

Управление народного образования
Администрации городского округа Дубна Московской области

Муниципальное бюджетное общеобразовательное учреждение
«Средняя общеобразовательная школа №5
г.Дубны Московской области»

СОГЛАСОВАНО
на Педагогическом совете
от «04» июля 2019г.



УТВЕРЖДАЮ

директор

Приказ № 40 от 17.07

2019г.

Дополнительная общеобразовательная программа
«Схемотехника с контроллером Arduino»

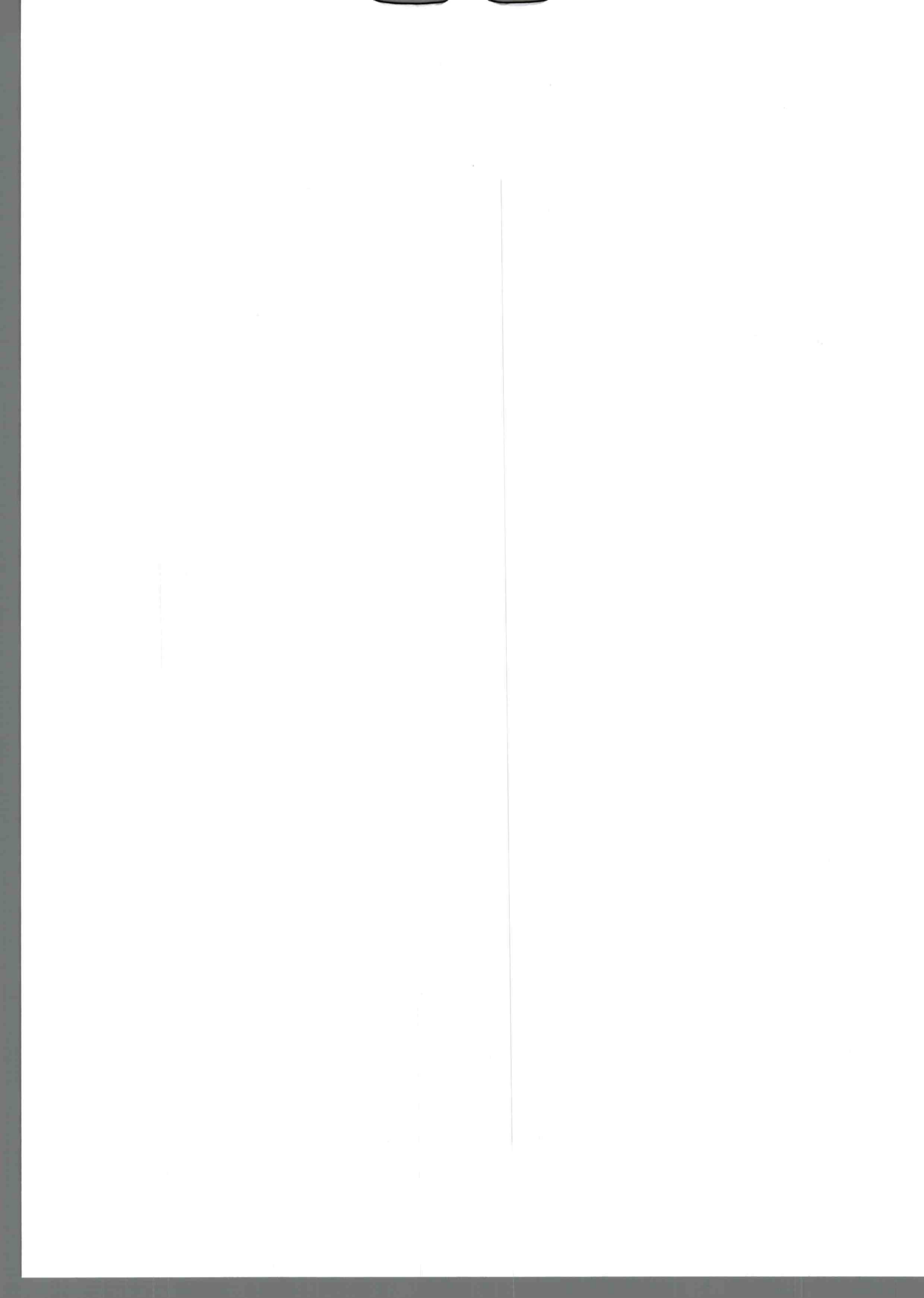
технической направленности
(базового уровня)

возраст детей: 9-16 лет
(разновозрастная, учитывающая возрастную дифференциацию)

срок реализации – 1 год

Руководитель
Голяков Николай Александрович

Дубна, 2019г.



Пояснительная записка

Начальная школа играет исключительно важную роль в общей системе образования. Это то звено, которое должно обеспечить целостное развитие личности ребенка, его социализацию, становление элементарной культуры деятельности и поведения, формирование интеллекта и общей культуры. Сегодня важно в образовании делать упор на формирование общей культуры обучающихся, на их духовно-нравственное, социальное, личностное и интеллектуальное развитие, на создание основы для самостоятельной реализации учебной деятельности, обеспечивающей социальную успешность, развитие творческих способностей, саморазвитие и самосовершенствование, сохранение и укрепление здоровья младших школьников.

Для развития личности ребенка требуется подготовить современную информационно-образовательную среду. Сегодня изменение сути профессионального образования, переход к применению и обслуживанию высокотехнологического оборудования, роботизированного производства, активного применения компьютерного моделирования и симулирования требует новых подходов к информационной образовательной среде. Одним звеном современной образовательной среды является наличие робототехники в образовании.

Робототехника в образовании — это междисциплинарные занятия, интегрирующие в себе науку, технологию, инженерное дело, математику (Science Technology Engineering Mathematics = STEM), основанные на активном обучении учащихся. Во многих ведущих странах есть национальные программы по развитию STEM образования. Робототехника представляет учащимся технологии 21 века, способствует развитию их коммуникативных способностей, развивает навыки взаимодействия, самостоятельности при принятии решений, раскрывает их творческий потенциал. Дети и подростки лучше понимают, когда они что-либо самостоятельно создают или изобретают.

Новые ФГОС требуют освоения основ конструкторской и проектно-исследовательской деятельности, и программы по робототехнике полностью удовлетворяют эти требования.

Область робототехники очень обширна и немаловажную роль имеет то, каким образом ее преподнести младшим школьникам, не имеющим еще познаний в физике, математике, программировании, но уже имеющим опыт общения дома с «умными» игрушками, на улице с информационными стендами (светофором), гаджетами. Подрастающему поколению предстоит общаться с роботизированным миром еще более тесно чем сегодня.

Срок реализации программы - 1 год. Группы комплектуются из обучающихся 9 – 16 лет. Количество обучающихся в группе – 20 человек. Продолжительность занятия – 1 час, перерыв 15 мин. Занятия проводятся в группах 1 раз в неделю.

Основными **формами** работы с обучающимися являются групповые занятия и индивидуальная работа.

Используются **методы** фронтальной работы: объяснение, показ, соревнования, а также методы индивидуальной работы: инструктаж, разработка и реализация индивидуальных творческих проектов, запуски моделей.

В программе затронуто большое количество компетенций, которые будут приобретены обучающимися:

- зарегистрироваться на сайтах и вести свои работы на них;
- работать руками не только при сборе макета, но и при создании проектов работ;
- работать со справочной информацией, разработками других пользователей, информационными ресурсами;
- освоить два урока по технике безопасности разбитых на технику безопасности в компьютерном классе и с низковольтным напряжением;
- освоить блочное программирование;
- научиться читать алгоритмы;
- познакомиться с базовыми элементами электротехники;
- освоить язык программирования arduino;

- научиться разбираться в принципиальных схемах;
- разработать свою первую макетную плату;
- научиться устанавливать ПО, настраивать оборудование (установка драйверов);
- расширить свои знания в микроэлектронике;
- сравнить все возможные разработки в различных системах.

Задания для школьников младшего и среднего возраста имеют разный уровень сложности. Например, для младших школьников используется блочное программирование, а для среднего звена программирование на Arduino (язык программирования с синтаксисом Си).

Цель программы - формирование знаний о назначении базовых элементов электроники, развитие навыков проектирования и программирования.

Программа идеально подходит для подготовки младших школьников к курсам робототехники.

Задачи:

Личностные:

- воспитывать интерес к техническому творчеству путем приобщения к робототехнике;
- формировать культуру труда и совершенствование трудовых навыков;
- воспитывать аккуратность, бережное отношение к материалам;
- мотивировать на здоровый и безопасный образ жизни
- развитие способностей проектирования и макетирования как для написания визуальной среды для работы с макетом, так и разработка макета для визуальной среды Scratch;
- привитие навыков проектной деятельности и группового творчества;
- развитие личности ребенка через участие в соревнованиях, конференциях и тематических семинарах.

Метапредметные:

- развивать внимание, память, логическое и пространственное воображение;
- развивать творчество, фантазию, воображение, интерес к процессу работы и получаемому результату;
- изучение основ программирования на основе визуальной среды Scratch;
- изучение базовых элементов электроники в игровой форме

Образовательные:

- обучать первоначальным правилам безопасности при работе за компьютером и с электричеством;
- пробуждать любознательность и интерес к устройству простейших технических устройств, развитие стремления разобраться в их конструкции и применения их в повседневной жизни;
- знакомить детей с основными понятиями построения алгоритмов.

Ожидаемые результаты реализации программы.

Знать:

- базовые электронные компоненты, используемые в устройствах;
- основные принципиальные схемы для подключения базовых электронных компонентов;
- методы и алгоритмы программирования на примере среды Scratch;
- правила техники безопасности при работе с электронными компонентами.

Уметь:

- самостоятельно воспроизводить и собирать простейшие схемы рассматриваемых электронных компонентов;
- самостоятельно работать с безопасной монтажной платой и собирать небольшие узлы для взаимодействия со средой программирования;

- самостоятельно писать программный код в среде программирования для взаимодействия с электронными узлами.
- навыками сборки электронных устройств на безопасной монтажной плате;
- навыками работы со справочной литературой в рамках курса.

Учебно-тематическое планирование

№ п/п	Содержание	Кол-во часов	В том числе на:		Выполнение практической части программы	Контроль (защита проектов)
			уроки	резерв		
1	Введение	2	2			
2	Знакомство со средой программирования и регистрация на ресурсах	10	4		6	
3	Работа с проектами	22		1	19	2
	Итого	34	6	1	25	2

Содержание программы

1. Введение

Введение в программу. Содержание и задачи программы. Комплектование группы. Правила поведения. Изучение правил техники безопасности при работе в компьютерном классе (лаборатории). Рассмотрение роли микроэлектроники на современном этапе развития общества, основные понятия микроэлектроники.

2. Знакомство со средой программирования и регистрация на ресурсах

Теория: Подготовка к работе, регистрация на сайтах, ознакомление с рабочими системами Tinkercad Circuits и S4A (mBlock). Демонстрация примитивных работ в Tinkercad Circuits, mBlock и S4A Знакомство с работой среды S4A (mBlock). Разработка виртуального светофора по алгоритму. Чтение линейного алгоритма. Замена повторяющихся блоков на цикл n раз.

Изучение правил техники безопасности при работе с электронными компонентами. Составные элементы платы Arduino Светодиод, сопротивление, макетная плата, как электронные комплектующие. Изучение требуемых блоков для подачи напряжения на диоды. Разбор алгоритма плавного погасания

Разбор соответствия блоков среды SA4 с программированием в arduino. Принципиальная схема устройства. Макетная плата устройства. Знакомство с элементом семисегментник, подключение. Ресурсы электронных компонентов. Оптимизация кода, таблица истинности для семисегментника.

Использование кнопок, их программирование, использование фоторезистора. Цифровая и аналоговая информации. Система Arduino. Естественный рабочий процесс платы Arduino.

Практические работы:

«Светофор» в различных исполнениях. Вывод значений на семисегментнике.

3. Работа с проектами

Теория: Знакомство с электронными элементами (светодиоды, кнопки, резисторы, потенциометры, джойстиком, оптроном, семисегментным индикатором, матрицей). Применение элементов на практике, составление проектов, внесение новых идей в разработку новых проектов на основе проведенной практики.

Практические работы:

Плавное изменение яркости диода. Подключение кнопки. Кнопочное переключение цвета. Кнопочный маячок. Трехцветный светодиод. Радуга цветов. Моргание шестью диодами. Терменвокс. Game-Ball через потенциометр. Вывод уровня яркости на семисегментник. Джойстик, управление танком. Джойстик, разработка карты и стрельба. Кнопочные ковбои. Подсчет нажатий на кнопку.

Календарный учебный график

<i>№ темы</i>	<i>Тема</i>	<i>Краткое содержание</i>	<i>Практические работы</i>	<i>Формируемые результаты</i>	<i>Количество часов</i>
	<i>Введение</i>				2
1	<i>Техника безопасности при работе с ПК</i>	Изучение правил техники безопасности при работе в компьютерном классе (лаборатории)		Ученик должен знать правила техники безопасности при работе с ПК в компьютерном классе	1
2	<i>Микроэлектроника, основные понятия, сферы применения</i>	Роль микроэлектроники на современном этапе развития общества, основные понятия микроэлектроники.		Ученик должен знать и понимать широкое применение микроэлектроники в современном мире, представлять масштабы распространения	1

	<p>ДЗ Составить список устройств, в состав которых входит электроника. Расписать задачи, выполняемые электроникой для одного из устройств. Ссылки: www.youtube.com-техника безопасности(пример)</p>				
	<p>Знакомство со средой программирования и регистрация на ресурсах</p>			<p>10</p>	
3	<p>Организационные моменты. Регистрация. Установка.</p>				<p>1</p>
	<p>Создание гугл аккаунта. Регистрация на ресурсе tinkercad.com tinkercad S4A mBlock</p>	<p>Подготовка к работе, регистрация на сайтах, ознакомление с рабочими системами <i>Tinkercad Circuits</i> и S4A (mBlock). Демонстрация примитивных работ в <i>Tinkercad Circuits</i>, mBlock и S4A</p>		<p>Ученик должен иметь гугл аккаунт, зарегистрирован на ресурсах tinkercad.com https://scratch.mit.edu/ Ученик должен уметь скачать требуемое приложение S4A (mBlock) на компьютер и установить</p>	
	<p>ДЗ Под своей учетной записью найти на сайте https://scratch.mit.edu/ несложную разработку понятное ученику, скопировать к себе. На сайте tinkercad.com посмотреть и так же подобрать разработку. Ссылки: robo-hunter.com-началоработы с S4A, circuits.io, scratch.mit.edu, google.ru, www.mblock.com</p>				
4	<p>Блочное программирование <i>scratch</i>. Линейный алгоритм.</p>			<p>1</p>	
	<p>Требуемые блоки для разработки, комментарии алгоритма Первая разработка в среде S4A (mBlock) Корректировка, замена повторяющихся блоков</p>	<p>Знакомство с работой среды S4A (mBlock). Разработка виртуального светофора по алгоритму. Чтение линейного алгоритма. Замена повторяющихся блоков на цикл n раз.</p>	<p>Практическая №1 Виртуальный светофор (в <i>Scratch</i>, mBlock)</p>	<p>Ученик должен знать расположение основных элементов системы скретч. Научится читать линейный алгоритм. Выделять в алгоритме повторяющиеся блоки и заменять их циклом. Научиться в ней ориентироваться и быстро набирать</p>	<p>1</p>

