

*SSS (Smart Security System)*

*Arduino проект на тема*

***Изготвил:***

Георги Тодоров Колев,

Пловдив, 2023

* Математическа гимназия

“Академик Кирил Попов”

**Съдържание**

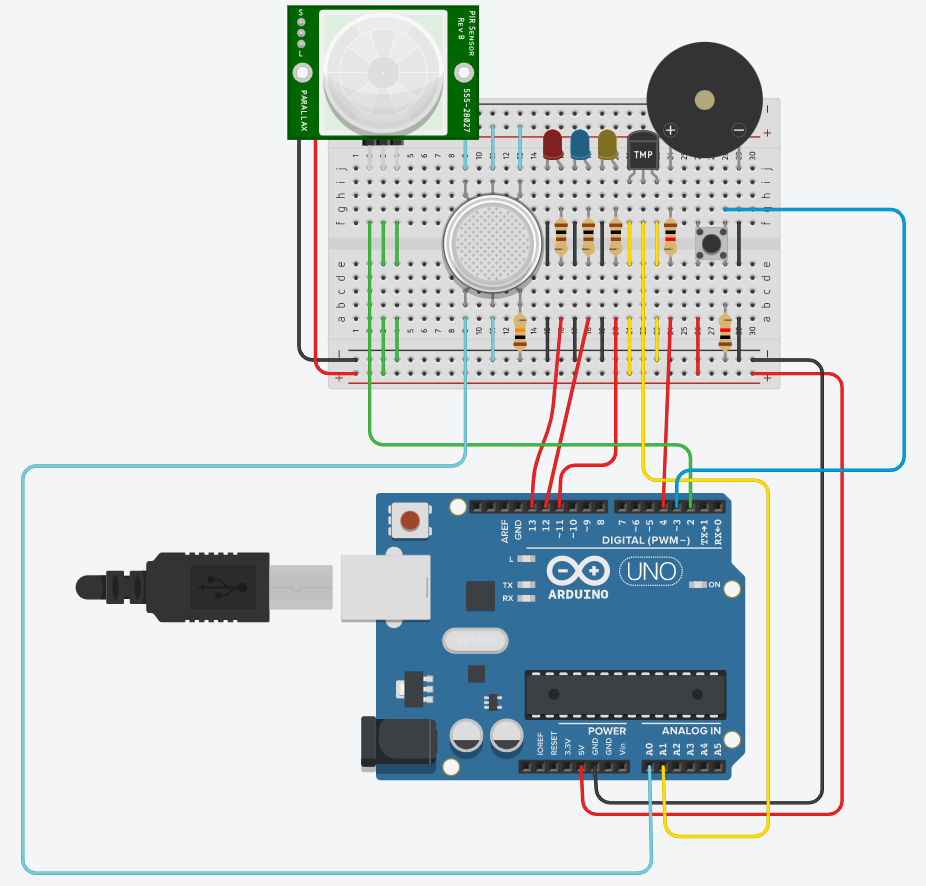
1. Описание на проекта
2. Блокова схема
3. Електрическа схема
4. Списък съставни части
5. Повече информация за съставните части
6. Какво представлява Arduino-то
7. Повече за Arduino Software
8. Сорс код – описание на функционалността
9. Бъдещо развитие
10. Полезни линкове
11. Приложение

