Покажем организацию внеурочной образовательной деятельности на примере занятия с применением проектной технологии обучения. Занятие будет предусматривать применение облачных технологий. Ниже будет представлен конспект урока.

Предмет: факультатив по информатике и ИКТ.

Класс: 9.

Тема урока: «Игра на реакцию».

Тип урока: урок открытия нового знания.

Цель урока: систематизировать полученные знания о языке программирования C++, применив их в разработке проекта на платформе Arduino.

Задачи урока:

* создать макет проекта, расположив электроэлементы на макетной плате и подключив их к Arduino Uno;
* написать первичный код для проверки работы схемы;
* написать код для работы проекта.

Планируемые результаты урока:

* предметные: систематизация умения писать программный код на языке программирования C++;
* метапредметные: формирование основ ИКТ-компетентности;
* личностные: понимание значения элементов программирования и моделирования для жизни человека и человечества; интерес к изучению информатики и робототехники.

Основные понятия: микроконтроллер, язык программирования, макет проекта, программный код, синтаксис C++, функции для работы с Arduino.

Оборудование: ПК учителя и обучающегося, подключенные к сети Интернет, мультимедийный проектор, смартфоны обучающихся, подключенные к сети Интернет, электронные образовательные ресурсы (облачные сервисы): PowerPoint, Kahoot, Tinkercad, OneDrive.

Ход урока:

Занятие начинается с того, что учитель объявляет тему проекта. После чего обучающиеся могут сформулировать цель проекта и задачи для достижения поставленной цели. Учитель показывает видео-ролик с подобным проектом, выложенным на сервисе YouTube по ссылке (<https://youtu.be/WNudyDRDpR8>).

В оригинале игра базируется лишь на светодиодах одного цвета. В связи с чем учитель предлагает обучающимся высказать свои идеи по поводу того, как сделать данную игру интереснее для использования. Обучающиеся предлагают сделать светодиод, который необходимо поймать в палитре RGB. Это приведет к усложнению игры с появлением дополнительных кнопок, которых в игре будет уже три (красная, зеленая и синяя), а не одна.

Далее учитель переходит к обсуждению первичного представления обучающихся о коде этой программы. Учитель на данном этапе занятия может давать некоторые подсказки ученикам, например, при программировании алгоритма последовательного переключения светодиодов мы будем использовать изменения состояния переменной.

Также здесь учителю важно подвести к размышлению о том, как правильно подключить светодиоды к пинам платы. Обучающиеся могут догадаться, что светодиоды должны быть соединены с пинами платы в порядке их возрастания (ко второму пину подключается первый светодиод, к третьему – второй и т.д.). Данный прием поможет упростить разработку кода, позволяя изменять переменную на единицу, сбрасывая ее значение на начальное в конце прохождения круга. Следует обратить внимание обучающихся на тот факт, что при составлении кода мы не должны использовать функцию delay (), которая будет тормозить работу основного цикла void loop (). Во избежание этой проблемы мы будем пользоваться функцией millis (), подсчитывающей количество пройденного времени с момента старта микроконтроллера в миллисекундах, а также условным оператором if ().

После осуществления учителем вводного этапа и этапа знакомства обучающихся с теоретической составляющей разрабатываемого проекта, класс переходит к созданию рабочей схемы на макетной плате (рис. 10).

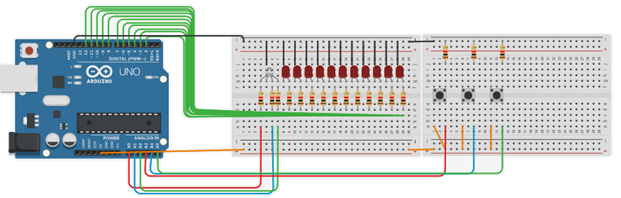


Рис. 10. Модель проекта, созданная на макетной плате

После сборки модели проекта учитель акцентирует внимание на том, что необходимо проверить работоспособность всех подключенных электроэлементов в нашей схеме. При этом педагог отмечает, что стоит написать отдельную программу от основного кода, в котором будет производиться эта проверка. Написание тестового и основного кода будет происходить на сервисе Tinkercad [12].

После прохождения тестирования функциональности схемы обучающиеся переходят к написанию основного кода программы. Здесь учитель акцентирует внимание на важность написания пояснительных комментариев к коду, а также макросов, в названии которых будет отражаться назначение пинов (рис. 11).

