



- [9]. Podorozhniak, A., Liubchenko, N., & Onishchenko, D. (2021). An intelligent system for recognizing vehicle number plates [Conference presentation]. I International science conference on multidisciplinary research, Berlin, Germany, 2021, January 19-21, pp. 1062-1066. <https://doi.org/10.46299/ISG.2021.II>
- [10]. Lecun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86 (11), pp. 2278–2324. <https://doi.org/10.1109/5.726791>.
- [11]. Kuchuk, H., Podorozhniak, A., Liubchenko, N., & Onishchenko, D. (2021). System of license plate recognition considering large camera shooting angles. Radioelectronic and Computer Systems, 2021, no. 4 (100), pp. 82 – 91. <https://doi.org/10.32620/reks.2021.4.07>
- [12]. He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017). Mask R-CNN / 2017 IEEE International Conference on Computer Vision (ICCV), Venice, October 22-29 2017. – Venice, 2017. – pp. 2980-2988. <https://doi.org/10.1109/ICCV.2017.322>
- [13]. Nomeroff Net. Automatic numberplate recognition system from AUTO.RIA.com. Version 0.2.3. [Web resource]. URL: <https://nomeroff.net.ua/> (last access 27.01.2023)
- [14]. FF-GROUP. [Web resource]. URL: <https://ff-group.org/seeauto> (last access 27.01.2023)
- [15]. Medialab LPR database. [Web resource]. URL: <http://www.medialab.ntua.gr/research/LPRdatabase.html> (last access 27.01.2023)
- [16]. Raspberry PI 4 Official site. [Web resource]. URL: <https://www.raspberrypi.com/products/raspberry-pi-4-model-b/> (last access 27.01.2023)
- [17]. Kaimkhani, N. A. K., Noman, M., Rahim, S., & Liaqat H. B. (2022). UAV with Vision to Recognize Vehicle Number Plates. Mobile Information System, 2022, vol. 4, article ID 7655833. <https://doi.org/10.1155/2022/7655833>

Отримана в редакції 28.06.2023. Прийнята до друку 02.08.2023. Received 28 June 2023. Approved 02 August 2023. Available in Internet 12 September 2023.

УДК 004.415.53:004.424

## SURVEILLANCE AND SECURITY MODELS BASED ON THE ARDUINO UNO INTERFACE

Zakariashvili M.Z.<sup>1</sup>, Sivsivadze G.<sup>2</sup>, Chidrashvili M.<sup>3</sup>, Khechoshvili L.<sup>4</sup>

<sup>1,2,3,4</sup> Iakob Gogebashvili Telavi State University, Telavi, Georgia

ORCID: <https://orcid.org/0000-0003-3903-6636><sup>1</sup>

E-mail: <sup>1</sup>[mariam.zaqariashvili@tesau.edu.ge](mailto:mariam.zaqariashvili@tesau.edu.ge), <sup>2</sup>[guram.sivsivadze@tesau.edu.ge](mailto:guram.sivsivadze@tesau.edu.ge), <sup>3</sup>[mate.chidrashvili@tesau.edu.ge](mailto:mate.chidrashvili@tesau.edu.ge),

<sup>4</sup>[luka.khechoshvili@tesau.edu.ge](mailto:luka.khechoshvili@tesau.edu.ge)

Copyright © 2021 by author and the journal “Automation of technological and business – processes”.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0>



DOI: 10.15673/atbp.v15i3.2624

**Abstract.** The issue demonstrated in the article covers construction of surveillance and security systems on Arduino UNO platform Technological and Software aspect which are based on Quality Semiconductor Sensor-MQ135; PIR Motion Sensor and use of blocking peculiarities.

The project model integration process in both a virtual and physical context is described in detail and analyzed in the paper. The following are specifically used: An Arduino UNO microcontroller board, the Arduino Software IDE (Integrated Development Environment), a local code editor on the PC, and the virtual online platform tinkercad.com. The work is supplemented by pertinent electronic diagrams, schemes, software code, descriptions of fundamental functionalities, links to virtual project simulation, real-world application, and links to video recordings of the experiment in order to demonstrate the project model's smooth operation.

**Анотація.** Проблема, що розглядається в статті, охоплює побудову систем відеоспостереження та безпеки на платформі Arduino UNO. Технологічний та програмний аспекти, які базуються на Quality Semiconductor Sensor-MQ135; PIR датчик руху та особливості використання блокування.

У статті детально описано та проаналізовано процес інтеграції моделі проекту як у віртуальному, так і у фізичному контексті. Конкретно використовуються: плата мікроконтролера Arduino UNO, Arduino Software



IDE (інтегроване середовище розробки), локальний редактор коду на ПК та віртуальна онлайн-платформа tinkercad.com. Робота доповнена відповідними електронними діаграмами, схемами, кодом програмного забезпечення, описом основних функціональних можливостей, посиланнями на віртуальне моделювання проекту, застосуванням у реальному світі та посиланнями на відеозаписи експерименту, щоб продемонструвати безперерйну роботу моделі проекту.