				указанные блоки	
<p>ДЗ повторить разработку светофора в scratch и опубликовать у себя на страничке https://www.youtube.com/watch?v=vd20J2r5wUQ&list=PLMIinhDclNR1GsZ9CJBZESbm7k3Xpr7awy Материал 1</p>					
5-6	<p>Плата Arduino, разновидности, аппаратная часть. Техника безопасности при работе с платой и малым током.</p>				2
	<p>4.1 Знакомство с платой Ардуино. 4.2 Аппаратная часть. 4.3 Техника безопасности с малым током. 4.4 Светодиод, сопротивление, макетная плата 4.5 Сбор светофора</p>	<p>Изучение правил техники безопасности при работе с электронными компонентами. Составные элементы платы Arduino Светодиод, сопротивление, макетная плата, как электронные комплектующие.</p>	<p>Практическая №2 Сборка светофора на макете с платой ардуино.</p>	<p>Знать технику безопасности, при работе с малым током и платой Arduino. Представлять какие контакты и элементы платы Arduino за что отвечают и какие возможности предоставляют</p>	.
<p>ДЗ сделать корпус светофора из картона, под собранный макет. Ссылки: arduino-kit.com.ua, amperka.ru-светодиод, amperka.ru-резистор, amperka.ru-беспаяная плата, youtube.com-TB http://wiki.amperka.ru/%D0%BF%D1%80%D0%BE%D0%B4%D1%83%D0%BA%D1%82%D1%8B:arduino-uno https://www.youtube.com/watch?v=rJcs-Q-vzPI</p>					
7	<p>Цифровые и аналоговые контакты ввода-вывода, при блочном программировании в S4A (mBlock)</p>				1
	<p>Блоки управления контактами платы Arduino в среде S4A (mBlock) Переменные Плавное переключение диодов. Алгоритм плавного погасания диода. Светофор с плавным вкл/откл</p>	<p>Изучение требуемых блоков для подачи напряжения на диоды. Разбор алгоритма плавного погасания</p>	<p>Практическая №3 Светофор на диодах</p>	<p>Закрепить полученные знания работы с платой, определения требуемых контактов. Знать блоки управления входами-выходами в среде S4A (mBlock)</p>	
<p>ДЗ составить разработку из 5 диодов, размещенных по кругу, и организовать мигание их по кругу</p>					

	<i>Приложение 2 - светофор.</i>				
8	<i>Разработка устройства с <u>tinkercad Circuits</u></i>				1
	<i>Базовые операторы языка Arduino Соответствие блоков операторам Первая разработка на ресурсе tinkercad.com Принципиальная схема устройства Разработка макетной платы для устройства</i>	<i>Разбор соответствия блоков среды SA4 с программированием в arduino Принципиальная схема устройства Макетная плата устройства</i>	<i>Практическая №4 Составляем светофор в среде <u>Tinkercad Circuits</u></i>	<i>Ученик должен познакомиться с виртуальной средой. Закрепить на практике основные команды ардуино соответствующие блокам в скретч. Познакомится с принципиальными схемами устройств и макетными платами.</i>	5 мин. 10 мин. 20 мин. 5 мин. 5 мин.
	<i>ДЗ составить принципиальную схему и макетную плату для разрабатываемого устройства в среде <u>tinkercad Circuits</u></i>				
9-10	<i>Электронный компонент для вывода сигнала. Семисегментник.</i>				2
	<i>Семисегментник, применение. распиновка семисегментника. Подключение к плате Объяснение алгоритма первых двух цифр на программу Программа на S4A Программа на <u>tinkercad Circuits</u> Таблица истинности выводов семисегментника 7.7 Оптимизация кода</i>	<i>Знакомство с элементом семисегментник, подключение. Ресурсы электронных компонентов. Закрепление пройденного материала. Оптимизация кода, таблица истинности для семисегментника. Таблица истинности.</i>	<i>Практическая №5 подключение к макетной плате семисегментника и платы Arduino Практическая №6 Составляем счет от 0 до 9 на семисегментнике. Первый час теория и объяснение для цифр 1,2</i>	<i>Ученик должен знать что такое распиновка и как найти информацию для нужного компонента (на примере семисегментник) ,уметь применить ее на практике. Ученик должен понять принцип составления таблицы истинности и ориентироваться в ней.</i>	5 мин. 10 мин. 15 мин. 15 мин. ДЗ 1 20 мин. 10 мин. 15 мин.

	<i>ДЗ 2 составить разработку с применением семисегментника, мигающий индикатор по восьмерке</i> <i>Приложение 3 - счет на семисегментнике через S4A</i>				
11	<i>Электронные компоненты ввода информации кнопка и фоторезистор.</i>			1	
	<i>Применение кнопок и фоторезистора</i> <i>Цифровая аналоговая информация.</i> <i>Практическая 7 на Autodesk Circuits</i>	Использование кнопок, их программирование, использование фоторезистора. Цифровая аналоговая информации.	Практическая №7 Решение нескольких небольших задач на Скретч. Изменение яркости светодиода от фоторезистора. Включение трех диодов по очереди от нажатия на кнопку.	Ученик должен понимать назначение кнопки и фоторезистора. Уметь составлять небольшие программы с этими элементами.	10 мин. 5 мин. 30 мин.
	<i>ДЗ Составить в S4A программу по изменению яркости светодиода от фоторезистора.</i>				
	<i>Практическая №8</i>	<i>Практическая №8</i> Смоделировать систему уличного освещения из 7 светодиодных столбов, в зависимости от естественного освещения (С включением 2,3,4,5,7 столбов)			
12	<i>Система Arduino, установка, подключение и настройка.</i>			1	
	<i>9.1 Установка Arduino.</i> <i>9.2 Установка драйвера платы.</i> <i>9.3 Сбор устройства</i> <i>9.4 Подключение и загрузка кода.</i>	Система Arduino. Естественный рабочий процесс платы Arduino.	Практическая №9 Собрать схему освещения столбов и загрузить программу в плату Arduino.	Ученик должен уметь скачать ПО для Arduino, подключить и настроить.	10 мин. 5 мин. 20 мин. 10 мин.
	Работа с проектами			22	
13.	Плавное изменение яркости диода. Аналоговые контакты 5,6,9	Материал 4 https://www.youtube.com/			
14.	Кнопка. Цифровые контакты 2,3	Материал 6			

	Подключение кнопки.	http://wiki.amperka.ru/
15.	Арифметические и логические операторы. Кнопочное переключение цвета.	Материал 7
16.	Кнопочный маячок	Материал 8
17.	Трехцветный светодиод. Радуга цветов	Материал 9
		http://amperka.ru/product/rgb-led-5mm
18.	Моргание шестью диодами	Материал 10
		https://www.youtube.com/
19.	Потенциометр. Аналоговые контакты А0-А6	Материал 11
		http://wiki.amperka.ru/
20.	Терменвокс	https://www.youtube.com/
21.	Защита собственного проекта, рассмотренную ранее работу со своими изменениями.	
22.	Game-Ball через потенциометр	Материал 12
23.	Фоторезистор	Материал 13
		https://www.youtube.com/
24.	Семисегментник, счет	Материал 14
		http://wiki.amperka.ru/
25.	Семисегментник, процедурное программирование	Материал 15
	Вывод уровня яркости на семисегментник	
26.	Джойстик, управление танком	Материал 16
27.	Джойстик, разработка карты и стрельба	Материал 17
28.	Подключаем несколько контроллеров, танковые бои	Материал 18

29.	Динамик	Материал 21
		https://www.youtube.com/
30.	Сервопривод	Материал 22
	Подключение сервопривода	http://wiki.amperka.ru/
31.	Кнопочные ковбои	https://www.youtube.com/
32.	Подсчет нажатий на кнопку	https://www.youtube.com/
33.	Защита собственного проекта, рассмотренную ранее работу со своими изменениями	
34.	Резерв	

Материально-технические условия реализации программы

Для организации занятий по «Схемотехнике с контроллером Arduino» требуется учебный кабинет и определенное оснащение образовательного процесса.

Оборудование: столы, стулья, учебная доска, проектор, компьютеры (ноутбуки), комплекты arduino.

Программное обеспечение: S4A, mBlock, Arduino.

Интерактивные ресурсы: Tinkercad Circuits, scratch.mit.edu

Методическое обеспечение программы

Материал 1. Среда S4A

Scratch разрабатывался как новая учебная среда для обучения школьников программированию. В Scratch можно создавать фильмы, играть с различными объектами, видоизменять их вид, перемещать их по экрану, устанавливать формы взаимодействия между объектами. Это объектно-ориентированная среда, в

которой блоки программ собираются из разноцветных кирпичиков команд точно так же, как собираются из разноцветных кирпичиков конструкторы Лего.

В результате выполнения простых команд может складываться сложная модель, в которой будут взаимодействовать множество объектов, наделенных различными свойствами. Начальный уровень программирования настолько прост и доступен, что Scratch рассматривается в качестве средства обучения не только старших, но и младших школьников[7].

Для работы нам понадобятся:

- плата ардуино uno (nano);
- набор комплектующих для работы с контроллером;
- среда программирования S4A.

S4A воспринимает контроллер как устройство ввода-вывода ПК и его программированием не занимается, поэтому для работы с ней требуется записать прошивку S4AFirmware15.ino скачав ее по адресу: <http://s4a.cat/downloads/S4AFirmware15.ino>

Также загрузить саму программу S4A возможно с официального сайта <http://s4a.cat/> в разделе Downloads.

Интерфейс программы представлен на Рисунке 2.1

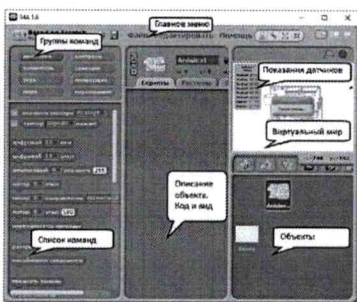



Рисунок 2.1 – Интерфейс S4A

При создании скриптов используется палитра блоков, которая занимает левую часть экрана. В ее верхней части располагается 8 разноцветных кнопок, которые выбирают нужную группу команд.

Команды выбранной группы отображаются списком команд.

В S4A, плата Arduino представляет собой особый вид спрайта (объекта). Объект Arduino автоматически найдет USB-порт, к которому подключена плата.

Существует возможность подключения к нескольким платам одновременно, просто добавьте новый спрайт Arduino .

Матриал 2.

Устройства ввода-вывода

С компьютером используется большое количество периферийных устройств, которые в свою очередь возможно подразделить на устройства ввода, вывода и хранения информации.

Так все устройства, которые позволяют воспринимать информацию предназначенную для компьютера, относятся к устройствам ввода.

Все устройства, которые позволяют передавать или воспроизводить информацию для человека (иного объекта) относятся к устройствам вывода.

Сегодня устройства становятся все более и более функциональные и таким образом устройство будет относиться к устройствам ввода-вывода. (Так, МФУ состоит из принтера – устройства вывода и сканера – устройства ввода.)

Таким образом нам известны такие устройства ввода, как компьютерная мышка, клавиатура, веб-камера, сканер, микрофон... и устройства вывода – монитор, колонки, проектор, принтер....

Контроллер Arduino Uno, как устройство ввода-вывода

Контроллер Arduino Uno мы будем рассматривать в рамках возможностей S4A, поэтому описание будет урезано и относиться только применительно с S4A.

Arduino Uno является устройством ввода-вывода, к которому подключаются различные элементы, позволяющие выполнять функции ввода или вывода.

Следует выделить следующие контакты контроллера Arduino Uno:

- цифровой ввод (2, 3);
- цифровой вывод (10, 11, 12, 13);
- аналоговый ввод (A0, A1, A2, A3, A4, A5);
- аналоговый вывод (5, 6, 9);
- вывод управления мотором (4, 7, 8)
- контакты питания

- контакт питания 5V
- контакт питания 3.3V
- контакт GND (заземления)

Контакты питания требуются для грамотного подключения устройств ввода-вывода, их следует подключать так, как будет показано на рисунках.

Обратите внимание на внешний вид контроллера и где расположены контакты на плате (Рисунок 2.2).

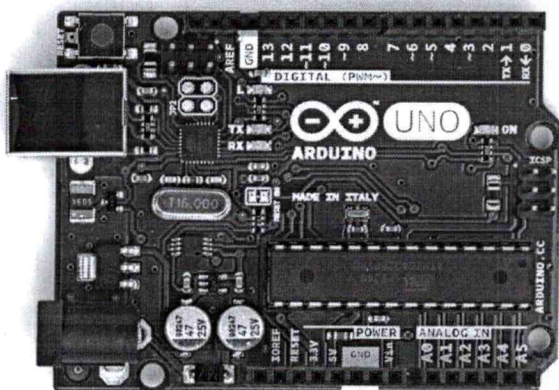


Рисунок 2.2 – Внешний вид Arduino Uno

Макетная плата

Для надёжной сборки устройств создаются индивидуальные печатные платы. Для быстрой сборки электрических схем без пайки и без проблем существует макетная плата.

Под слоем пластика скрываются медные пластины-рельсы, выложенные по незамысловатому принципу, как на Рисунке 2.3. Эти пластинки позволяют соединять провода создавая электрическую цепь. Важно, чтобы элемент ввода (вывода) заработал он должен подключен полной цепью [6].

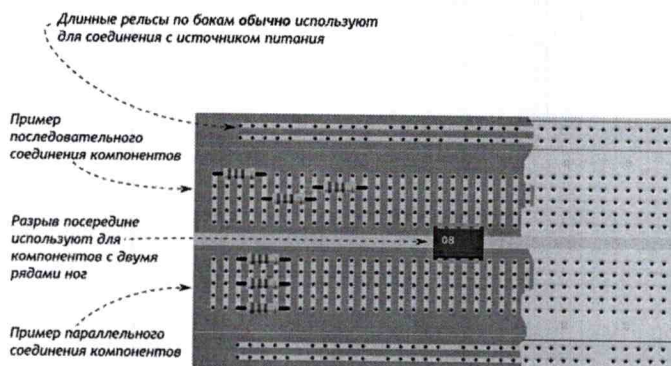


Рисунок 2.3 - Макетная плата в разрезе

(рисунок с сайта <http://wiki.amperka.ru>)

В своих проектах мы будем подключать 5V и GND к общим линиям.

