

Создание аналога игры «Touch me» от Atari на базе платы Arduino

Кизянов Антон Олегович

Приамурский государственный университет имени Шолом-Алейхема

Студент

Аннотация

В данной статье описан процесс построения аналога старой игры “Touch Me” от Atari на базе платформы Arduino. Для создания понадобится сама плата, несколько кнопок, светодиодов и резисторов. Созданный проект имитирует игру “Touch Me” и позволяет дать понятие о разработке примитивных игр на платформе Arduino.

Ключевые слова: Arduino, Игра

Creating an analogue of the Atari Touch me game based on the Arduino board

Kizyanov Anton Olegovich

Sholom-Aleichem Priamursky State University

student

Abstract

This article describes the process of building an analogue of the old Atari “Touch Me” game based on the Arduino platform. To create, you need the board itself, several buttons, LEDs and resistors. The created project imitates the game “Touch Me” and allows you to give an idea about the development of primitive games on the Arduino platform.

Keywords: Arduino, Game

Оригинальная игра Atari имела четыре цветные панели, каждая со светодиодом, который горел в определенном порядке, игроки должны были каждый раз повторять нарастающую последовательность. Как выглядела такая игра можно увидеть на рисунке 1.

Игра начинается с проигрывания мелодии и загорания светодиода определенного цвета. Когда игрок нажимает правильную соответствующую кнопку, индикаторы снова мигают в более длинной последовательности. Каждый раз, когда последовательность введена правильно, игра добавляет дополнительный шаг, чтобы сделать последовательность более сложной. Когда игрок делает ошибку, игра переходит в начало.



Рис. 1 игра “Touch Me” от Atari

Цель исследования – создание аналога игры «Touch me» от Atari на базе платы Arduino.

Ранее этим вопросом интересовались С.Н. Шайланов, А.И. Насырова развивали тему «Разработка детской развивающей игры "проведи линию" на arduino nano» [1] в которой приводится описание развивающей электрифицированной игры «Проведи линию». Приводится порядок разработки, изготовления и налаживания. В конструкции игры применена платформа Arduino Nano. П.Е. Некрасов с темой «Игра "перетягивание каната" на основе arduino uno» [2], а подробнее про это концепцию, основанную на применении игровых механик, методов, принципов и приемов к неигровым видам деятельности, таким, к примеру, как подбор персонала, образование или пропаганда здорового образа жизни. Например, с помощью моей игры показать учащимся 5-7 классов как интересна физика в совокупности с информатикой. Ю.А. Чиркин опубликовал статью «Использование микроконтроллера arduino при реализации игровой технологии в условиях введения ФГОС ООО» [3] рассказал про программу контролирующую работу 2 кнопок команд и кнопки ведущего. Если сигнал с кнопки ведущего не поступил, то нажатие на кнопку команды вызовет звуковой сигнал «фальстарт», при поступлении сигнала с кнопки ведущего нажатие на кнопку команды приводит к появлению на экране графической индикации о готовности команды к ответу и звукового сигнала. Кнопка второй команды при этом блокируется, позволяя однозначно выявить лидера в соревновании.

Для создания аналога игры “Touch me” потребуется:

- Arduino

- Макетная плата
- Соединительные провода
- Пьезо элемент
- 4 кнопки с 4 пинами
- 4 светодиода
- 4 резистора на 220 ом.

Сначала нужно разместить кнопки на макетной плате согласно схеме на рисунке 2.