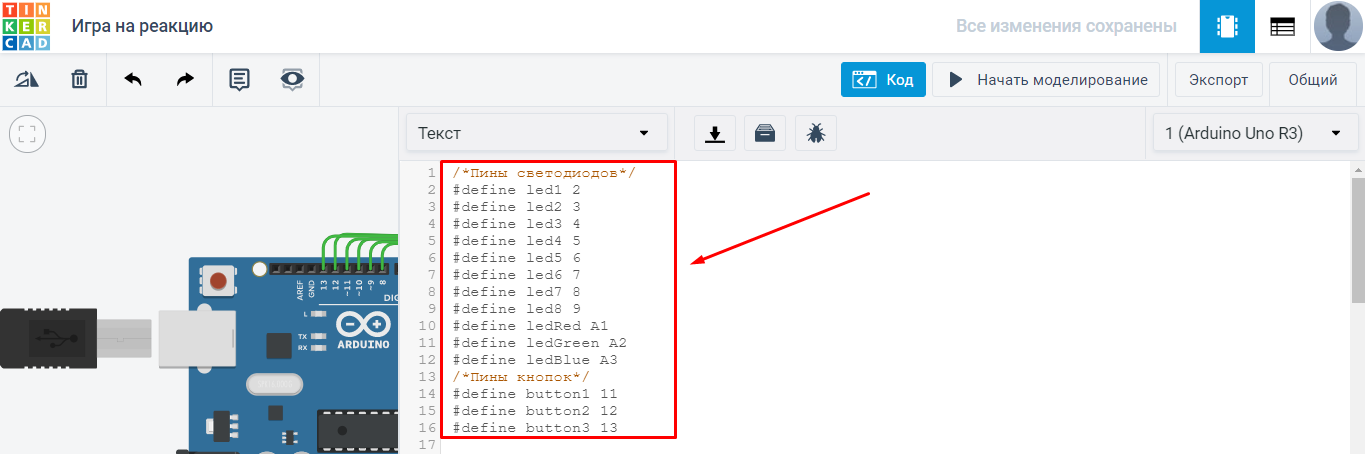


Рис. 11. Написание первичного кода в облачном сервисе Tinkercad

Далее ход урока переходит на этап самостоятельной работы обучающихся, где каждому ученику требуется попробовать прописать настройки всех используемых пинов в проекте. Учитель может дать подсказку: сокращения кода может происходить с использованием цикла for ().

После выполнения задания учитель может проверить на платформе Tinkercad правильность кода. Каждый обучающийся состоит в заранее созданном учителем виртуальном классе, где отображаются все существующие проекты обучающихся в режиме реального времени. Педагог может корректировать работу абсолютно каждого ребенка, оставляя комментарии непосредственно в его индивидуально созданном коде. В итоге, все обучающиеся должны прийти к одинаковому результату:

void setup() {

//Настройка пинов на выход

for (int svetodiot = 2; svetodiot <= 16; svetodiot++)

{

pinMode(svetodiot, OUTPUT);

}

}

Следующим этапом создания проекта будет написание кода, способного включать последовательно светодиоды без задержки основного цикла. Для чего будет использована функция millis () и условный оператор if ().

Алгоритм реализации будет основан на подсчете разницы во времени с функцией millis () и времени, сохраненного в отдельной переменной. С целью демонстрации этого алгоритма учитель прописывает код программы самостоятельно и выводит его на интерактивной доске:

int times = 70; //Время задержки

void loop() {

if (millis() - times > Second1)

{

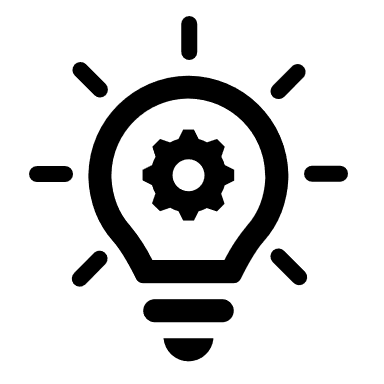
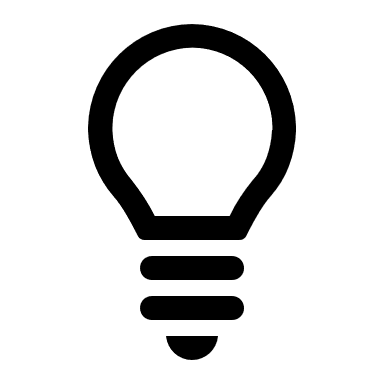
//Выполняется 1 раз в Second1 мс

Second1 = millis();