**Keywords:** Arduino UNO; Arduino IDE; tinkercad.com; Quality Sensor- MQ135; PIR Motion Sensor

**Ключові слова:** Arduino UNO; Arduino IDE; tinkercad.com; Датчик якості - MQ135; PIR датчик руху

## I. Introduction

The "Fourth Industrial Revolution" era is defined by the fields of artificial intelligence, electronics, and programming... Our lives are being rapidly impacted by technological advancements that are erasing the distinctions between the physical, digital, and biological realms. There are a lot of "smart" gadgets in the technological world, some of which even have intelligence that is akin to human intellect and is built like it.

The article will discuss the use of one of these unique devices, the Arduino UNO.

In general, Arduino was developed with students in mind, enabling them to quickly make electronic projects. But today, high school students, college students, and anybody else interested in the subject are all fascinated by studying the technology behind using Arduino in both official and informal settings.

Naturally, interest in using Arduino in our practice also increased. The research's objective was to examine the technical facets of using an Arduino UNO. When creating the models for our research project, "Arduino UNO interface-based sensors" (AIR Quality Sensor- MQ135; PIR Motion Sensor), this included developing a methodology for the integration and practical implementation of the online service <https://www.tinkercad.com/>, the Arduino IDE programming editor, and the Arduino UNO microcontroller in both a virtual and real environment.

The technological and software resources needed for projects based on the Arduino UNO interface are freely accessible on the Internet via its official website <https://www.arduino.cc/>. Recommendations for working with Arduino and helpful practical tutorials are also provided in the Georgian educational space. (Zakariashvili M., 2022); (Mosashvili I., Oniani S. 2016); (Tabatadze Z., Todua T. 2019); (Geddes, Mark. 2016).

In our opinion, this work will assist students learn about and explore the interesting world of Arduino to some extent in addition to existing instructional resources.

## II. Object, Subject, And Methods Of Research

### 2.1 Basic Functionalities

- **What is Arduino?** Arduino UNO -- is an open source electronics platform based on hardware and software. Basically, Arduino is a small computer that can be programmed to interact with physical objects through various input and output signals. Today, the most common model is the Arduino UNO. (See Fig. N 1)
- **What is Tinkercad.com?** <https://www.tinkercad.com/> is an online portal that can be used to create/simulate virtual projects based on the Arduino UNO microcontroller; Additionally, it is used to execute various commands through electronic circuits, write and debug software codes, automatically manage the system in accordance with the code. The platform is open source. Registration/authorization is possible with a gmail.com account (See Fig. N2; N3; N4)



Fig. N1 Arduino UNO

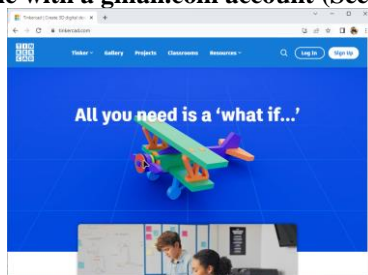


Fig. N2 tinkercad.com

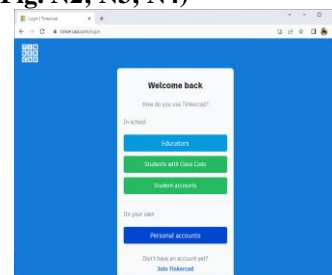


Fig. N3 tinkercad.com

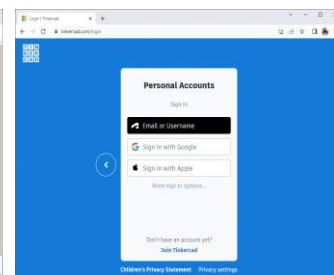


Fig. N4 tinkercad.com

- **What is Arduino IDE?** [Arduino Software IDE](#) is an open source software editor that enables you to use the Arduino IDE programming language and the Arduino development environment to write the codes required to make the



<http://www.atbp.ontu.edu.ua/>

product function and specify what it will accomplish. The Windows installation package of the Arduino IDE can be downloaded and installed on a computer for free via the link <https://www.arduino.cc/en/software>. (See Fig. N5; N6)



Fig. N5 Arduino.cc



Fig. N6 Arduino IDE

## 2.2 Integration of Arduino UNO and Arduino IDE

Putting the project's software code in the Arduino IDE environment, saving it there, and uploading it to the ARDUINO UNO.