**Описание на проекта**

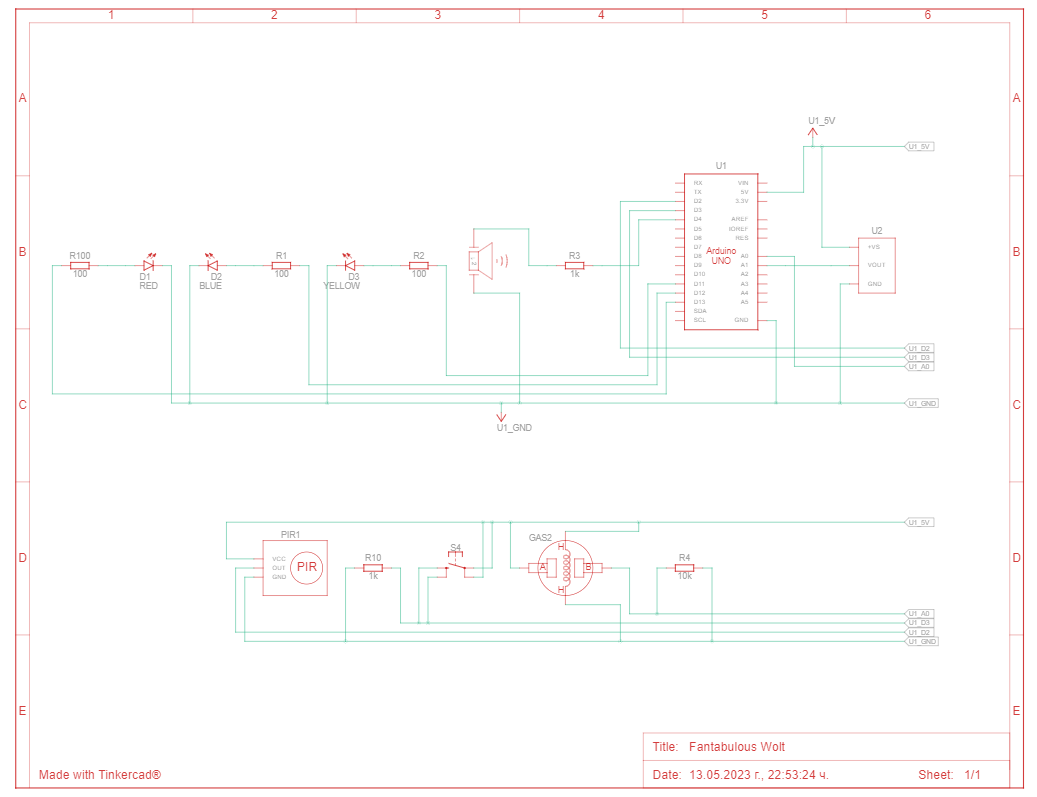
Основната цел на проекта е да подобри безопасността във всякакъв вид помещения. Устройството ще може да известява собствениците на помещението за различни видове опасности: влизане с взлом, пожар, задимяване и опасно високи температури чрез звукови и светлинни сигнали.

Подобно устройство би намалило разходите на лица, които по принцип би трябвало да закупят защитна система за всяко едно от изброените произшествия. Заради това Smart Security System би се превърнала в предпочитан избор за домакинства и малки бизнеси.

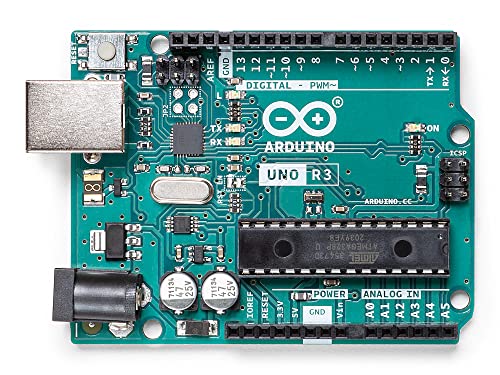
**Блокова схема**

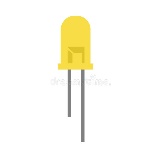


**Електрическа схема**



**Списък на съставните части**

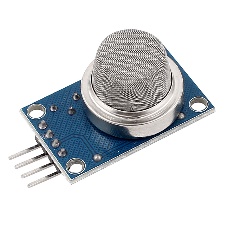
* Arduino Uno R3
* Red LED, Blue LED, Yellow LED





* 3 x 100 Ω Resistor, 2 x 1 kΩ Resistor
* Piezo



* PIR Sensor
* Temperature Sensor
* Gas Sensor
* Pushbutton



**Повече информация за съставните части**

* Arduino Uno R3 – микроконтролер, който захранва и управлява цялото устройство
* Red LED, Blue LED, Yellow LED – лед диоди, които индикират случай на опасност
* 3 x 100 Ω Resistor, 2 x 1 kΩ Resistor – резистори, които ограничават големината на тока, който тече през съответната верига
* Piezo – аларма, която издава звуков сигнал при опасност
* PIR Sensor – сензор, който засича движение
* Temperature Sensor – сензор, който отчита големината на температурата
* Gas Sensor – сензор, който отчита количеството газ около него
* Pushbutton – бутон, който служи за включване и изключване на устройството