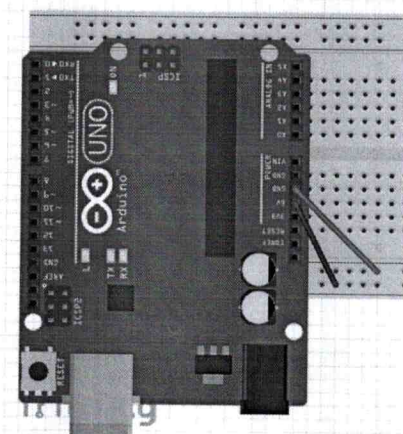
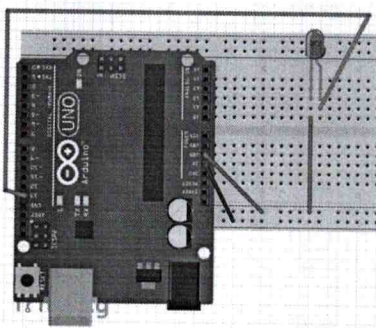


Рисунок 2.4 – Подключение 5V и GND к макетной плате

Материал 3. Элемент вывода, светодиод

Светодиод (англ. Light Emitting Diode или просто LED) – устройство вывода в виде миниатюрной «лампочки». Элемент светится, когда через него проходит



ток от анода (+) к катоду (-).

Рисунок 2.5 – Пример подключения светодиода

Цифровые контакты (10,11,12,13)

Для диода отлично подходят цифровые контакты 10, 11, 12, 13. Для их управления в группе «Движения» используются команды:

(для включения) и (для отключения).

По нажатию на треугольник выпадает список контактов для выбора нужного.

Моргание диодом

Для выполнения данного задания, следует подключить диод по рисунку 2.5.

В окне программы перенесем в область скриптов следующие блоки:

контроль	когда клавиша пробел нажата	Будет выполнять скрипт под блоком по нажатию на указанную клавишу
	ждать 1 секунд	Делает паузу указанное время в секундах
движение	цифровой 13 вкл	Активирует указанный контакт
	цифровой 13 откл	Отключает указанный контакт

Составим скрипт блоков как показано на Рисунке 2.6.

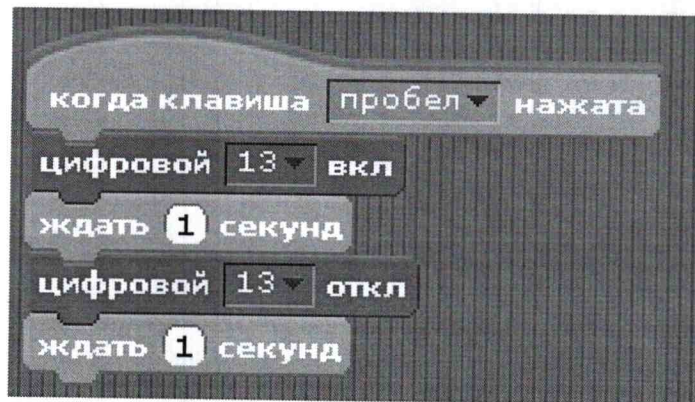


Рисунок 2.6 – Моргание светодионом по нажатию на пробел

Из указанных блоков составим алгоритмы по Рисунку 2.7



Рисунок 2.7 – Моргание светодионом по нажатию на стрелки

Блоки "всегда", "повторить", "когда щелкнуть по ..."

Чтобы сделать моргание светодиода постоянным, можно воспользоваться блоками повторения.

контроль	всегда	Блок повторяет содержимое внутри
	повторить 10	Блок повторяет указанное число раз
	когда щелкнут по [мышь]	Активирует скрипт по запуску программы

Соберите скрипт, указанный на Рисунке 2.8.

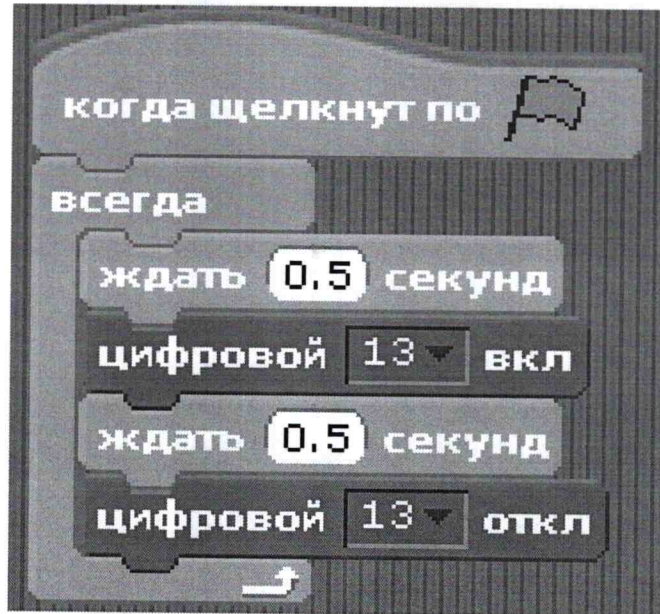





Рисунок 2.8 – Моргание светодиодом

Нажмем на значок флажка  в верхней правой области приложения, тем самым запустив проект. По значку красного круга – остановим проект.

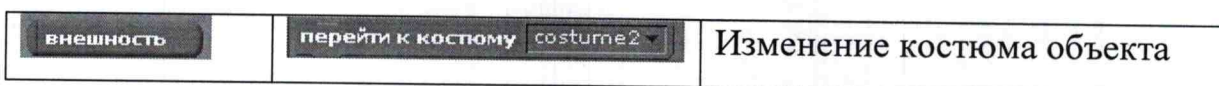
Давайте сделаем моргание на нашей плате симметрично диоду, для этого создадим костюмы (Рисунок 2.9) и скорректируем скрипт.

У нашего объекта, кроме скриптов имеются вкладки костюмы и звуки , перейдите на вкладку костюмы и нажмите кнопку редактировать .

Постарайтесь дорисовать красный круг ниже платы. Сохраните изменения.

Скопируйте костюм и отредактируйте цвет круга на черный.

Воспользуйтесь группой команд внешность, добавьте блоки перейти к костюму.



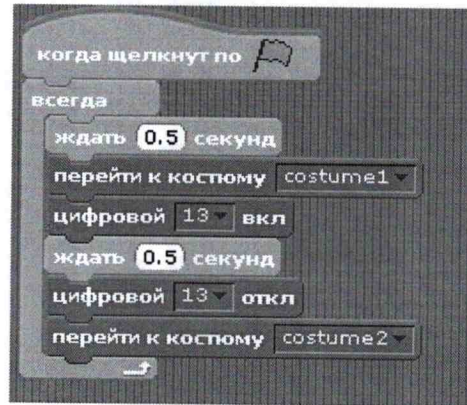



Рисунок 2.9 – Костюмы и скрипт для моргания диода и платы

Материал 4.

Аналоговый контакты (5, 6, 9)

Аналоговые контакты 5, 6 и 9 предназначены для подключения элементов вывода. Они могут принимать значения в диапазоне от 0 до 255, при использовании блока аналоговый значение .

Плавное изменение яркости диода

Если диод подключить к аналоговому входу 5 (6 или 9), то возможно задавать ему значение от 0 до 255, в свою очередь это значение будет влиять на яркость этого диода.

Подключим диод к 5-му контакту и напишем скрипт плавного изменения яркости (Рисунок 2.10).

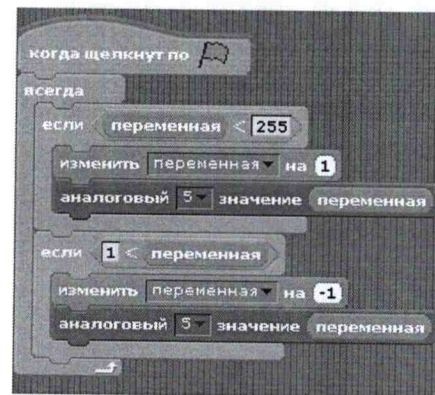
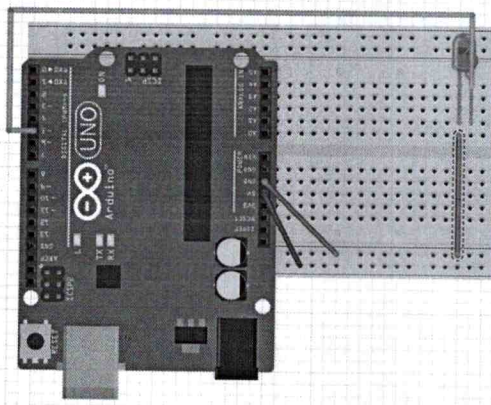


Рисунок 2.10 – Схема подключения диода и скрипт плавного изменения яркости

Материал 5. Проект светофора

Для данного проекта нам понадобится три светодиода красного, желтого и зеленого цветов. Подключим светодиоды, как показано на Рисунке 2.11.

Данные светодиоды будут подключены к 13, 12 и 11 контактам, а значит для управления светодиодами мы будем использовать именно их.

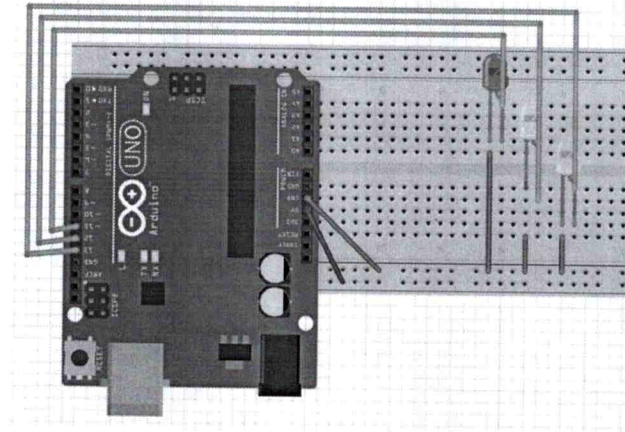


Рисунок 2.11 – Подключение светодиодов для светофора

Прежде, чем писать скрипт распишем алгоритм нашего светофора:

- горит красный 5 секунд;
- загорается желтый на 5 секунд, в это-же время красный моргает 3-и раза за 3-и секунды;
- тухнет желтый и загорается зеленый;
- загорается желтый на 5 секунд, в это-же время зеленый моргает 3-и раза за 3-и секунды.

Попробуйте самостоятельно составить свой скрипт для собственного светофора.

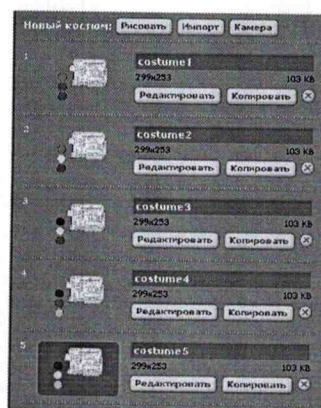


Рисунок 2.12 – Костюмы светофора

Также на мониторе сделаем отображение светофора, для этого понадобятся костюмы (Рисунок 2.12):

- горит красный (costume1);
- горит красный и желтый (costume2);
- горит желтый (costume3);
- горит зеленый (costume4);
- горит зеленый и желтый (costume5).

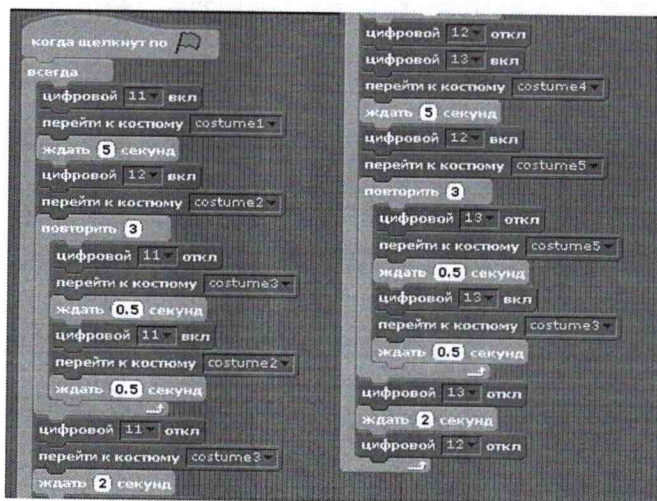
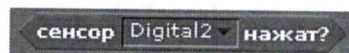


Рисунок 2.13 – Скрипт работы светофора

Материал 6. Кнопка.

Цифровые контакты (2,3)

Цифровые контакты 2 и 3 предназначены для подключения элементов ввода. При этом значение этих элементов должны принимать два параметра – наличие или отсутствие сигнала при использовании блока сенсор нажат



Наилучшим примером такого элемента будет кнопка. И значение в диапазоне от 0 до 255, при использовании блока значение сенсора



Элемент ввода, кнопка

Элемент достаточно простой, в котором имеются два положения зажато и отпущено. Вот эти два положения кнопка передает компьютеру.

Пример подключения кнопки показан на Рисунке 2.14.

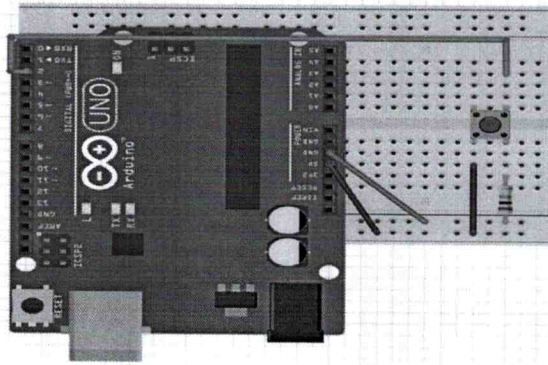


Рисунок 2.14 – Пример подключения кнопки

Материал 7. Арифметические операторы

Арифметические операторы выполняют не сложные арифметические действия, которые требуются для расчета. Когда требуется сделать более сложную операцию, то вкладывают один оператор в другой:

	Одиночный арифметический оператор
	Несколько арифметических операторов
	Действие $2+(2*2)$ результат будет 6
	Действие $(2+2)*2$ результат будет 8
	Арифметическое действие с переменной

Следует строго следить за последовательность действий арифметики.

Часто в арифметических операциях участвуют переменные. Изначально в группе переменный лежат только два блока **Создать список** и **Создать переменную**. По щелчку на кнопке "Создать переменную" появляется окошко, в котором необходимо напечатать имя переменной (Рисунок 2.15).

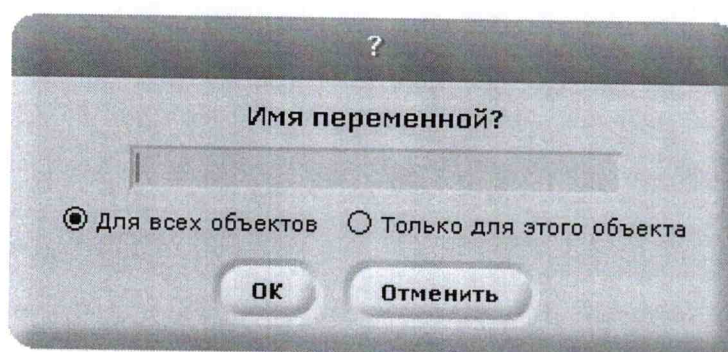


Рисунок 2.15 – Окно добавления переменной

После того как переменная создана, можно устанавливать, изменять и передавать это значение другим исполнителям.