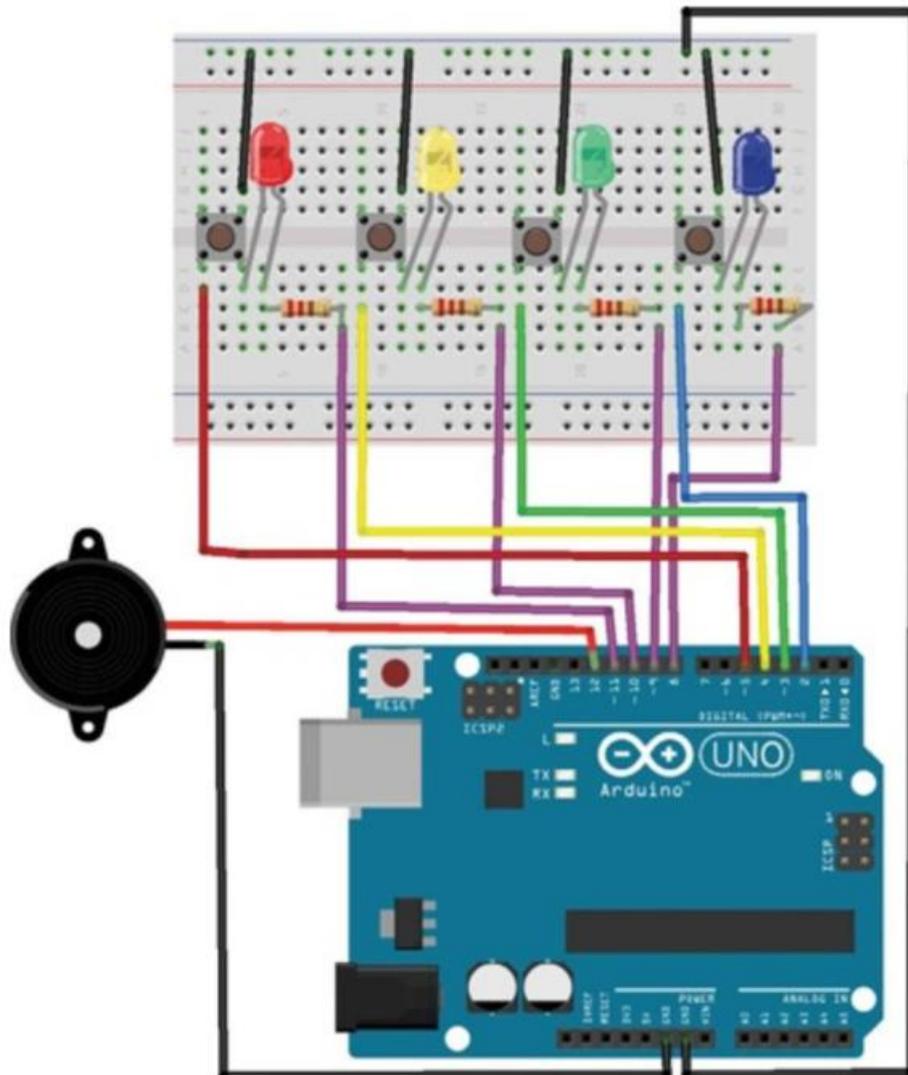


Рис. 2 Схема подключения

Стандартное схема кнопки с 4 пинами изображена на рисунке 3.

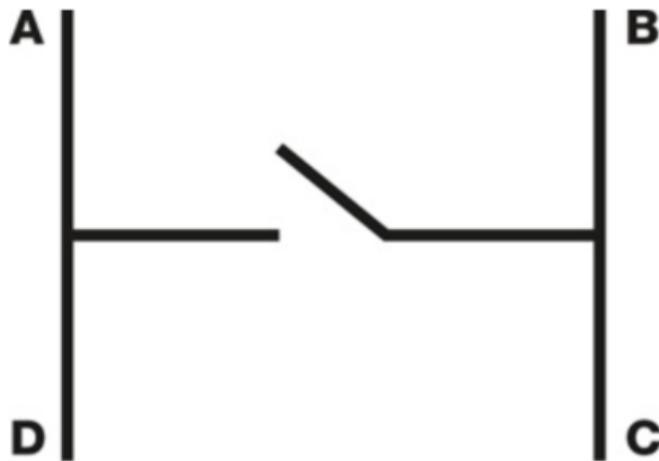


Рис. 3 Схема кнопки на 4 пина

Выход В нужно подключить к направляющей GND, а её подключить к GND Arduino.

Выход D каждой кнопки нужно соединить с цифровыми выходами Arduino со 2 по 5 по порядку.

Светодиоды отрицательным контактом соединяются с С каждой кнопки. Каждый положительный контакт нужно соединить с резистором на 220 ом, а другой контакт резистора нужно соединить с Arduino.

Нужно соблюдать аккуратность чтобы красный светодиод, подключенный к контакту 11, соединен с кнопкой, подключенной к контакту 5, желтый светодиод, подключенный к контакту 10, соединен с кнопкой, подключенной к контакту 4, зеленый светодиод, подключенный к контакту 9, соединен с кнопкой, подключенной к контакт 3, и синий светодиод, подключенный к контакту 8, соединен с кнопкой, подключенной к контакту 2.