**Какво представлява Arduino-то**

**Ардуино** е проект с [отворен код](https://bg.wikipedia.org/wiki/%D0%9E%D1%82%D0%B2%D0%BE%D1%80%D0%B5%D0%BD_%D0%BA%D0%BE%D0%B4) и едноименна компания, произлязла от него, както и свързаното с него потребителско общество. Основната цел е проектиране и производство на електронна [платформа](https://bg.wikipedia.org/wiki/%D0%9F%D0%BB%D0%B0%D1%82%D1%84%D0%BE%D1%80%D0%BC%D0%B0_(%D0%BA%D0%BE%D0%BC%D0%BF%D1%8E%D1%82%D1%80%D0%B8)) с лесен за ползване [свободен хардуер](https://bg.wikipedia.org/wiki/%D0%A1%D0%B2%D0%BE%D0%B1%D0%BE%D0%B4%D0%B5%D0%BD_%D1%85%D0%B0%D1%80%D0%B4%D1%83%D0%B5%D1%80) и [софтуер](https://bg.wikipedia.org/wiki/%D0%A1%D0%BE%D1%84%D1%82%D1%83%D0%B5%D1%80), позволяващи постигането на [интерактивност](https://bg.wikipedia.org/wiki/%D0%98%D0%BD%D1%82%D0%B5%D1%80%D0%B0%D0%BA%D1%82%D0%B8%D0%B2%D0%BD%D0%BE%D1%81%D1%82) за неспециалисти. [Електронните платки](https://bg.wikipedia.org/wiki/%D0%9F%D0%B5%D1%87%D0%B0%D1%82%D0%BD%D0%B0_%D0%BF%D0%BB%D0%B0%D1%82%D0%BA%D0%B0) „Ардуино“ могат да бъдат закупени или готови, или като „направи си сам“ комплекти, като [схемите](https://bg.wikipedia.org/wiki/%D0%95%D0%BB%D0%B5%D0%BA%D1%82%D1%80%D0%BE%D0%BD%D0%BD%D0%B0_%D1%81%D1%85%D0%B5%D0%BC%D0%B0) им са свободно достъпни за всеки, който би искал да ги сглоби сам. Проектът е основан на семейство платки с [микроконтролери](https://bg.wikipedia.org/wiki/%D0%9C%D0%B8%D0%BA%D1%80%D0%BE%D0%BA%D0%BE%D0%BD%D1%82%D1%80%D0%BE%D0%BB%D0%B5%D1%80), произвеждани главно от SmartProjects, Италия, както и редица други доставчици, които използват различни 8-битови (AVR) микроконтролери или 32-битови (ARM) [процесори](https://bg.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D1%86%D0%B5%D1%81%D0%BE%D1%80) Atmel. Осигурени са групи от [цифрови](https://bg.wikipedia.org/wiki/%D0%A6%D0%B8%D1%84%D1%80%D0%BE%D0%B2_%D1%81%D0%B8%D0%B3%D0%BD%D0%B0%D0%BB) и [аналогови](https://bg.wikipedia.org/wiki/%D0%90%D0%BD%D0%B0%D0%BB%D0%BE%D0%B3%D0%BE%D0%B2_%D1%81%D0%B8%D0%B3%D0%BD%D0%B0%D0%BB) щифтове за вход-изход (I/O), които позволяват свързване с други платки и вериги. Платките включват сериен комуникационен интерфейс, а при някои модели и [USB](https://bg.wikipedia.org/wiki/USB), за зареждане на програми. С оглед програмирането на микроконтролерите „Ардуино“ платформата предоставя [интегрирана среда за разработка](https://bg.wikipedia.org/wiki/%D0%98%D0%BD%D1%82%D0%B5%D0%B3%D1%80%D0%B8%D1%80%D0%B0%D0%BD%D0%B0_%D1%81%D1%80%D0%B5%D0%B4%D0%B0_%D0%B7%D0%B0_%D1%80%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B0) (IDE), основана на проекта [Processing](https://bg.wikipedia.org/w/index.php?title=Processing&action=edit&redlink=1" \o "Processing (страницата не съществува)), който поддържа програмните езици [C](https://bg.wikipedia.org/wiki/C_(%D0%B5%D0%B7%D0%B8%D0%BA_%D0%B7%D0%B0_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%B8%D1%80%D0%B0%D0%BD%D0%B5)) и [C++](https://bg.wikipedia.org/wiki/C%2B%2B). Първата платка „Ардуино“ е представена през 2005 г. Екипът на проекта иска да предостави на любители, ученици и професионалисти евтин и лесен начин да създават устройства, способни да взаимодействат с околната среда чрез [сензори](https://bg.wikipedia.org/wiki/%D0%A1%D0%B5%D0%BD%D0%B7%D0%BE%D1%80) и [изпълнителни устройства](https://bg.wikipedia.org/wiki/%D0%98%D0%B7%D0%BF%D1%8A%D0%BB%D0%BD%D0%B8%D1%82%D0%B5%D0%BB%D0%BD%D0%BE_%D1%83%D1%81%D1%82%D1%80%D0%BE%D0%B9%D1%81%D1%82%D0%B2%D0%BE). Обичайни примери са създаването на прости [роботи](https://bg.wikipedia.org/wiki/%D0%A0%D0%BE%D0%B1%D0%BE%D1%82), [термостати](https://bg.wikipedia.org/wiki/%D0%A2%D0%B5%D1%80%D0%BC%D0%BE%D1%81%D1%82%D0%B0%D1%82) и [датчици](https://bg.wikipedia.org/wiki/%D0%94%D0%B0%D1%82%D1%87%D0%B8%D0%BA) за движение. В средата на 2011 година са произведени приблизително 300 000 комерсиални „Ардуино“ устройства, а през 2013 година броят им нараства до 700 000.