Если переменная не нужна, её можно удалить: **Удалить переменную**

Операторы сравнения

Операторы сравнения выполняют логическую операцию и дают двусмысленный ответ ДА или НЕТ. Используется в условных блоках.

	знак меньше
	знак равно
	знак больше
	пример блока всегда с условием
	знак меньше или равно

Кнопочное переключение цвета

Подключите к имеющей схеме две кнопки ко второму и третьему контактам, как показано на картинке 3.3.

а) Составьте скрипт который при нажатии на одну кнопку будет увеличивать яркость, а при нажатии на вторую – уменьшать.

б) Составьте скрипт который при зажатии кнопки будет увеличивать яркость, а когда кнопка отпущена уменьшать.

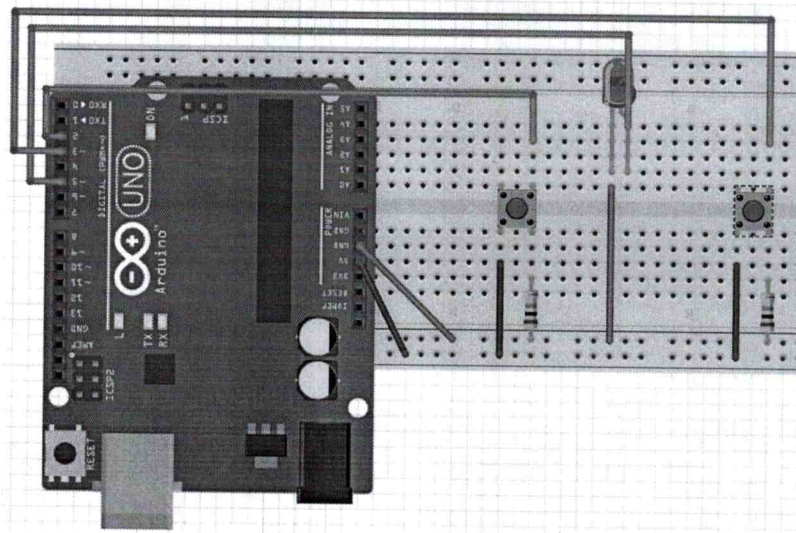


Рисунок 2.16 – Схема подключения диода и двух кнопок

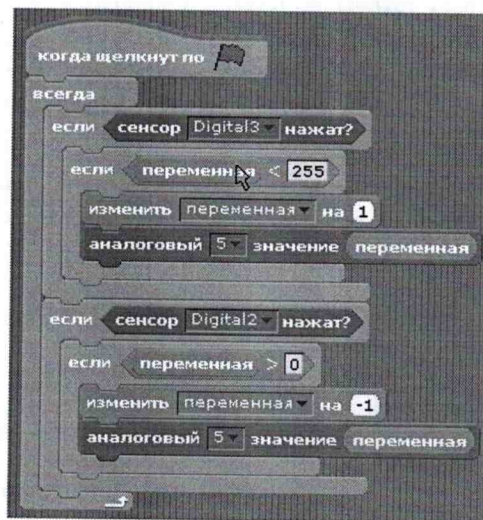


Рисунок 2.17 – Скрипт обработки нажатия кнопок (а)

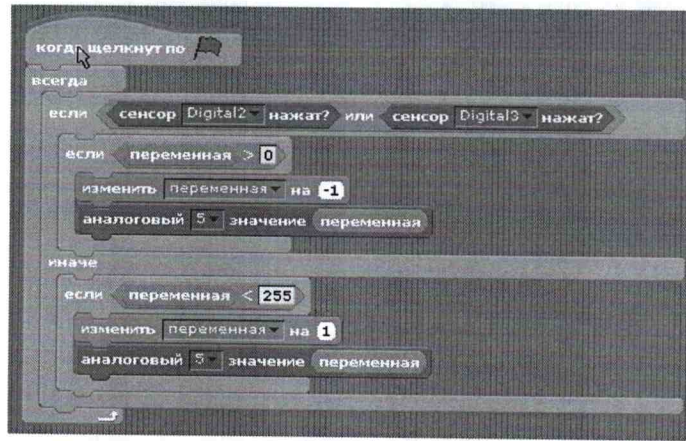


Рисунок 2.18 – Скрипт обработки нажатия кнопок (б)

В последнем случае используется блок ИЛИ, что будет, если за место него использовать блок И. Объясните изменение работы устройства.

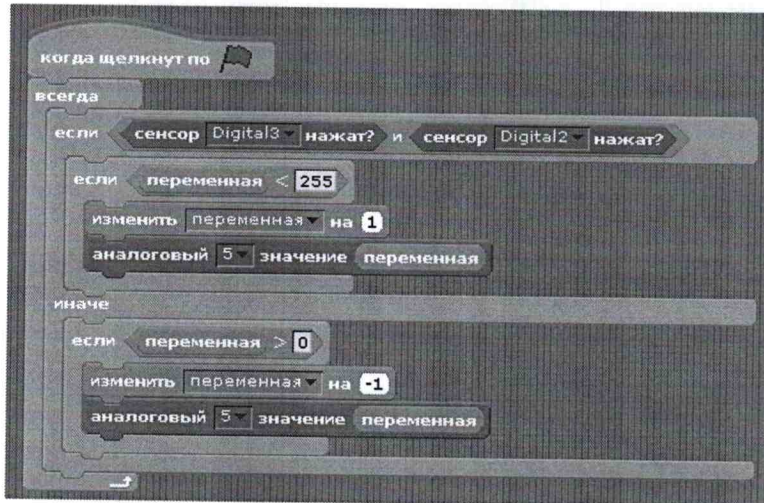


Рисунок 2.19 – Скрипт обработки нажатия кнопок (в)

Материал 8. Кнопочный маячок

Подключите два светодиода и одну кнопку, как на Рисунке 2.20.

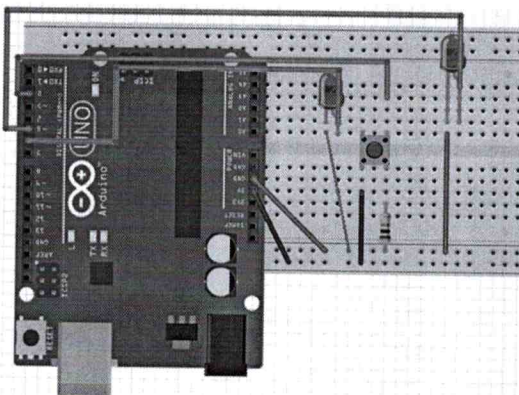


Рисунок 2.20 – Кнопочный маячок

Материал 9. Радуга цветов, через трехцветный диод

Трехцветный светодиод в своем составе имеет три диода красного, зеленого и синего цветов. При сочетании этих цветов получаются различные оттенки.

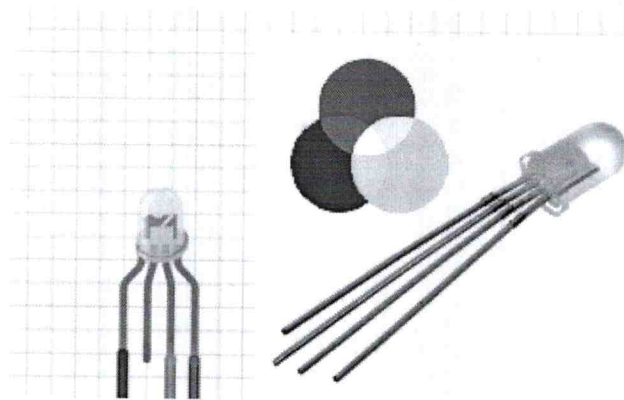


Рисунок 2.21 – Трехцветный светодиод

Подключим светодиод следующим принципом (Рисунок 2.22):

первый контакт (красный цвет) – 5 контакт контроллера

второй контакт (Ground) – GND контакт на макетной плате

третий контакт (зеленый цвет) – 6 контакт контроллера

четвертый контакт (синий цвет) – 9 контакт контроллера

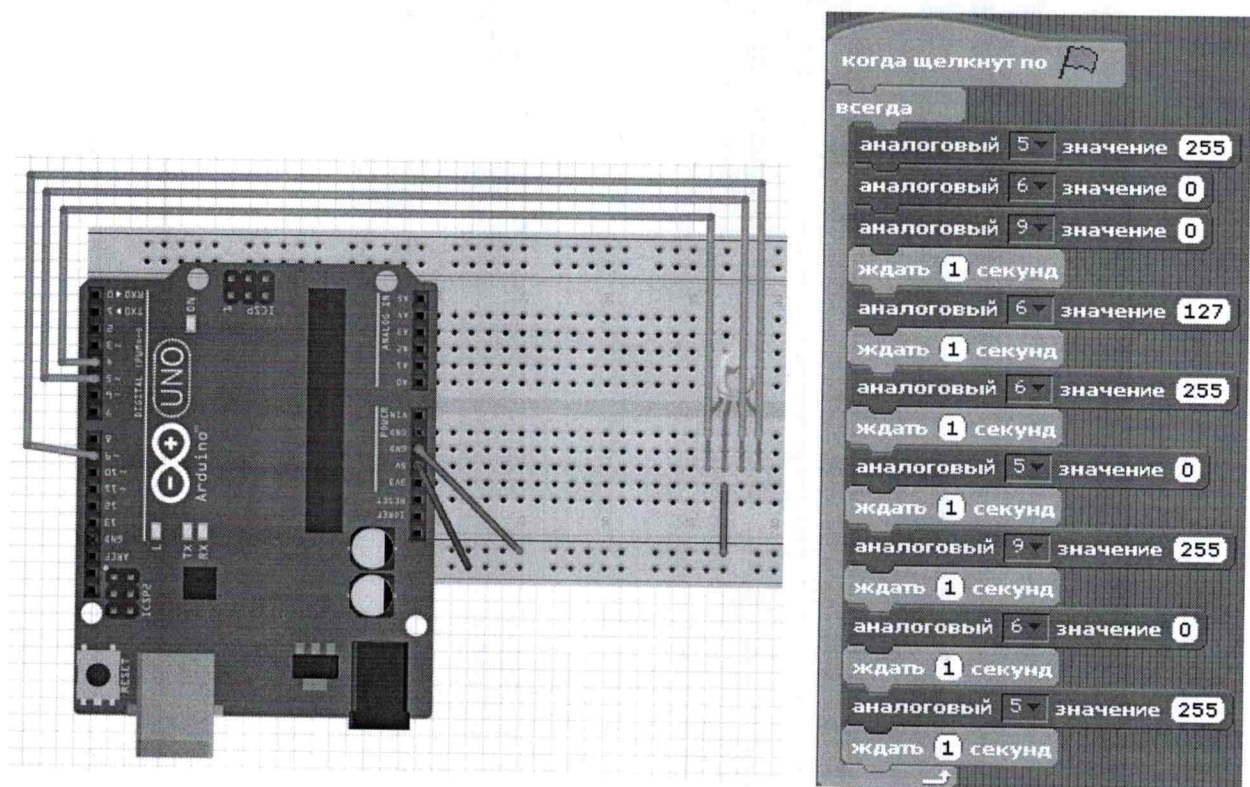


Рисунок 2.22 – Подключение трехцветного светодиода

Нам требуется написать скрипт, который по очереди будет подсвечивать на светодиоде семь цветов радуги. Красный, синий и зеленый цвета понятные, а вот остальные цвета будут сочетанием имеющихся трех.

	Значение красного диода (5)	Значение зеленого диода (6)	Значение синего диода (9)
Красный	255	0	0
Оранжевый	255	127	0
Желтый	255	255	0
Зеленый	0	255	0
Голубой	0	255	255
Синий	0	0	255
Фиолетовый	255	0	255

Возможно написать программный код, где цвета будут перебираться последовательно. Для этого:

- задаем начальное значения 0,0,0;
- поднимаем значение красного до 255;
- поднимаем значение зеленого до 255;
- опускаем значение красного до 0;
- поднимаем значение синего до 255;
- опускаем значение зеленого до 0;
- поднимаем значение красного до 255;
- опускаем значение синего до 0;

Конечно, это не будут все возможные оттенки, но давайте посмотрим, что из этого может получиться (Рисунок 2.23).



Рисунок 2.23 – Плавное изменение цветов

Материал 10. Моргание шестью диодами

Данный проект будет смотреться довольно эффектно при аккуратном расположении всех элементов. Задача и реализация не сложная.

Требуется расположить шесть светодиодов в небольшой круг и прописать по очереди для каждого включение и выключение. Результатом выполнения будет мигание диодов по кругу. Отмечу важным не запутаться в подключении диодов и выписать последовательность подключения по кругу. В моем случае 6, 9, 10, 13, 11, 12 (Рисунок 2.24)

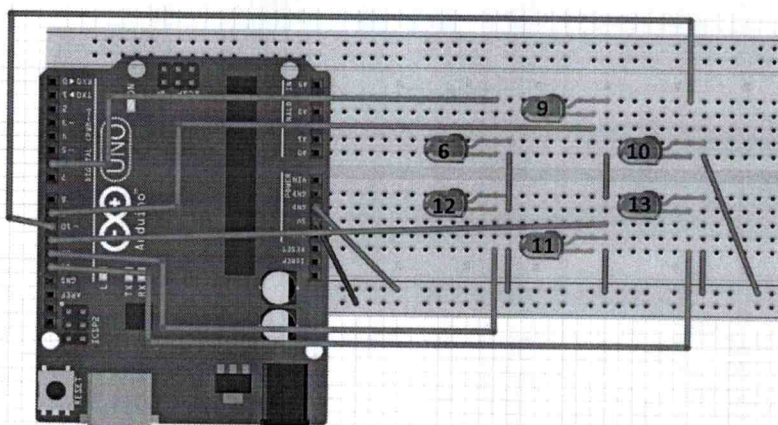


Рисунок 2.24 – Подключение диодов (мигание по кругу)

Программный код не затейливый – последовательное включение и выключение диодов. Возможны варианты исполнения, когда будут гореть два (один, три, четыре или пять), а остальные нет.

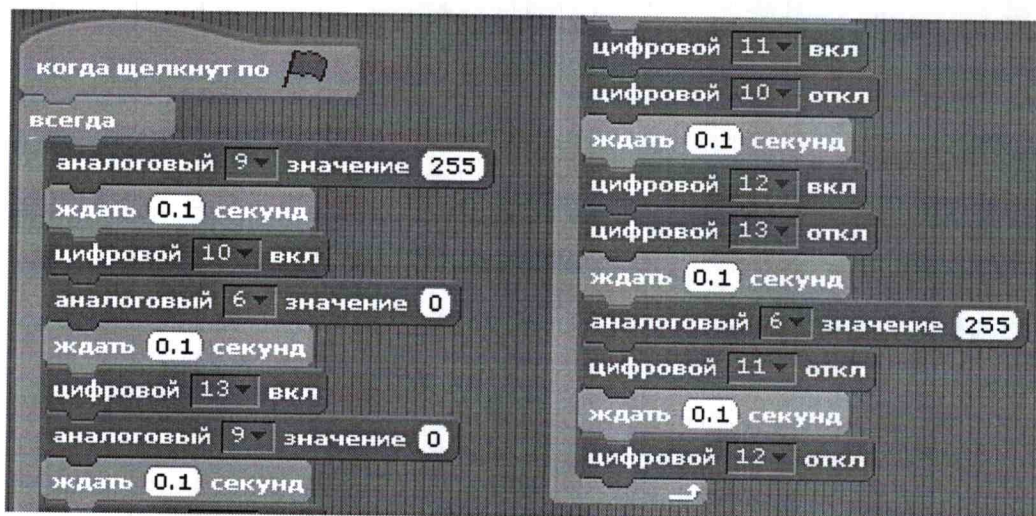


Рисунок 2.25 – Мигание по кругу (горят два диода)

Материал 11. Аналоговые контакты (А0-А6)

Аналоговые контакты А0-А6 предназначены для подключения элементов ввода. Они могут принимать значения в диапазоне от 0 до 1023, при использовании блока аналоговый значение значение сенсора Analog0.