Черный провод от пьезо элемента нужно соединить с GNU Arduino, а красный - с контактом 12 на Arduino.

```
#include <Tone.h>
Tone SpeakPin;
int stune[] = {NOTE_C4, NOTE_F4, NOTE_C4, NOTE_F4, NOTE_C4,
              NOTE_F4, NOTE_C4, NOTE_F4, NOTE_G4, NOTE_F4,
              NOTE_E4, NOTE_F4, NOTE_G4
              };
int dur2[] = {100, 200, 100, 200, 100, 400, 100, 100, 100, 100,
             200, 100, 500
             };
int note[] = {NOTE_C4, NOTE_C4, NOTE_G4, NOTE_C5, NOTE_G4,
             NOTE_C5};
int dur[] = {100, 100, 100, 300, 100, 300};
boolean btn[] = {2, 3, 4, 5};
boolean LPin[] = {8, 9, 10, 11};
```

```
int tur = 0;
int btnstatus = 0;
int ranArray[100];
int inpArray[100];
void setup() {
  Serial.begin(9600);
  SpeakPin.begin(12);
  for (int x = 0; x < 4; x++) {
    pinMode(LPin[x], OUTPUT);
  } for (int x = 0; x < 4; x++) {
    pinMode(btn[x], INPUT);
    digitalWrite(btn[x], HIGH);
  }

  randomSeed(analogRead(0));
  for (int thisNote = 0; thisNote < 13; thisNote++) {
    SpeakPin.play(stune[thisNote]); // Play the next note
    if (thisNote == 0 || thisNote == 2 || thisNote == 4 ||
        thisNote == 6) { // Hold the note
      digitalWrite(LPin[0], HIGH);
    } if (thisNote == 1 || thisNote == 3 || thisNote == 5 ||
        thisNote == 7 || thisNote == 9 || thisNote == 11) {
      digitalWrite(LPin[1], HIGH);
    } if (thisNote == 8 || thisNote == 12) {
      digitalWrite(LPin[2], HIGH);
    } if (thisNote == 10) {
      digitalWrite(LPin[3], HIGH);
    } delay(dur2[thisNote]);
    SpeakPin.stop(); // Stop for the next note
    digitalWrite(LPin[0], LOW);
    digitalWrite(LPin[1], LOW);
    digitalWrite(LPin[2], LOW);
    digitalWrite(LPin[3], LOW);
    delay(25);
  } delay(1000);
}

void loop() {
  for (int y = 0; y <= 99; y++) {
    digitalWrite(LPin[0], HIGH);
    digitalWrite(LPin[1], HIGH);
    digitalWrite(LPin[2], HIGH);
    digitalWrite(LPin[3], HIGH);
    for (int thisNote = 0; thisNote < 6; thisNote++) {
      SpeakPin.play(note[thisNote]);
      delay(dur[thisNote]);
    }
  }
}
```

```
    SpeakPin.stop();
    delay(25);
} digitalWrite(LPin[0], LOW);
digitalWrite(LPin[1], LOW);
digitalWrite(LPin[2], LOW);
digitalWrite(LPin[3], LOW);
delay(1000);
for (int y = tur; y <= tur; y++) {
    Serial.println("");
    Serial.print("tur: ");
    Serial.print(y);
    Serial.println("");
    ranArray[y] = random(1, 5);
    for (int x = 0; x <= tur; x++) {
        Serial.print(ranArray[x]);
        for (int y = 0; y < 4; y++) {
            if (ranArray[x] == 1 && LPin[y] == 8) {
                digitalWrite(LPin[y], HIGH);
                SpeakPin.play(NOTE_G3, 100);
                delay(400);
                digitalWrite(LPin[y], LOW);
                delay(100);