## История

## През 2001 г. Кейси Рийс и Бенджамин Фрай стартират проекта Processing, чиято цел е да дадат възможност на непрограмисти да визуалзират графики на екран бързо и лесно. Той вдъхновява други подобни проекти, като един от тях е проектът „Уайринг“ (на [английски](https://bg.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8_%D0%B5%D0%B7%D0%B8%D0%BA): Wiring – в буквален превод: електрическа инсталация), насочен към потребители като художници, дизайнери и други хора с различно от техническото образование. Негов автор е колумбийския студент Ернандо Бараган (на [испански](https://bg.wikipedia.org/wiki/%D0%98%D1%81%D0%BF%D0%B0%D0%BD%D1%81%D0%BA%D0%B8_%D0%B5%D0%B7%D0%B8%D0%BA): Hernando Barragán), който през 2003 г. прилага принципите на Processing, за да създаде платформа за изработване на електронни прототипи, състояща се от [програмен език](https://bg.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%B5%D0%BD_%D0%B5%D0%B7%D0%B8%D0%BA), интегрирана среда за разработка и едноплатков [микроконтролер](https://bg.wikipedia.org/wiki/%D0%9C%D0%B8%D0%BA%D1%80%D0%BE%D0%BA%D0%BE%D0%BD%D1%82%D1%80%D0%BE%D0%BB%D0%B5%D1%80" \o "Микроконтролер). За разлика от Processing, ориентиран към визуалните изкуства, „Уайринг“ е проектиран да увлича хората към електрониката. Тази платка е предшественик на „Ардуино“.

„Ардуино“ е свързан с Висшето училище за интерактивен дизайн в [Иврея](https://bg.wikipedia.org/wiki/%D0%98%D0%B2%D1%80%D0%B5%D1%8F" \o "Иврея), Италия (Interaction Design Institute Ivrea), в което обучението се фокусира върху човешкото взаимодействие с цифровите продукти и системи и тяхното обратно въздействие върху потребителите. През 2005 г. група студенти създава проект на име „Arduino“, подобен на „Уайринг“. Името идва от бар, наречен на свой ред на маркграф и крал [Ардуин от Иврея](https://bg.wikipedia.org/wiki/%D0%90%D1%80%D0%B4%D1%83%D0%B8%D0%BD_%D0%BE%D1%82_%D0%98%D0%B2%D1%80%D0%B5%D1%8F), където част от основателите обичали да се събират. Други имена на проекта са Margrave of Ivrea и King of Italy.

Ядрото на групата е съставено от Масимо Банци (Massimo Banzi), Дейвид Куартелие (David Cuartielles), Том Иго (Tom Igoe), Джанлука Мартино (Gianluca Martino), Дейвид Мелис (David Mellis).