Элемент ввода, потенциометр

Потенциометр – это элемент, рассчитывающий положение поворота, через сопротивление. Этот параметр он передает в качестве элемента ввода.

Подключение потенциометра показано на Рисунке 2.26.

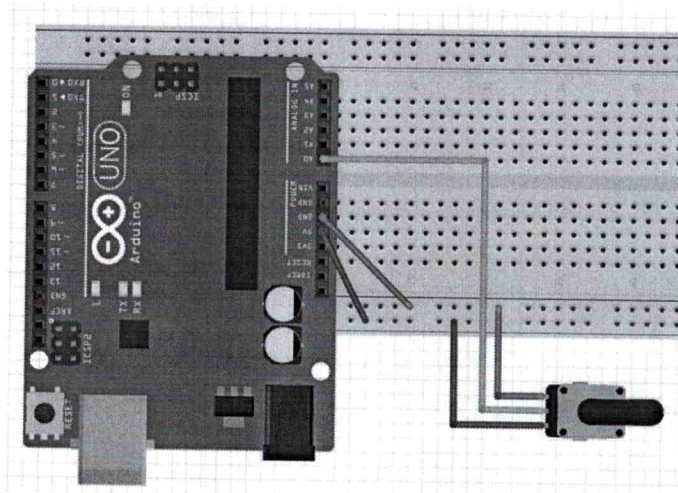


Рисунок 2.26 – Подключение потенциометра

Логические операторы И, ИЛИ, НЕ

Ранее мы затрагивали эти операторы. Разберемся с ними более подробно.

Выражение – утверждение, на которое можно дать ответ в виде ДА/НЕТ.

Оператор НЕ преобразует результат утверждения на противоположное.

Выражение1 – «Два плюс два равно четыре»	
Выражение2 – «Кошка имеет пять ног»	
Выражение1 = ДА	НЕ (Выражение1) = НЕТ «Два плюс два НЕ равно четыре»
Выражение2 = НЕТ	НЕ (Выражение2) = ДА «Кошка НЕ имеет пять ног»


Операторы И, ИЛИ объединяют два выражения в сложное выражение. При этом результат ДА с оператором ИЛИ будет в том случае, когда хотя бы одно из

выражений имеет результат ДА. Результат ДА с оператором И будет в том случае, когда оба выражения будут иметь результат ДА.

Выражение1 – «Два плюс два равно четыре»	
Выражение2 – «Кошка имеет пять ног»	
Выражение1 И Выражение2 ДА И НЕТ	«Два плюс два равно четыре» И «Кошка имеет пять ног» = НЕТ
Выражение1 ИЛИ Выражение2 ДА ИЛИ НЕТ	«Два плюс два равно четыре» ИЛИ «Кошка имеет пять ног» = ДА

Операторы можно совмещать, а выражение будет все более сложным. Главное не запутаться и аккуратно расписывать выражение.

Материал 12. Game-Ball через потенциометр

Добавим объекты в наше приложение. Для этого щелкнем на кнопку рисовать новый объект или открыть новый объект из файла .

Нам потребуется небольшой круг. Сдвинем плату, чтобы ее не было заметно и скроем панель сенсоров.

Любой объект имеет направление и положение, пример показан на Рисунке 2.27.

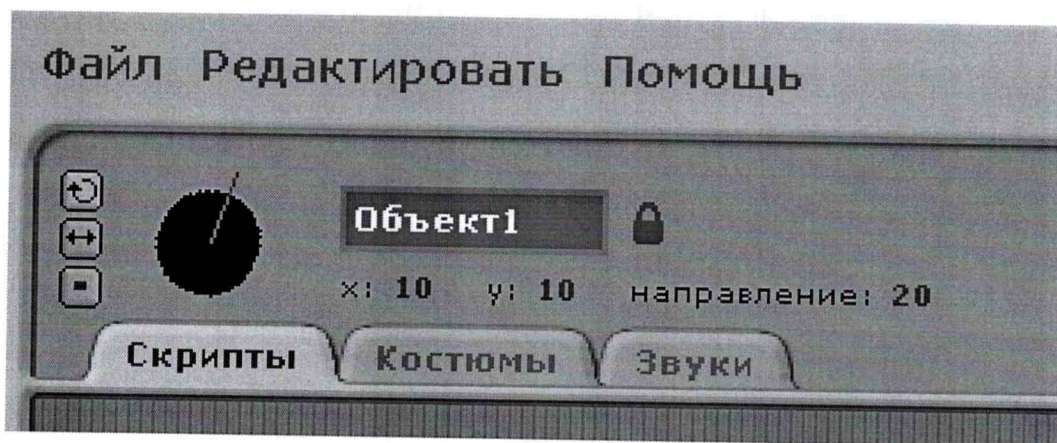


Рисунок 2.27 – Направление и положение нашего объекта

Напишем небольшой скрипт для нашего объекта в виде шара, как на Рисунке 2.28.

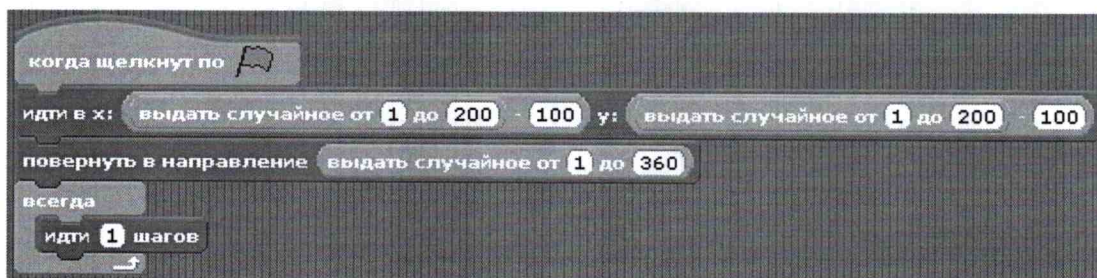


Рисунок 2.28 – Скрипт случайного появления и движения

Скрипт имеет небольшое количество действий. Оператор идти в ... перемещает объект в указанное место, при этом точка с координатами $x=0$, $y=0$ по умолчанию является центром экрана анимации. Оператор повернуть в направление разворачивает объект в случайное направление, а оператор идти шагов перемещает объект в этом направлении.

Но наш шарик уходит из области экрана и его становится не видно. Давайте заставим его при касании стенок отталкиваться (Рисунок 2.29).



Рисунок 2.29 – Скрипт летающего объекта

Добавим еще два объекта в виде прямых линий. Одна во всю ширину окна, вторая на небольшую его часть, как на Рисунке 2.30.



Рисунок 2.30 – Объекты приложения

Добавим в скрипт обработку касания шара с нашими полосами.

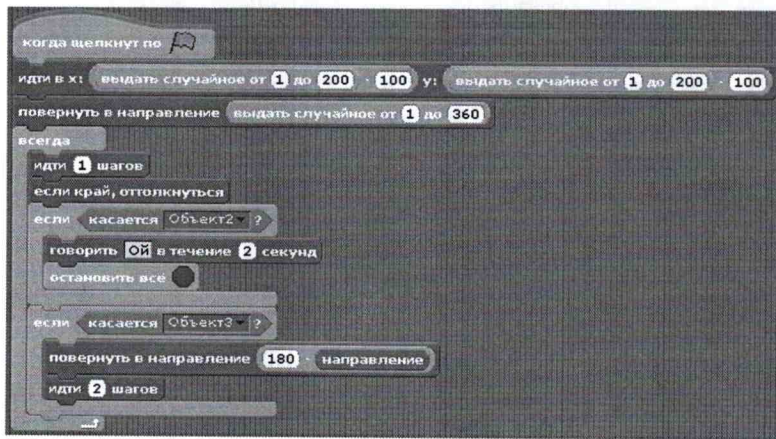


Рисунок 2.31 – Скрипт движения шарика

В зависимости от ширины доски и рабочего поля следующий кусок кода может быть скорректирован. Попробуйте подобрать наиболее подходящие значения в уравнение «Положение/2.5 – 300».

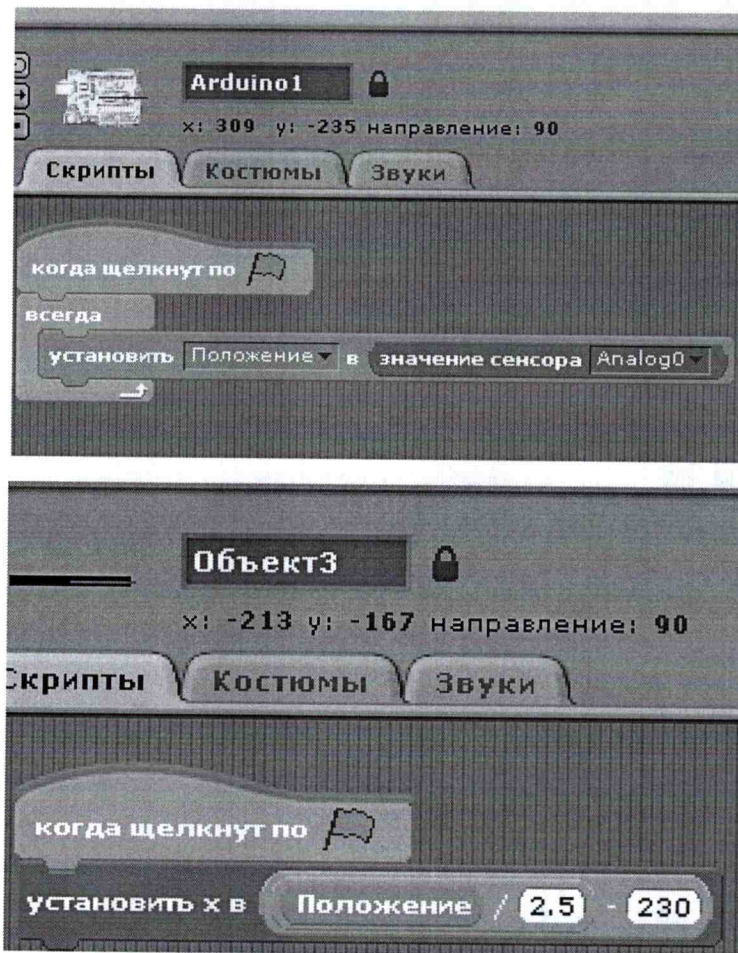


Рисунок 2.32 – Скрипты для управляющих объектов

Материал 13. Фоторезистор

Элемент ввода, фоторезистор

Фоторезистор – элемент, определяющий яркость падающего света на него и выражающего это значение числом диапазона от 0 до 1023. Подключение элемента осуществляется к контактам А0-А5. Пример подключения показан на Рисунке 2.33. В зависимости от модели фоторезистора следует использовать тот или иной номинал сопротивления. В моем случае я применял 220 Ом или 1КОм.

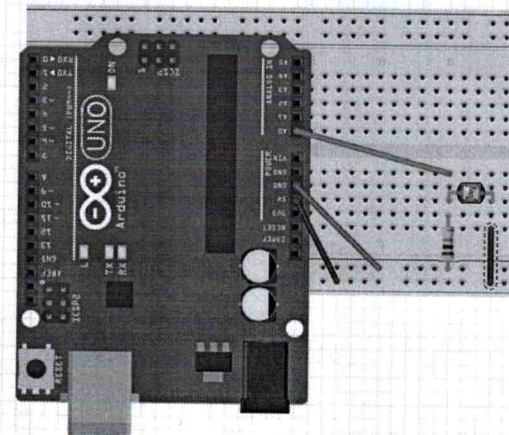


Рисунок 2.33 – Пример подключения фоторезистора

Пример работы с фоторезистором

Как мы уже знаем термистор передает значение в зависимости от уровня освещения попадаемого на него.

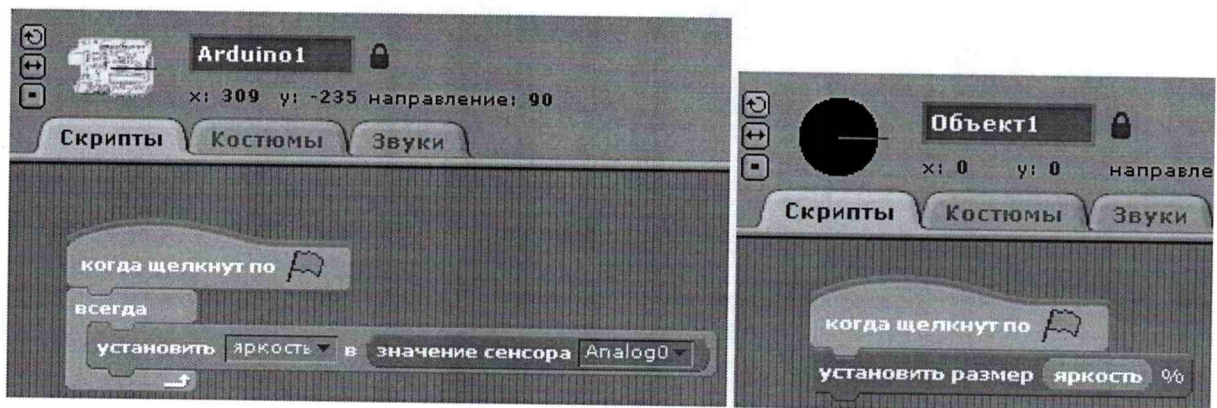


Рисунок 2.34 – Скрипт примера с термистором

Добавьте второй объект в виде шара, по Рисункам 2.31, 2.34 соберите приложение.

Материал 14. Элемент вывода, семисегментник

Семисегментник представляет из себя группу светодиодов, выстроенных в форме восьмерки. Каждый диод занимает свой сегмент. На рисунке 2.35 приведен пример подключения семисегментника.

