже - перейти на новый фон и принять решение о помощи лунной станции и дать команду двигаться к Луне;

- если все отлично, тоже перейти на фон космического корабля и поздравить Робота с успешной операцией.