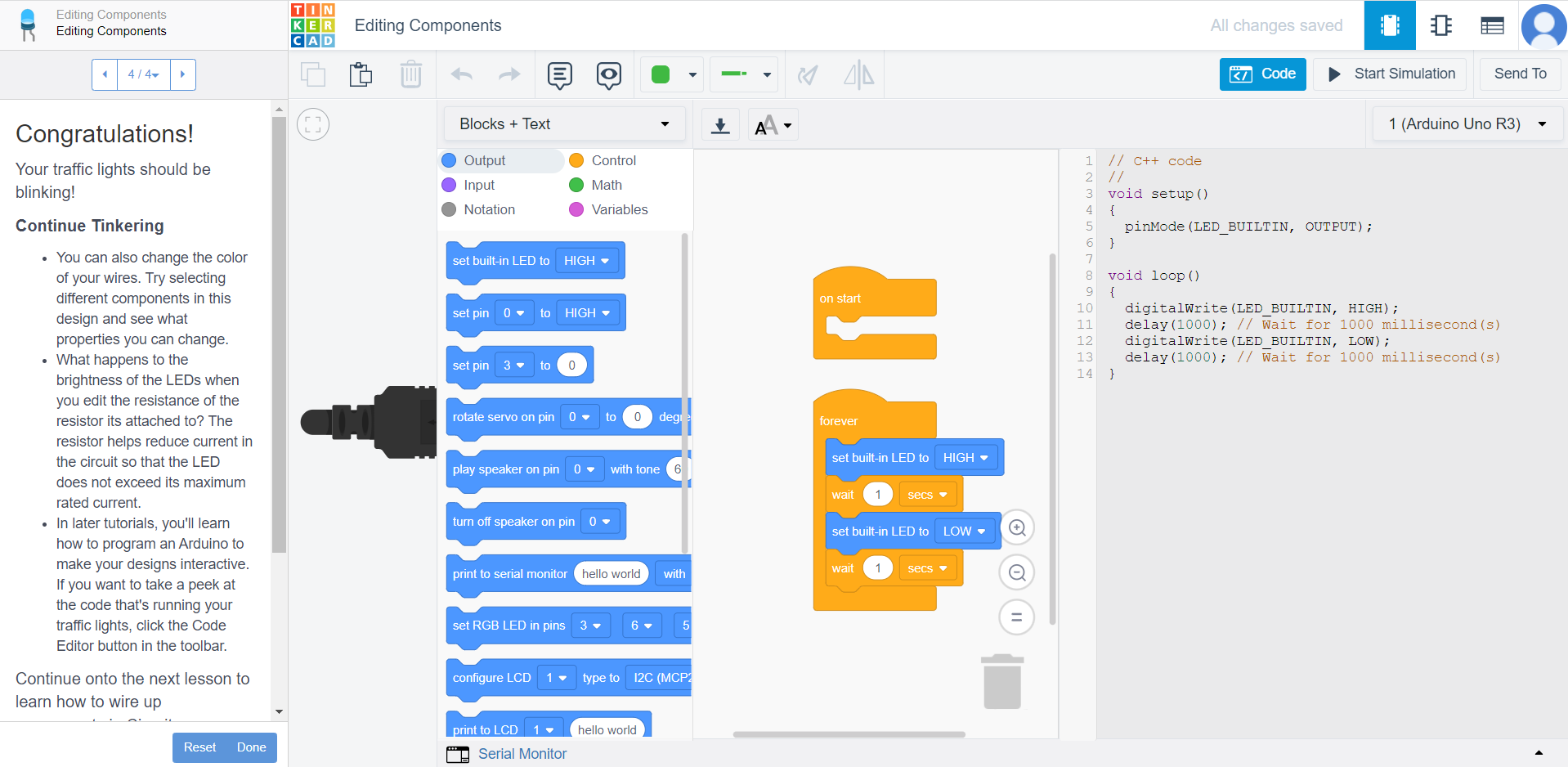
От 2008 г. обаче в групата настъпва разцепление. Джанлука Мартино регистрира друга компания, регистрира [търговска марка](https://bg.wikipedia.org/wiki/%D0%A2%D1%8A%D1%80%D0%B3%D0%BE%D0%B2%D1%81%D0%BA%D0%B0_%D0%BC%D0%B0%D1%80%D0%BA%D0%B0) Arduino в някои страни и започва продажби на нов сайт [arduino.org](http://www.arduino.org/) [Архив на оригинала от](https://web.archive.org/web/20170627202550/http:/www.arduino.org/) 2017-06-27 в [Wayback Machine](https://bg.wikipedia.org/wiki/Wayback_Machine)., докато първоначалната компания контролира продажбите на сайта [arduino.cc](https://www.arduino.cc/). Наборът от изделия на двата сайта е различен. Съществуват и две разклонения на Arduino IDE, поддържащи различни платки и библиотеки. Еднаквите названия и номера на версиите на IDE внасят силно объркване.