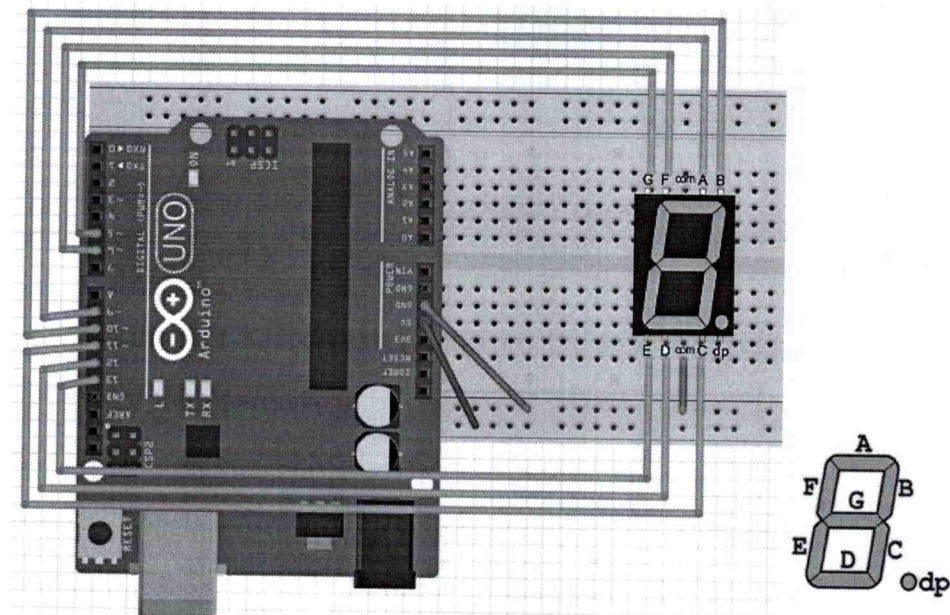


Рисунок 2.35 – Подключение семисегментника

Счет на семисегментнике

Для семисегментника по Рисунку 2.35 задействуются 5, 6, 9, 10, 11, 12, 13 контакты. Рассмотрим высвечивание той или иной цифры при данном подключении.

	8	8	8	8	8	8	8	8	8	8
A(9)	255	0	255	255	0	255	255	255	255	255
B(10)	Вкл	Вкл	Вкл	Вкл	Вкл	Откл	Откл	Вкл	Вкл	Вкл
C(11)	Вкл	Вкл	Откл	Вкл	Вкл	Вкл	Вкл	Вкл	Вкл	Вкл
D(12)	Вкл	Откл	Вкл	Вкл	Откл	Вкл	Вкл	Откл	Вкл	Вкл
E(13)	Вкл	Откл	Вкл	Откл	Откл	Откл	Вкл	Откл	Вкл	Откл
F(6)	255	0	0	0	255	255	255	0	255	255
G(5)	0	0	255	255	255	255	255	0	255	255

Для вывода мы чисел мы будем использовать процедуры, передавая возникновение параметра Диод А . Саму процедуру будем описывать начиная с блока когда я получу Диод А . Нам понадобится 10

процедур отображения чисел, а основной скрипт будет выглядеть, как на Рисунке 2.36.

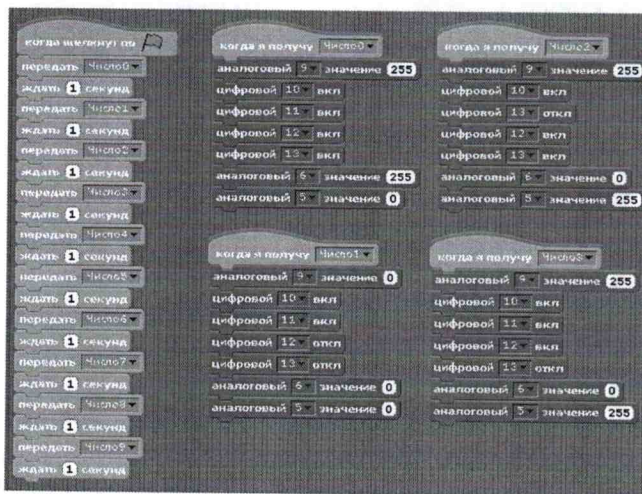


Рисунок 2.36 – Часть скрипта работы семисегментника

Попробуйте самостоятельно дописать процедуры для счета семисегментника.

Для счета есть множество решений и хочу предложить еще одно, записав в процедуру самые популярные значения сегментов.

Заметим, что для всех сегментов чаще всего используется включенное положение, за исключением сегмента E(13). Составим процедуру предустановка и прежде, чем записывать цифру будем вызывать ее (Рисунок 2.37). Такое состояние соответствует числу 9.

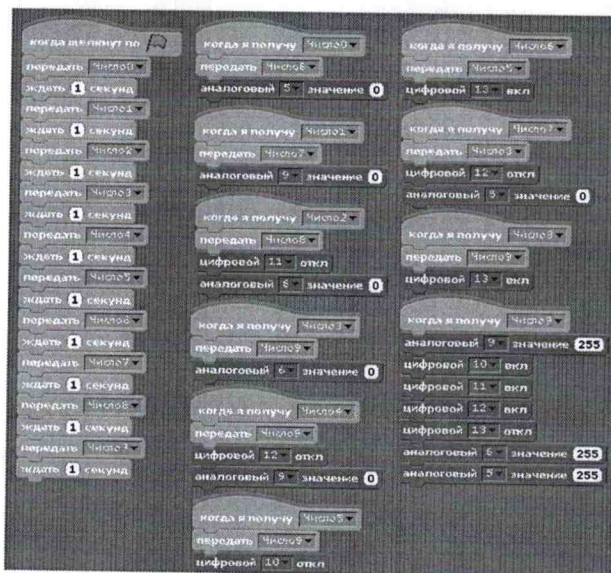


Рисунок 2.37 – Счет от 0 до 9, через предустановку девятки

Материал 15. Вывод уровня яркости на семисегментник

Совместив два предыдущих примера можно выводить яркость фоторезистора на семисегментник. Схема подключения элементов приведена на Рисунке 2.39., а пример рабочего кода на Рисунке 2.38.

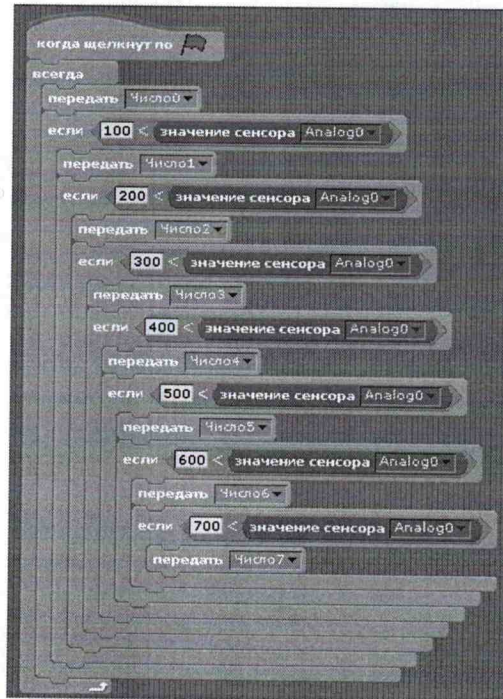


Рисунок 2.38 – Пример алгоритма вывода яркости на семисегментник

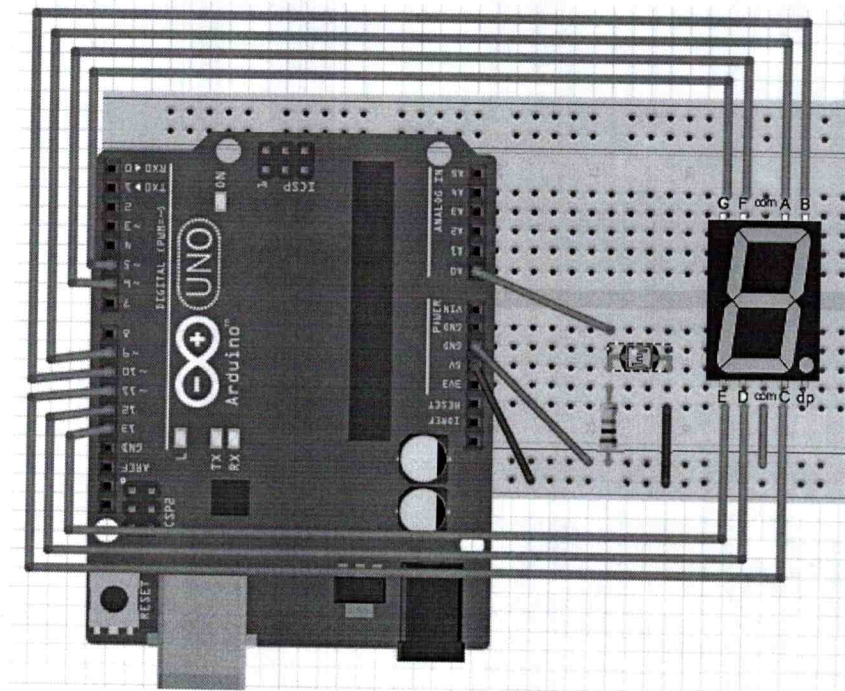


Рисунок 2.39 – Схема подключения семисегментника и фоторезистора

Материал 16. Элемент ввода джойстик

Это более сложный элемент, который в своем составе имеет несколько мелких элементов: кнопку и два потенциометра. Один потенциометр определяет положение рычажка по оси x, а второй – по оси y. Изначальное положение рычажка дают средние значения = 512 (Рисунок 2.40).

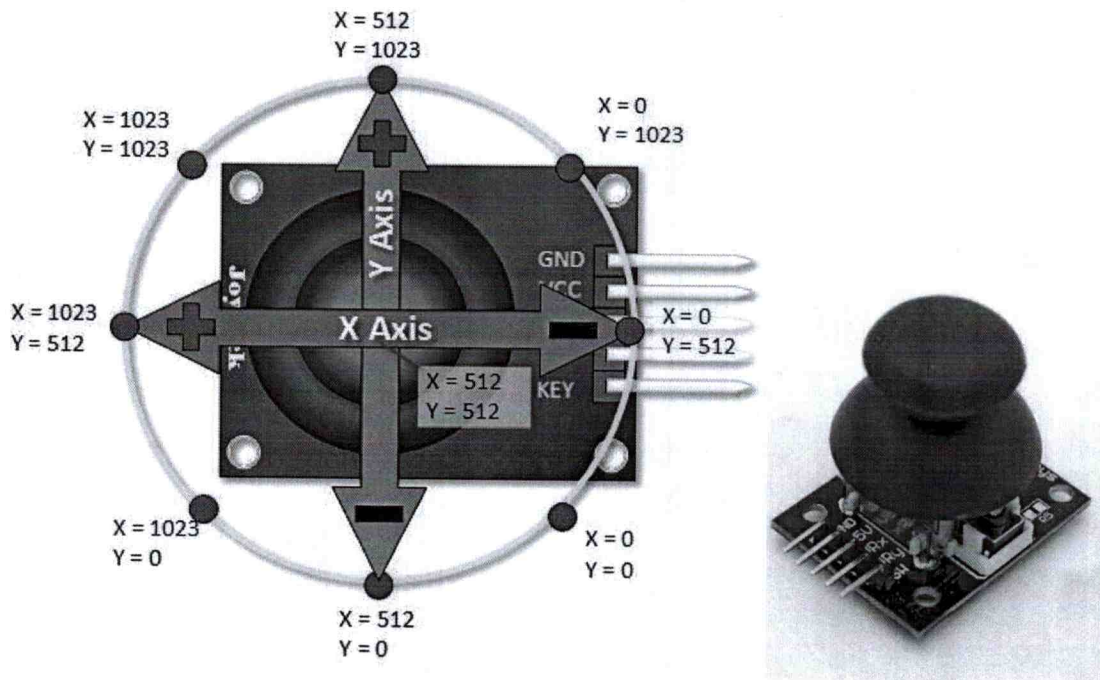


Рисунок 2.40 – Внешний вид джойстика

Джойстик имеет пять контактов подключения, один для кнопки, два для координат x и y, 5V и контакт GND (Рисунок 2.41).

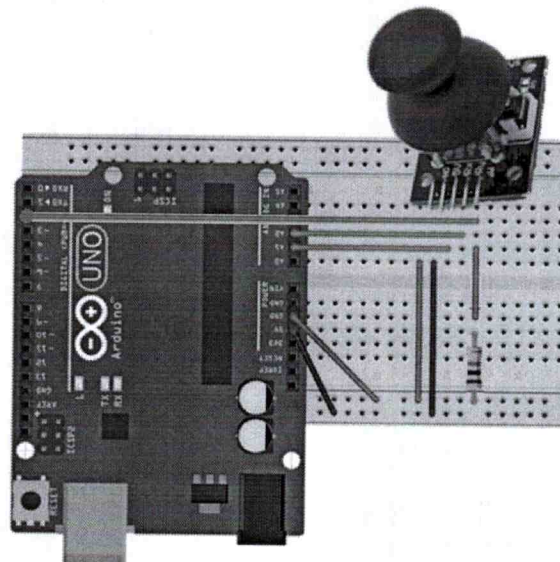





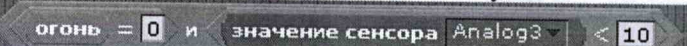


Рисунок 2.41 – Схема подключения джойстика

Создаем управление танком

Подключим джойстик, как сделано на Рисунке 2.41.

Проект будет объемным и содержать три объекта (контроллер ардуино, танк и снаряд). Начнем со скрипта контроллера ардуино. Его задачей будет получение данных с джойстика.

A1	Движение вперед (назад)	<p>Уменьшим диапазон значений в сто раз (получим значения от 0 до 10). Сдвинем диапазон на 5 (от -5 до 5).</p> <p>Это равно </p> <p>Создадим переменную вперед для передачи движения</p> <p></p>
A2	Поворот вправо (влево)	<p>Уменьшим диапазон значений в сто раз (получим значения от 0 до 10). Сдвинем диапазон на 5 (от -5 до 5).</p> <p>Это равно </p> <p>Создадим переменную повернуть для передачи поворота</p> <p></p>
A3	Огонь	<p>Будем ловить момент низкого значения.</p> <p></p> <p>Чтобы пуля стреляла один раз, создадим переменную состояния, когда она выстрелила и летит (значение =1) и когда она готова к новому полету (значение = 0). Тогда условие огня будет зависеть от двух выражений</p> <p></p>

Добавим две процедуры огонь и движение, которые будут вызываться у снаряда и танка. В целом скрипт должен выглядеть, как на Рисунке 2.42.

Добавим новые объекты танк и снаряд, их можно загрузить как картинку найдя нужную в интернете. В моем случае танк получился немного больше, чем я этого планировал, но в последствии я могу скорректировать размер через оператор установить размер в %. Внешне мои объекты стали выглядеть, как на Рисунке 2.43.