The actual electronic scheme of the project will be produced with the help of the virtual scheme created on the online portal <https://www.tinkercad.com/>; The project code will be saved as a "\*.ino" file and placed on the Arduino IDE desktop. The software code will then be uploaded to the Arduino UNO board, the main control element of the actual electronic circuit, which is connected to the computer using a special USB cable for the Arduino UNO microcontroller. This allows data transmission and reception, program loading, and powering of the Arduino. The Arduino can also be powered by a typical power supply device. In this instance, the project will be implemented practically without the use of a computer.

### III. Projects Models Based on the Arduino UNO Interface

We present the following projects:

- ✓ PIR Motion Sensor - Surveillance Alarm
- ✓ AIR Quality Sensor MQ135 - Security Alarm

Before the projects are actually implemented, a preliminary online virtual simulation will be performed on the <https://www.tinkercad.com/> portal; we'll then carry out an actual experiment with the electronic model of the project by integrating the Arduino IDE software application on the Arduino UNO interface from the computer, which will then be transferred to the real space independently of the computer.

#### 3.1 Interfacing Arduino uno with PIR motion sensor

**Project name:** PIR Motion Sensor - Surveillance Alarm

**Project goal:** Building a surveillance model using a PIR sensor/ Automatic management on the Arduino UNO interface.

**The project model's operational algorithm is described as follows:** The PIR sensor built into the Arduino UNO interface can detect and sense movement up close, such as people moving around a room. The red notification lamp turns on, the alarm begins to sound, and a message or warning about a change in the designated space is shown on the monitor in accordance with the software code. Otherwise, the light is green, the alarm is off, and peace-related information is shown on the monitor.

**Components needed to build the project model:** Arduino UNO; Breadboard; Arduino USB 2.0.Cable; MQ135 Semiconductor Sensor for Air Quality; Piezo; LCD 16x2; Potentiometer; Resistors; Red LED; Green LED; Jumper wires. (See Table N1- Project Components)

Table N1- Project Components: Surveillance Alarm

Name	Quantity	Component
U1	1	Arduino Uno R3
U2	1	MCP23008-based, 32 LCD 16 x 2 (I2C)
PIEZO2	1	Piezo
R1, R4	2	1 kΩ Resistor
D3	1	Red LED
D4	1	Green LED
PIR2	1	11.107423059381404 , -216.01903510641006 , -261.05971972043847 PIR Sensor

Source: <https://create.arduino.cc/projecthub/electronicxfan123/interfacing-arduino-uno-with-pir-motion-sensor-f05ae2>



<http://www.atbp.ontu.edu.ua/>

**Applications and online services:** <https://www.tinkercad.com/> [Arduino Software IDE](#)

**The Essence of Functioning PIR Motion Sensor:**

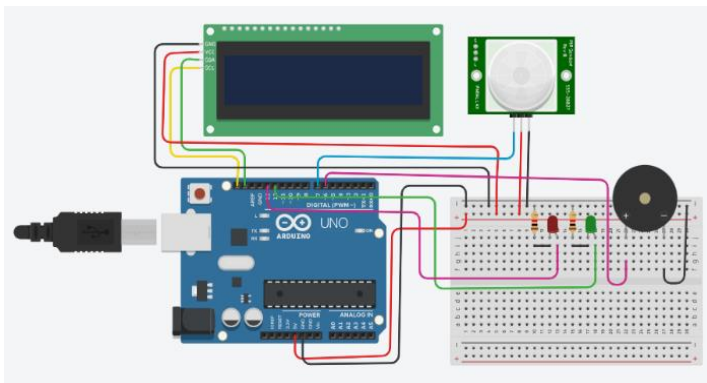
A PIR - Passive Infrared Sensor, also called as motion sensor is an electronic device which senses motion using a pair of pyroelectric sensors to detect heat energy in the surrounding environment. These two sensors sit beside each other, and when the signal differential between the two sensors changes (suppose if a person enters the room), the sensor will engage. It basically catches movement. It has three terminals namely Gnd, Vcc and signal pin with 3V regulator, time delay controller, sensitivity controller and BIS001. (See Fig. N7)



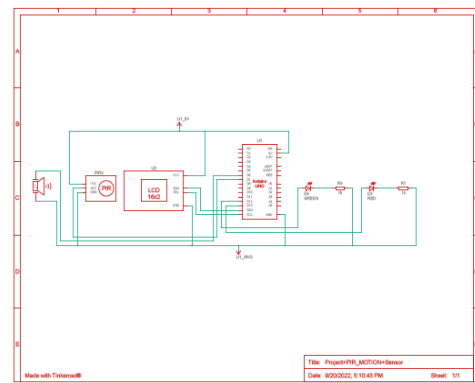
**Fig. N7 PIR Motion Sensor**

- **First Stage: tinkercad.com virtual environment**

Project circuit diagrams in tinkercad virtual online environment (See Fig. N8, N9)