            } if (ranArray[x] == 2 && LPin[y] == 9) {
                digitalWrite(LPin[y], HIGH);
                SpeakPin.play(NOTE_A3, 100);
                delay(400); digitalWrite(LPin[y], LOW);
                delay(100);
            } if (ranArray[x] == 3 && LPin[y] == 10) {
                digitalWrite(LPin[y], HIGH);
                SpeakPin.play(NOTE_B3, 100);
                delay(400);
                digitalWrite(LPin[y], LOW);
                delay(100);
            } if (ranArray[x] == 4 && LPin[y] == 11) {
                digitalWrite(LPin[y], HIGH);
                SpeakPin.play(NOTE_C4, 100);
                delay(400);
                digitalWrite(LPin[y], LOW);
                delay(100);
            }
        }
    }
} input();
}
```

```
void input() {
  for (int x = 0; x <= tur;) {
    for (int y = 0; y < 4; y++) {
      btnstatus = digitalRead(btn[y]);
      if (btnstatus == LOW && btn[y] == 2) {
        digitalWrite(LPin[0], HIGH);
        SpeakPin.play(NOTE_G3, 100);
        delay(200);
        digitalWrite(LPin[0], LOW);
        inpArray[x] = 1;
        delay(250);
        Serial.print(" ");
        Serial.print(1);
        if (inpArray[x] != ranArray[x]) {
          fail();
        } x++;
      } if (btnstatus == LOW && btn[y] == 3) {
        digitalWrite(LPin[1], HIGH);
        SpeakPin.play(NOTE_A3, 100);
        delay(200);
        digitalWrite(LPin[1], LOW);
        inpArray[x] = 2;
        delay(250);
        Serial.print(" "); Serial.print(2);
        if (inpArray[x] != ranArray[x]) {
          fail();
        } x++;
      }
      if (btnstatus == LOW && btn[y] == 4) {
        digitalWrite(LPin[2], HIGH);
        SpeakPin.play(NOTE_B3, 100);
        delay(200);
        digitalWrite(LPin[2], LOW);
        inpArray[x] = 3;
        delay(250);
        Serial.print(" ");
        Serial.print(3);
        if (inpArray[x] != ranArray[x]) {
          fail();
        } x++;
      } if (btnstatus == LOW && btn[y] == 5) {
        digitalWrite(LPin[3], HIGH);
        SpeakPin.play(NOTE_C4, 100);
        delay(200);
        digitalWrite(LPin[3], LOW);
```

```
    inpArray[x] = 4;
    delay(250);
    Serial.print(" ");
    Serial.print(4);
    if (inpArray[x] != ranArray[x]) {
        fail();
    } x++;
}
}
} delay(500);
tur++;
}
void fail() {
    for (int y = 0; y <= 2; y++) {
        digitalWrite(LPin[0], HIGH);
        digitalWrite(LPin[1], HIGH);
        digitalWrite(LPin[2], HIGH);
        digitalWrite(LPin[3], HIGH);
        SpeakPin.play(NOTE_G3, 300);
        delay(200);
        digitalWrite(LPin[0], LOW);
        digitalWrite(LPin[1], LOW);
        digitalWrite(LPin[2], LOW); digitalWrite(LPin[3], LOW);
        SpeakPin.play(NOTE_C3, 300);
        delay(200);
    } delay(500);
    tur = -1;
}
```