## Приложения

* Xoscillo: open-source [Осцилоскоп](https://bg.wikipedia.org/wiki/%D0%9E%D1%81%D1%86%D0%B8%D0%BB%D0%BE%D1%81%D0%BA%D0%BE%D0%BF)
* Научна екипировка
* [Arduinome](https://bg.wikipedia.org/w/index.php?title=Arduinome&action=edit&redlink=1): [Midi controller](https://bg.wikipedia.org/w/index.php?title=Midi_controller&action=edit&redlink=1) който имитира [Monome](https://bg.wikipedia.org/w/index.php?title=Monome&action=edit&redlink=1)
* [Ardupilot](https://bg.wikipedia.org/w/index.php?title=Ardupilot&action=edit&redlink=1): Софтуер за дрони
* ArduinoPhone
* GertDuino, an Arduino mate for the [Raspberry Pi](https://bg.wikipedia.org/wiki/Raspberry_Pi)
* Water quality testing platform

**Повече за Arduino Software и Tinkercad**

Tinkercad е безплатна онлайн програма за 3D моделиране, която работи в уеб браузър. Откакто става наличен през 2011 г., тя се превръща в популярна платформа за създаване на модели за 3D принтиране, както и въведение на начално ниво в конструктивната стереометрия в училищата.



Програмата позволява създаване на 3D Arduino модели чрез код (за напреднали) и блокове (за начинаещи)

Arduino Софтуер

Интегрираната среда за разработка на „Ардуино“ е мултиплатформено приложение, написано на програмния език [Java](https://bg.wikipedia.org/wiki/Java) и произлиза от езика за програмиране „Processing“ и проекта „Wiring“. Конструирана е така, че да улесни програмирането от хора, които не са запознати с писането на софтуер. Съдържа редактор на код с функции като:

* syntax highlighting (показване на текст в различен цвят в зависимост от принадлежността на термините)
* brace matching (функция, която следи отварящите и затварящите скоби с цел по-лесна навигация през програмния код),
* автоматично подравняване

и също така е способен да [компилира](https://bg.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%D0%B8%D0%BB%D0%B0%D1%82%D0%BE%D1%80) и да качва програми към устройството с един клик.

Програма, написана за „Ардуино“, се нарича „скеч“ и е на програмните езици C и C++. Средата за разработка „Ардуино“ идва със [софтуерна библиотека](https://bg.wikipedia.org/wiki/%D0%A1%D0%BE%D1%84%D1%82%D1%83%D0%B5%D1%80%D0%BD%D0%B0_%D0%B1%D0%B8%D0%B1%D0%BB%D0%B8%D0%BE%D1%82%D0%B5%D0%BA%D0%B0) Wiring (името идва от оригиналния Wiring проект), което улеснява много входни/изходни операции.

За да направят циклично изпълняваща се програма, потребителите трябва да дефинират само две функции,:

* Setup(): Функция, която се изпълнява веднъж в началото на програмата и може да инициализира настройки;
* Loop(): Функция, която не спира да се извиква докато „Ардуино“ не се изключи.

Типична първа програма за микроконтролер е просто примигваща LED светлинка. В средата за разработка на „Ардуино“ потребителят може да напише програма като:

#define LED\_PIN 13

void setup() {

pinMode(LED\_PIN, OUTPUT); *// Enable pin 13 for digital output*

}

void loop() {

digitalWrite(LED\_PIN, HIGH); *// Turn on the LED*

delay(1000); *// Wait one second (1000 milliseconds)*

digitalWrite(LED\_PIN, LOW); *// Turn off the LED*

delay(1000); *// Wait one second*

}

* Предимството на повечето „Ардуино“ платформи е, че имат светлинен индикатор и специфичен резистор, свързани между (pin 13 и земя), което е доста удобно за прости тестове. Горният код не може да бъде видян от стандартен C++ компилатор като валидна програма, затова, когато потребителят кликне бутона „Upload to I/O board“ в средата за разработка, копие на кода се записва във временен файл, в който се добавя функцията main(), за да се валидира кода. Средата за разработка на „Ардуино“ използва GNU toolchain и AVR Libc, за да компилира програми и използва Avrdude, за да качва програмите на устройството. Вследствие на това, че „Ардуино“ използва Atmel микроконтролери, средата за разработка на Atmel AVR Studio може също да се използва за писане на софтуер за „Ардуино“.