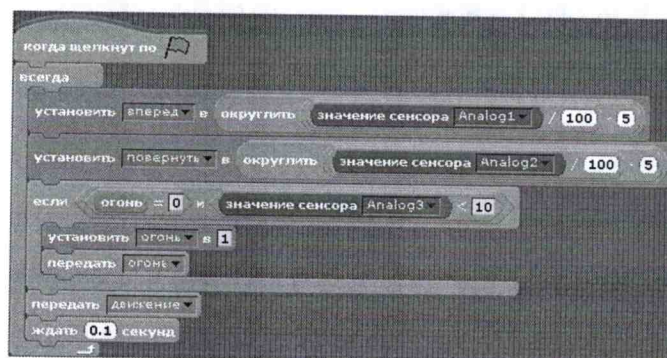


Рисунок 2.42 – Скрипт обработки сигналов, получаемых с джойстика

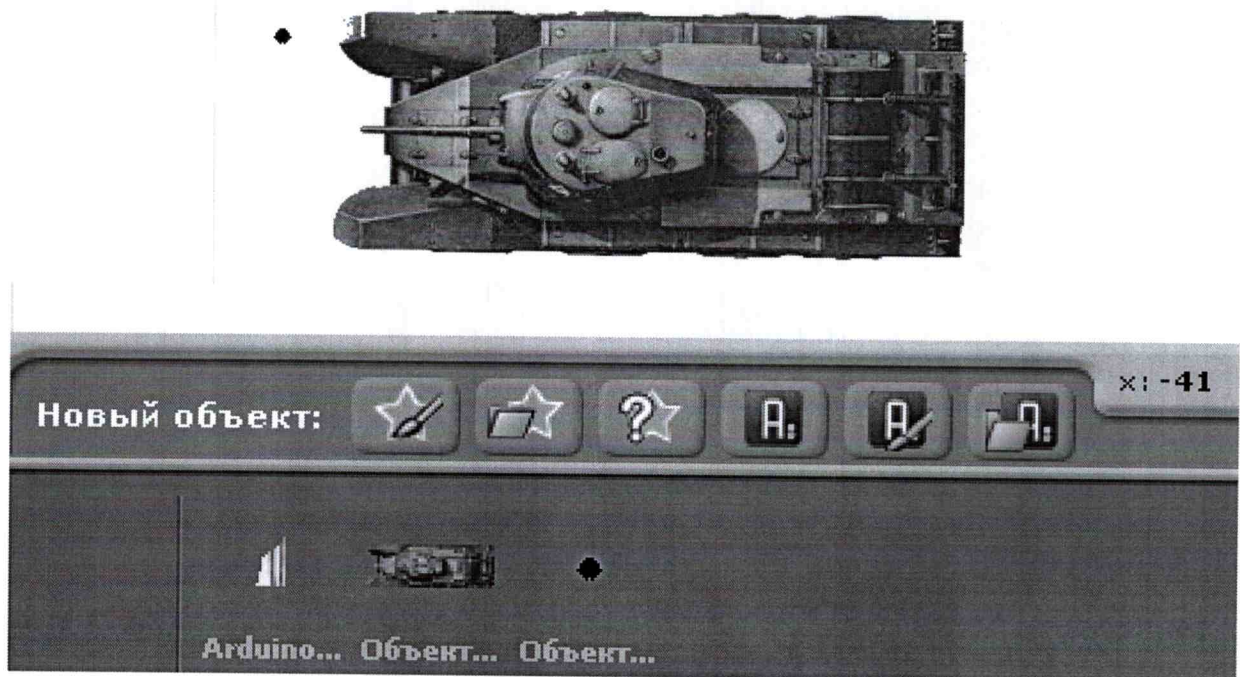


Рисунок 2.43 – Список объектов проекта

Составим скрипт для движения нашего танка. Само движение будет осуществляться обрабатывая переменные вперед и повернуть. Так как у меня на картинке танк повернут на лево, то повороты я буду совершать налево. Скорректируйте значения, для вашего танка.

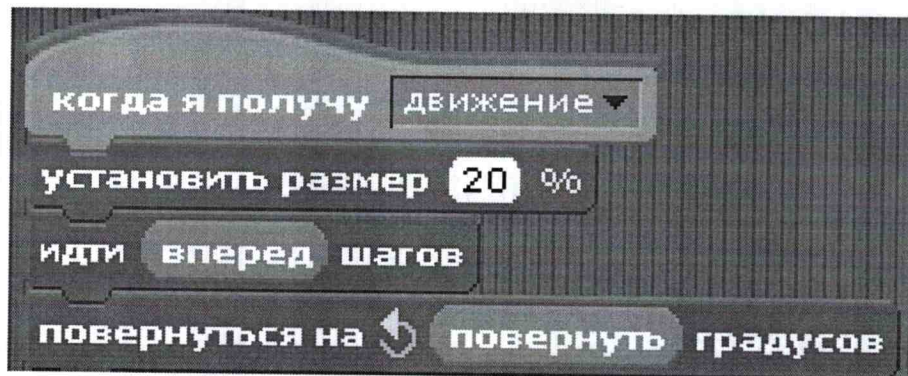


Рисунок 2.44 – Скрипт управления танком

Материал 17.

После этих двух скриптов наш танк уже управляется с пульта и может передвигаться вперед-назад, совершать повороты вправо-влево. Но выстрела до сих пор еще не делает. Добавим скрипт для снаряда, которая будет выстреливать по событию огонь (Рисунок 2.45).

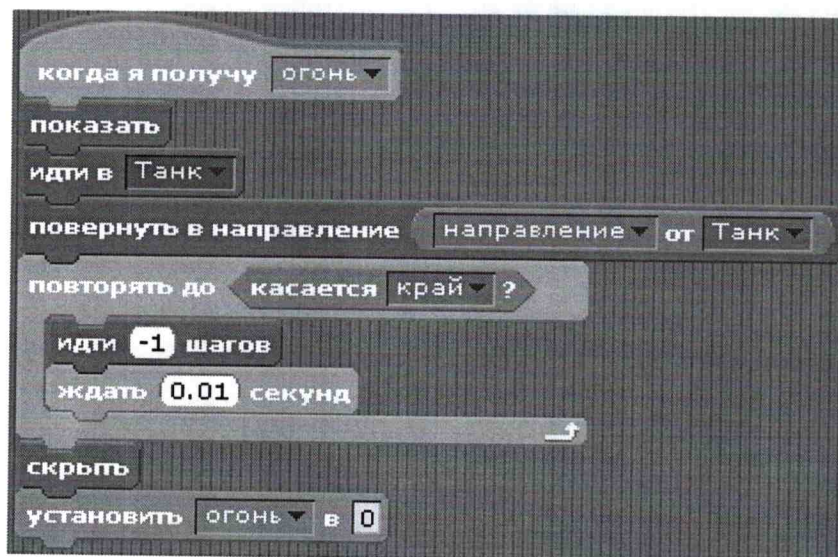


Рисунок 2.45 – Скрипт обработки движения снаряда

Так как мой танк по картинке смотрит на лево, то движение снаряда будет на шаг = (-1). Как только снаряд достигнет края, скрипт остановится и снаряд будет готов к новому выстрелу.

Теперь наш танк и передвигается и стреляет. Давайте добавим на поле препятствия. Для этого выделим сцену (Это и есть наше поле) перейдем к редактированию фона и нарисуем себе препятствия (В моем случае это выглядит как показано на Рисунке 2.46).

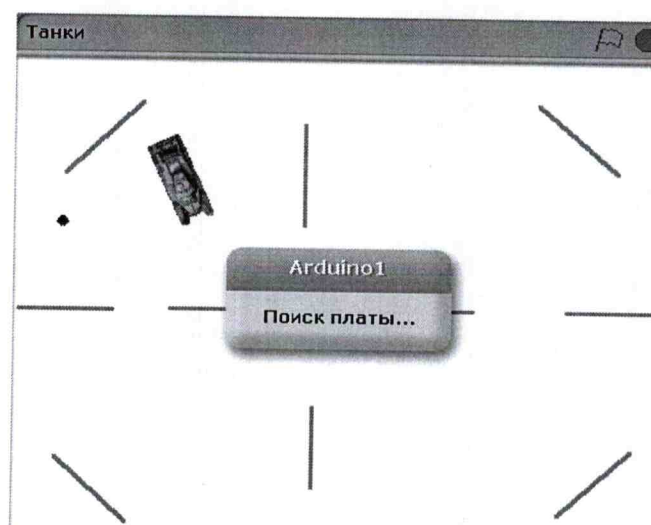


Рисунок 2.46 – Внешний вид поля

Для того, чтобы танк не мог преодолевать препятствия и снаряд прекращал движение при соприкосновении с препятствием, мы добавим в наши скрипты

для танка и для снаряда обработку соприкосновения с препятствием по цвету (Рисунок 2.47, 2.48).

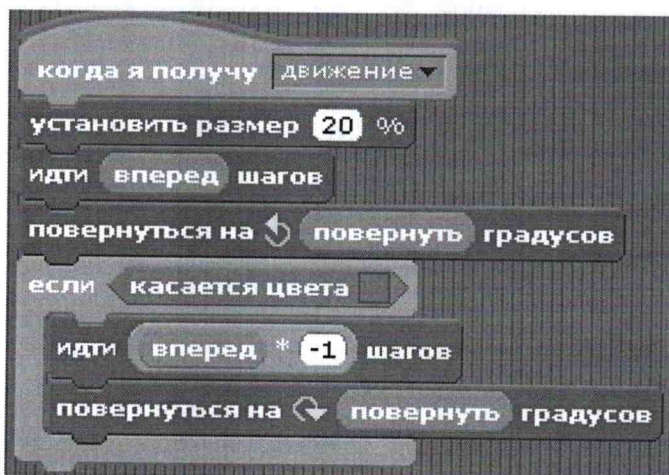


Рисунок 2.47 – Скрипт танка с обработкой препятствия

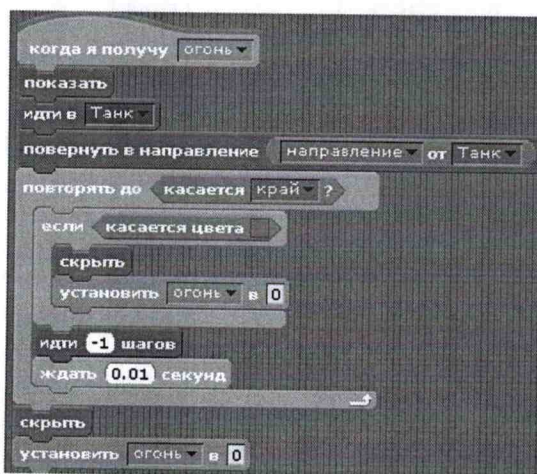


Рисунок 2.48 – Скрипт снаряда

Материал 18.

Теперь мы можем объединить усилия – подключив к компьютеру несколько джойстиков и добавить другие танки... Также потребуется доработать скрипты, чтобы при попадании снаряда врага танк считался подбитым (Рисунок 2.49).

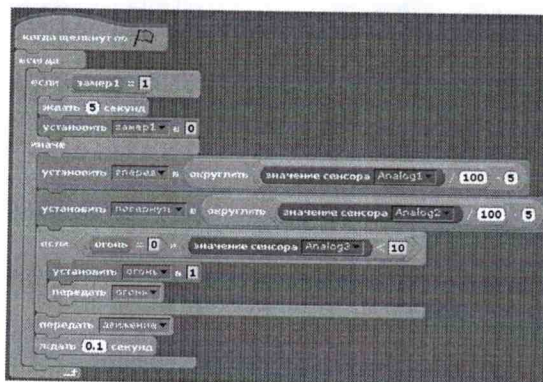


Рисунок 2.49

– Скрипт обработки сигналов, получаемых с джойстика

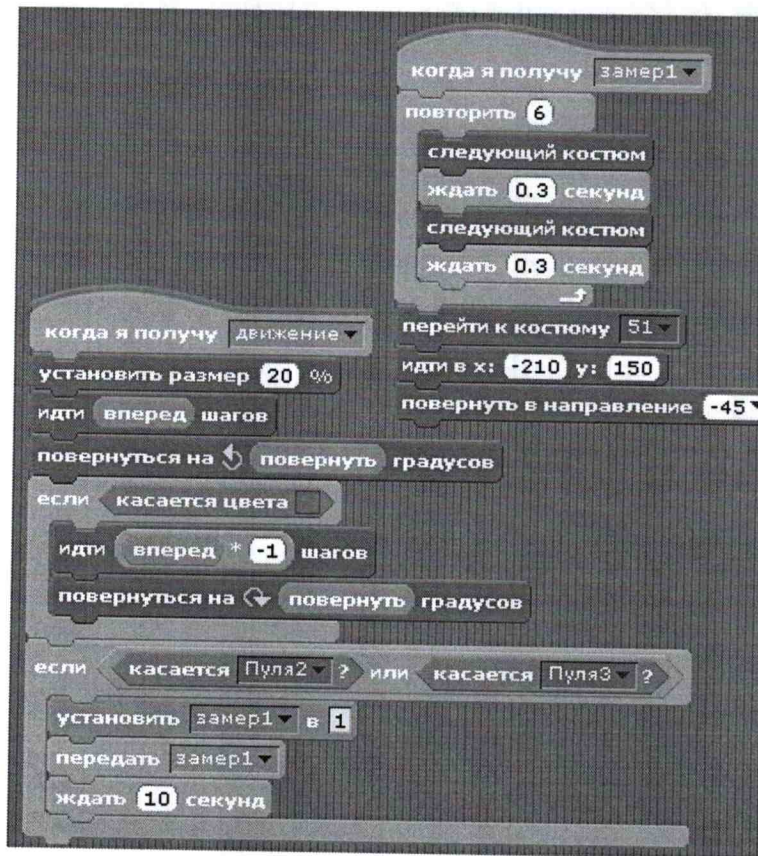


Рисунок 2.50

– Скрипт танка с двумя противниками

Материал 19. Игра "Кроты"

Смысл игры «Кроты» заключается в том, что через случайный промежуток времени (от 0.5с до 1с) загорается случайный светодиод и горит секунду. За это время нужно успеть нажать находящуюся под ним кнопку. Когда кнопка была нажата, светодиод гаснет и ведется счет попаданий и промахов [8].

Соберем схему состоящую из 4-рех светодиодов и 4-рех кнопок, как показано на Рисунке 2.51.

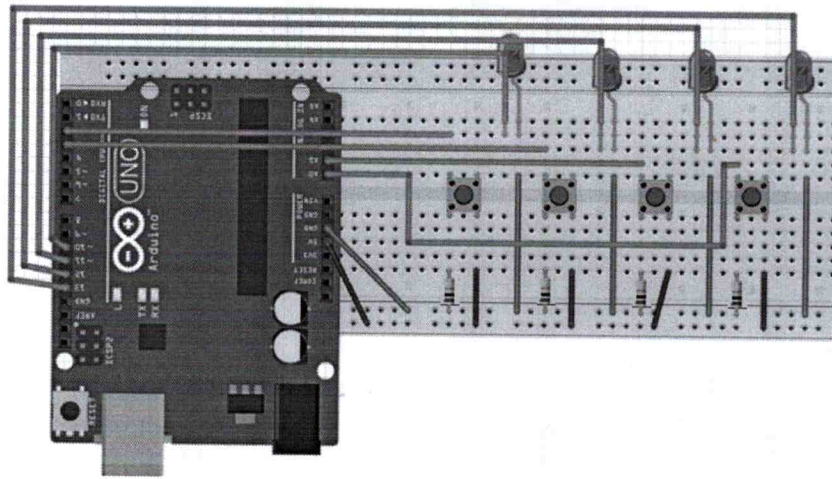


Рисунок 2.51 – Схема подключения светодиодов и кнопок
Сделаем подготовку к работе (Рисунок 2.52), для этого:

- отключим все диоды;
- сформируем случайное число равное номеру активного диода;
- вызовем процедуру с активным диодом.