Эскиз генерирует случайную последовательность, в которой светятся светодиоды; случайное значение, сгенерированное для y в контуре шаблона, определяет, какой светодиод горит (например, если y равен 2, загорится светодиод, подключенный к контакту 2). Нужно повторять образец, чтобы перейти на следующий уровень.

На каждом уровне предыдущие уровни повторяются, и к шаблону добавляется еще один случайно сгенерированный шаблон. Каждый светодиод связан с разным тоном пьезо элемента, поэтому каждый раз будут разные мелодии. Если ошибиться в последовательности, игра перезапустится с другой случайной последовательностью.

Результат работы можно увидеть на рисунке 4.

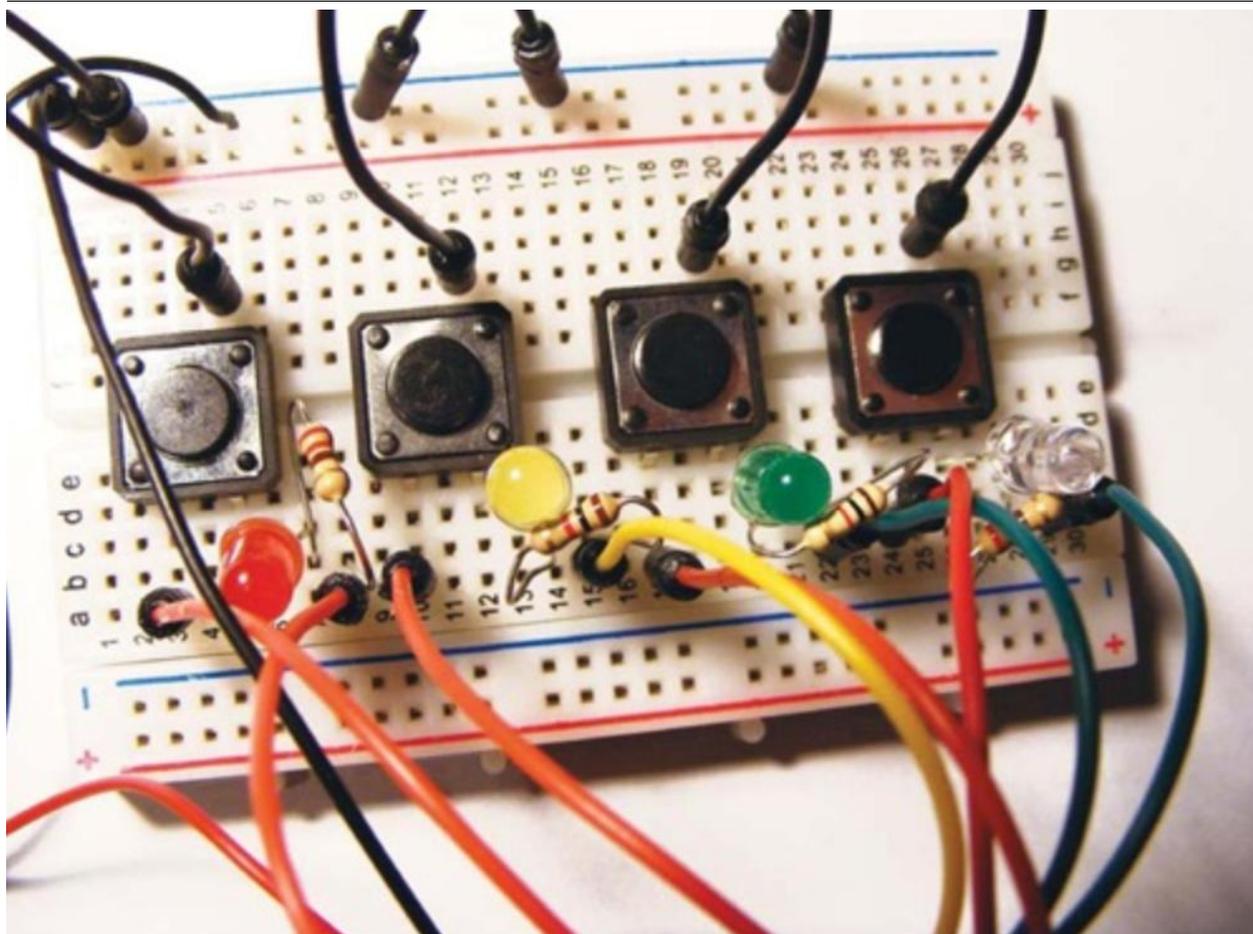


Рис. 4 Схема в собранном состоянии

Вывод

Результатом статьи стал работающий аналог старой игры “Touch Me” от Atari. Схема получилось довольно легкой, но сам скетч громоздким и трудным. Такая схема может помочь в понимании работы построения примитивных игр на микроконтроллерах.

Библиографический список

1. Шайланов С.Н., Насырова А.И. Разработка детской развивающей игры "проведи линию" на arduino nano // В сборнике: Актуальные вопросы преподавания технических дисциплин материалы Всероссийской заочной научно-практической конференции. 2017. С. 129-133. URL: <https://elibrary.ru/item.asp?id=29977693> (Дата обращения: 08.08.2019)
2. Некрасов П.Е. Игра "перетягивание каната" на основе arduino uno // В сборнике: РоботоБУМ - Будущее Умных Машин Сборник материалов участников Всероссийской научно-практической конференции обучающихся. 2017. С. 52-61. URL: <https://elibrary.ru/item.asp?id=35036141> (Дата обращения: 08.08.2019)
3. Чиркин Ю.А. Использование микроконтроллера arduino при реализации игровой технологии в условиях введения ФГОС ООО // В сборнике: Информатика: проблемы, методология, технологии Материалы XVI

Международной научно-методической конференции. Под редакцией
Крыловецкого А.А.. 2016. С. 822-824. URL:
<https://elibrary.ru/item.asp?id=26611915> (Дата обращения: 08.08.2019)