**Сорс код**

// C++ code

//

int motionSensor = 0;

int gasSensor = 0;

int temp = 0;

int isOn = 1;

int counter;

void setup()

{

  Serial.begin(9600);

  pinMode(2, INPUT);

  pinMode(A1, INPUT);

  pinMode(A0, INPUT);

  pinMode(3, INPUT);

  pinMode(11, OUTPUT);

  pinMode(12, OUTPUT);

  pinMode(13, OUTPUT);

  pinMode(4, OUTPUT);

  pinMode(LED\_BUILTIN, OUTPUT);

}

void loop()

{

  motionSensor = digitalRead(2);

  temp = map(((analogRead(A1) - 20) \* 3.04), 0, 1023, -40, 125);

  gasSensor = analogRead(A0);

  int buttonState = digitalRead(3);

  Serial.println(buttonState);

  if (buttonState == 1) {

    isOn = abs(isOn-1);

  }

  if (isOn == 1 && (temp > 40 && gasSensor > 500)) {

    digitalWrite(11, HIGH);

    digitalWrite(12, LOW);

    digitalWrite(13, LOW);

    delay(100); // Wait for 100 millisecond(s)

    digitalWrite(11, LOW);

    digitalWrite(12, HIGH);

    digitalWrite(13, LOW);

    delay(100); // Wait for 100 millisecond(s)

    digitalWrite(11, LOW);

    digitalWrite(12, LOW);

    digitalWrite(13, HIGH);

    delay(100); // Wait for 100 millisecond(s)

  }

  if (isOn == 1 && gasSensor > 500) {

    tone(4, 548668578, 200); // play tone 300 (C25 = 548668578 Hz)

    digitalWrite(11, HIGH);

    delay(200); // Wait for 200 millisecond(s)

    digitalWrite(11, LOW);

    delay(200); // Wait for 200 millisecond(s)

  }

  if (isOn == 1 && temp > 40) {

    tone(4, 57079043007393, 200); // play tone 500 (G#41 = 57079043007393 Hz)

    digitalWrite(11, HIGH);

    digitalWrite(13, LOW);

    delay(200); // Wait for 200 millisecond(s)

    digitalWrite(11, LOW);

    digitalWrite(13, HIGH);

    delay(200); // Wait for 200 millisecond(s)

    digitalWrite(13, LOW);

  }

  if (isOn == 1 && motionSensor == HIGH) {

    for (counter = 0; counter < 15; ++counter) {

      tone(4, 5274, 200); // play tone 100 (E8 = 5274 Hz)

      digitalWrite(LED\_BUILTIN, HIGH);

      digitalWrite(12, LOW);

      delay(200); // Wait for 200 millisecond(s)

      digitalWrite(LED\_BUILTIN, LOW);

      digitalWrite(12, HIGH);

      delay(200); // Wait for 200 millisecond(s)

      tone(4, 5274, 200); // play tone 100 (E8 = 5274 Hz)

    }

    digitalWrite(12, LOW);

    noTone(4);

  }

  delay(200);

}

**Бъдещо развитие**

В бъдеще към устройството може да бъде добавен Keypad, на който да се въвежда парола, която да изключва алармата.

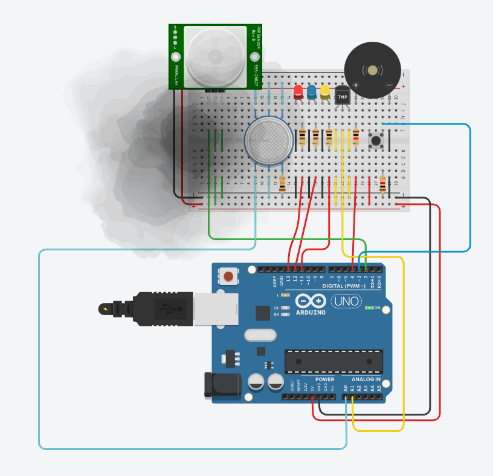
**Полезни линкове**

<https://www.tinkercad.com/>

<https://docs.arduino.cc/>

**Приложение**

GitHub Link: <https://github.com/georgikolev17/Smart-Security-System/>

Tinkercad Link: <https://www.tinkercad.com/things/7hBKZXrnYli/>