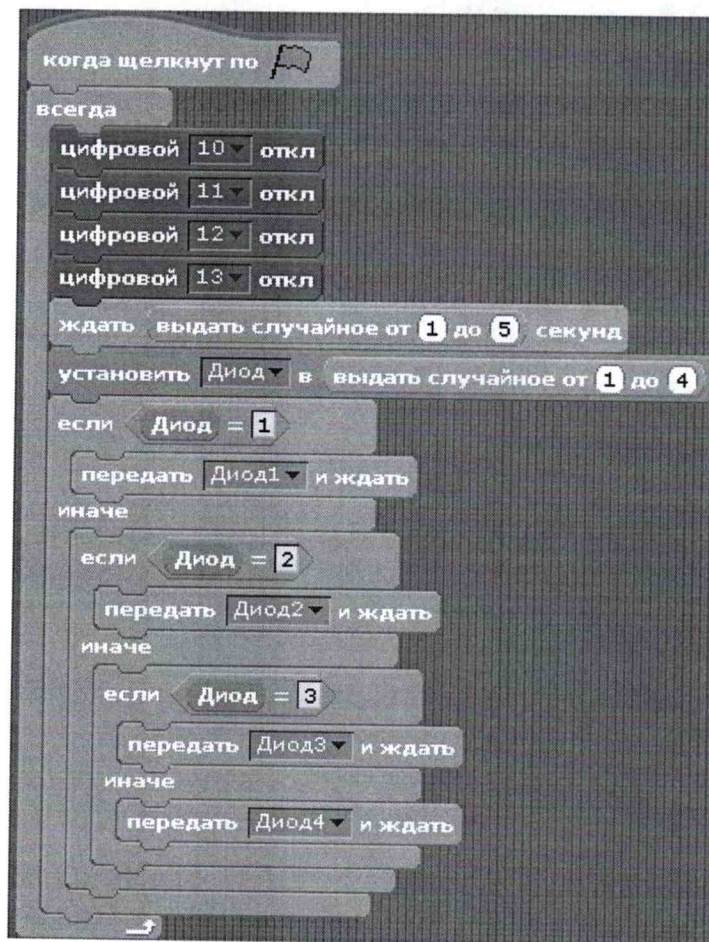


Рисунок 2.52 – Скрипт подготовки и вызова процедуры активного диода

Материал 20. Написание программного кода «Кроты»

Теперь рассмотрим, как будет выполняться скрипт (Рисунок 2.53), при вызове одного из диодов:

- подсветить нужный диод;
- ждать небольшой промежуток времени;
- проверить нажатую кнопку;
- скорректировать счет и вывести на экран.



Рисунок 2.53 – Скрипт обработки первого светодиода

Составьте скрипты для остальных трех случаев. Можно поэкспериментировать с параметрами, сделать вывод результата иначе.

Давайте попробуем усложнить игру, сделав активными не один, а случайное количество диодов. Для этого наш скрипт придется перестроить как на Рисунке 2.54.

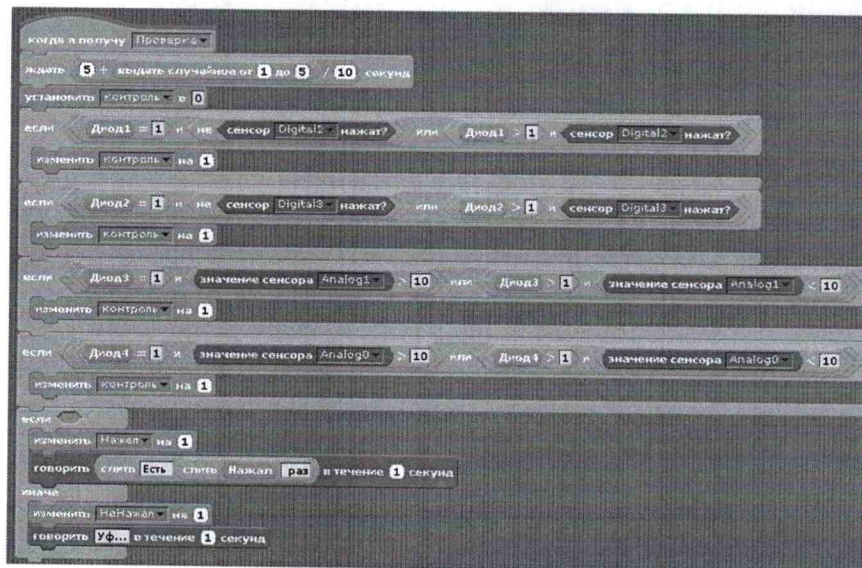


Рисунок 2.54 – Скрипт проверки в игре «Кроты»

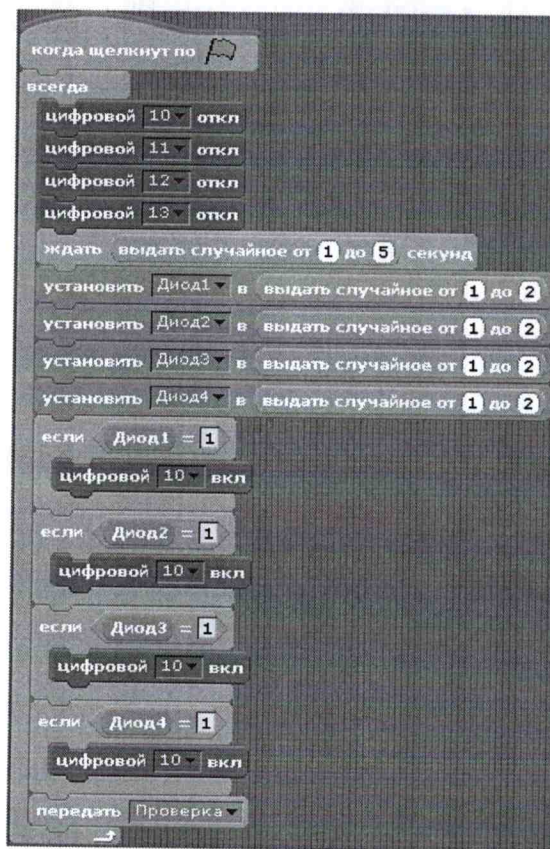


Рисунок 2.55 – Скрипт ошибка в номерах в игре «Кроты»

Материал 21. Элемент вывода, динамик

Динамик (зуммер) - акустическое устройство для воспроизведения звука, которое звучит под колебаниями сигналов. Для подключения динамику будем использовать 5 контакт.

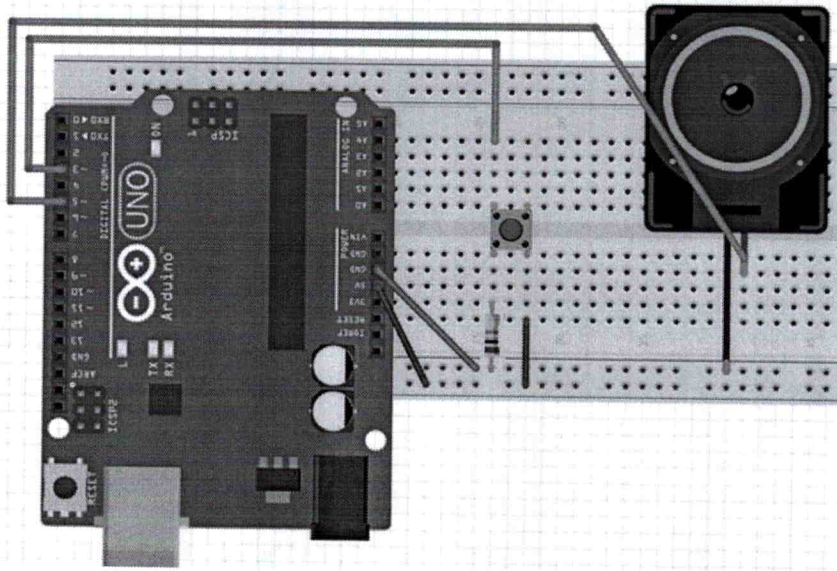


Рисунок 2.56 – Схема подключения динамика

Изучаем азбуку-Морзе

В данном уроке мы будем знакомиться с азбукой-Морзе, в которой каждая буква алфавита зашифрована последовательностью коротких и длинных сигналов.

Русский символ	Код Морзе	Русский символ	Код Морзе	Русский символ	Код Морзе
А	..	Л	Ц	----
Б	----	М	--	Ч	----.
В	...-	Н	-.	Ш	----
Г	---	О	---	Щ	----
Д	---	П	Ъ	-----
Е (Ё)	.	Р	...-	Ы	----
Ж-	С	...-	Ь	----
З	----.	Т	-	Э
И	..	У	..-	Ю	----
Й	----	Ф	Я	----
К	- . -	Х		

Список используемой литературы:

1. Голиков Д., 40 проектов на Scratch для юных программистов, БХВ-Петербург, 2016.
2. Голиков Д., 40 проектов на Scratch для юных программистов, БХВ-Петербург, 2016.
3. Голиков Д., Scratch для юных программистов, БХВ-Петербург, 2016.
4. Голиков Д., Scratch и Arduino. 18 игровых проектов для юных программистов микроконтроллеров, БХВ-Петербург, 2018.
5. Голиков Д., Дерзай! Наборы по электронике. Scratch+Arduino. 18 проектов для юных программистов КНИГА, БХВ-Петербург, 2016.
6. Джереми Блум. Изучаем Arduino: инструменты и методы технического волшебства. Издательство, БХВ-Петербург. 2015 г.
7. Комков Н.И., Бондарева Н.Н. ПЕРСПЕКТИВЫ И УСЛОВИЯ РАЗВИТИЯ РОБОТОТЕХНИКИ В РОССИИ. МИР (Модернизация. Инновации. Развитие). 2016;7(2(26)):8-21. DOI:10.18184/2079-4665.2016.7.2.8.21
8. Ревич Ю. В. Занимательная электроника. 3-е издание. Издательство: БХВ-Петербург, 2015 г. Страниц: 576.
9. РАДИО ЕЖЕГОДНИК, выпуск 24 «Путеводитель по Arduino»
10. Информационный ресурс <http://amperka.ru/>
11. Информационный ресурс <http://arduino.ru/>
12. Информационный ресурс <http://s4a.cat/>
13. Информационный ресурс www.tinkercad.com
14. Информационный ресурс <http://boteon.com/blogs/obuchayuschie-lekcii-po-arduino/uroki-po-arduino-oglavlenie.html>

Календарный учебный график на 2019 – 2020 учебный год

<i>№ темы</i>	<i>Дата (план)</i>	<i>Форма занятия</i>	<i>Тема</i>	<i>Кол-во часов</i>	<i>Форма контроля</i>
1	03.09.2019	Теоретическое	Техника безопасности при работе с ПК	1	Педагогическое наблюдение
2	10.09.2019	Теоретическое	Микроэлектроника, основные понятия, сферы применения	1	Педагогическое наблюдение, опрос
3	17.09.2019	Теоретическое	Организационные моменты. Регистрация. Установка.	1	Педагогическое наблюдение
4	24.09.2019	Теоретическое	Блочное программирование scratch. Линейный алгоритм.	1	Педагогическое наблюдение
5-6	01.10.2019 08.10.2019	Теоретическое	Плата Arduino, разновидности, аппаратная часть. Техника безопасности при работе с платой и малым током.	2	Педагогическое наблюдение, опрос
7	15.10.2019	Практическое	Цифровые и аналоговые контакты ввода-вывода, при блочном программировании в S4A (mBlock)	1	Проверка работоспособности макета.
8	22.10.2019	Практическое	Разработка устройства с <u>tinkercad Circuits</u>	1	Проверка работоспособности макета.
9-10	05.11.2019 12.11.2019	Практическое	Электронный компонент для вывода сигнала. Семисегментник.	2	Проверка работоспособности макета.
11	19.11.2019	Практическое	Электронные компоненты ввода информации кнопка и фоторезистор.	1	Проверка работоспособности макета.
12	26.11.2019	Практическое	Система Arduino, установка,	1	Проверка работоспособности

			подключение и настройка.		макета.
13.	03.12.2019	Практическое	Плавное изменение яркости диода. Аналоговые контакты 5,6,9	1	Проверка работоспособности макета.
14.	10.12.2019	Практическое	Кнопка. Цифровые контакты 2,3 Подключение кнопки.	1	Проверка работоспособности макета.
15.	17.12.2019	Практическое	Арифметические и логические операторы. Кнопочное переключение цвета.	1	Проверка работоспособности макета.
16.	24.12.2019	Практическое	Кнопочный маячок	1	Проверка работоспособности макета.
17.	14.01.2020	Практическое	Трехцветный светодиод. Радуга цветов	1	Проверка работоспособности макета.
18.	21.01.2020	Практическое	Моргание шестью диодами	1	Проверка работоспособности макета.
19.	28.01.2020	Практическое	Потенциометр. Аналоговые контакты А0-А6	1	Проверка работоспособности макета.
20.	04.02.2020	Практическое	Терменвокс	1	Проверка работоспособности макета.
21.	11.02.2020	Практическое	Защита собственного проекта, рассмотренную ранее работу со своими изменениями.	1	Защита проектов
22.	18.02.2020	Практическое	Game-Ball через потенциометр	1	Проверка работоспособности макета.
23.	25.02.2020	Практическое	Фоторезистор	1	Проверка работоспособности макета.
24.	03.03.2020	Практическое	Семисегментник, счет	1	Проверка работоспособности макета.

25.	10.03.2020	Практическое	Семисегментник, процедурное программирование Вывод уровня яркости на семисегментник	1	Проверка работоспособности макета.
26.	17.03.2020	Практическое	Джойстик, управление танком	1	Проверка работоспособности макета.
27.	31.03.2020	Практическое	Джойстик, разработка карты и стрельба	1	Проверка работоспособности макета.
28.	07.04.2020	Практическое	Подключаем несколько контроллеров, танковые бои	1	Проверка работоспособности макета.
29.	14.04.2020	Практическое	Динамик	1	Проверка работоспособности макета.
30.	21.04.2020	Практическое	Сервопривод Подключение сервопривода	1	Проверка работоспособности макета.
31.	28.04.2020	Практическое	Кнопочные ковбои	1	Проверка работоспособности макета.
32.	05.05.2020	Практическое	Подсчет нажатий на кнопку	1	Проверка работоспособности макета.
33.	12.05.2020	Практическое	Защита собственного проекта, рассмотренную ранее работу со своими изменениями	1	Защита проектов
34.	19.05.2020		Резерв	1	