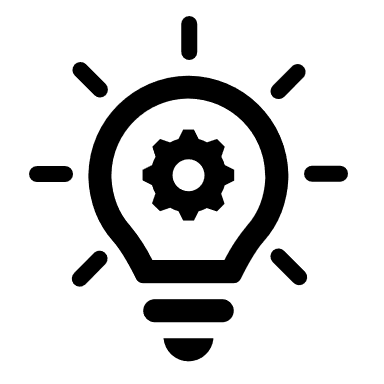
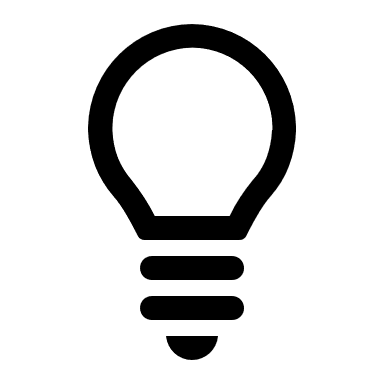
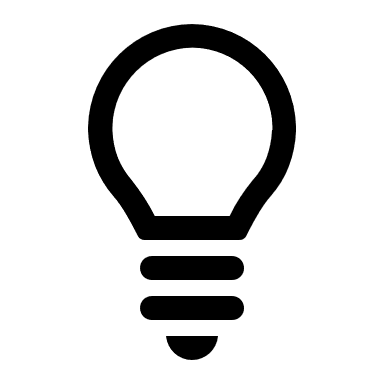
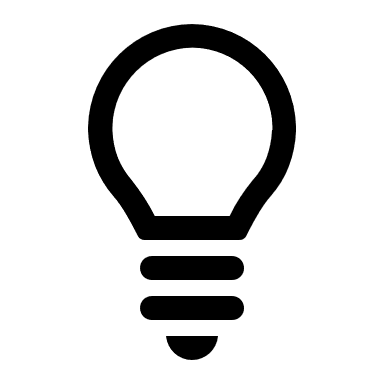
}

}

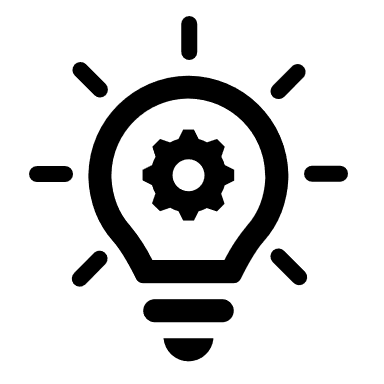
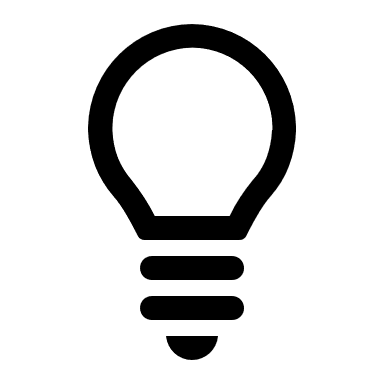
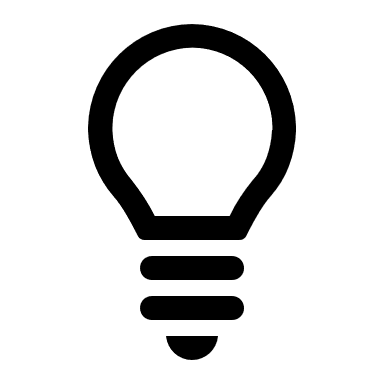
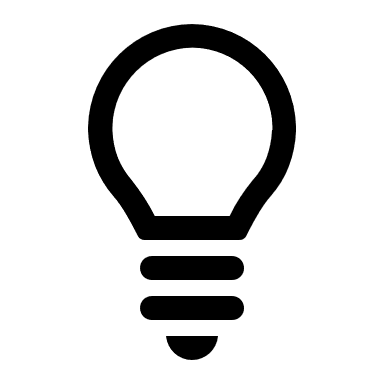
После демонстрации кода на доске, обучающиеся тестируют его на собранной схеме. Далее учителем объявляется новое задание, в котором обучающиеся должны дополнить код введением переменной, отвечающей за номер включенного светодиода, совершая последовательное включение последующего и выключение предыдущего (рис. 12).



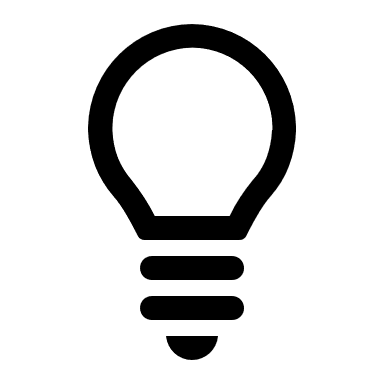
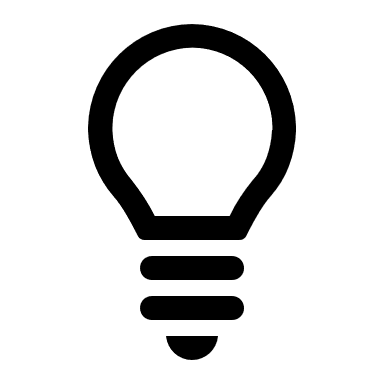
**1 итерация**



**2 итерация**



**3 итерация**



**4 итерация**

**…**

**…**

**…**

**…**

**…**

**…**

**…**

**…**

Рис. 12. Алгоритм последовательного включения светодиодов

Так как сервис Tinkercad позволяет производить разработку программы на основе облачных технологий, то обучающиеся могут продолжить написание кода в домашних условиях.

По окончании занятия учитель предлагает задание: самостоятельно доработать код проекта для загорания случайным цветом RGB светодиода, проверку задания учителем сможет осуществить дистанционно, воспользовавшись виртуальным классом в этом же облачном сервисе.