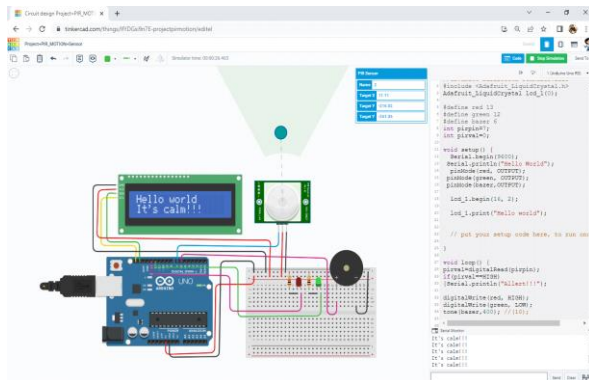


**Fig. N8 Virtual circuit**

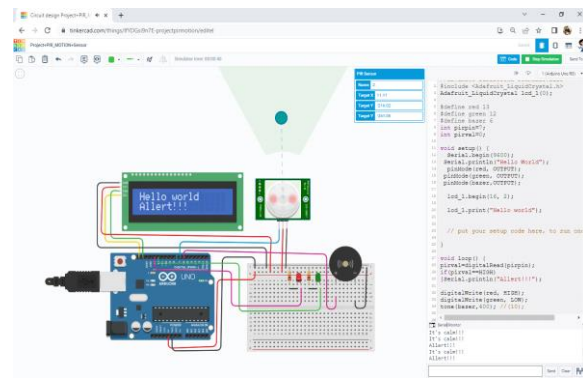


**Fig. N9 Virtual Diagram**

Project simulation in Tinkercad.com virtual online environment (See Fig. N10, N11)



**Fig. N10 Virtual simulation: It's Calme**



**Fig. N11 Virtual simulation: alert!**

The following link will allow you to view the project model, software code, and simulation in tinkercad.com's virtual online environment: <https://www.tinkercad.com/things/lfYDGSi9n7E>

- **Second stage : Arduino UNO and AIR Quality Sensor;**

Actual circuit diagram of the project. The picture shows a model of the real scheme of the project, which is built in accordance with the virtual scheme made in Tinkercad. (See Fig. N12)

- **Third stage: Arduino IDE, Arduino Code;**

Placing the project software code, saving it in the Arduino IDE environment, uploading it from the computer to the ARDUINO UNO, experiment

The software code of the project built in the Tinkercad.com online space will be placed on the Arduino IDE working board. It will be saved as "\*.ino" file and uploaded to the board of the actual built circuit-Arduino that is connected to the computer using the Arduino UNO's unique USB cable. Arduino IDE environment (See Fig. N13)

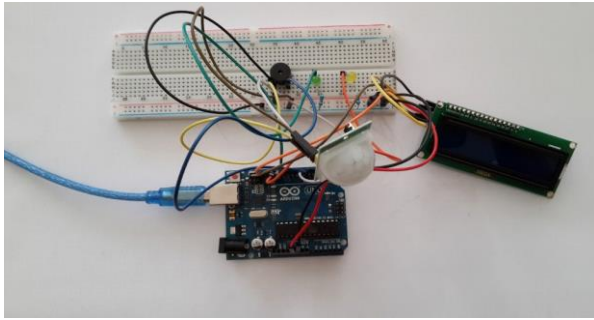


Figure N12 Actual Circuit

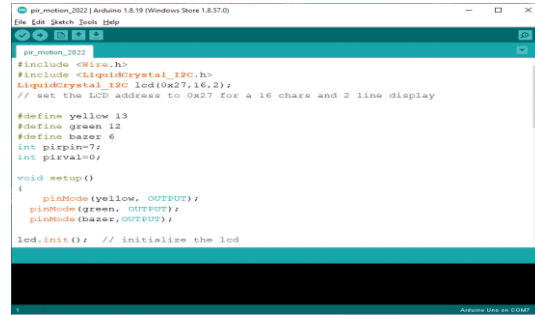


Fig. N13 Arduino IDE environment

### Software code

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27,16,2);
// set the LCD address to 0x27 for a 16 chars and 2 line
display
#define yellow 13
#define green 12
#define bazer 6
int pirpin=7;
int pirval=0;
void setup() {
  pinMode(yellow, OUTPUT);
  pinMode(green, OUTPUT);
  pinMode(bazer,OUTPUT);
  lcd.init(); // initialize the lcd i2C
  lcd.backlight();
  lcd.begin(16, 2);
  lcd.setCursor(0,0);
  Serial.begin(9600);
  // put your setup code here, to run once:
}
```

```
void loop() {
  pirval=digitalRead(pirpin);
  if(pirval==HIGH)
  {Serial.println("Alert!!!");
  digitalWrite(yellow, HIGH);
  digitalWrite(green, LOW);
  lcd.clear();
  lcd.print(" Alert");
  lcd.setCursor(12, 0);
  tone(bazer,200); //(10);
  delay(500);
  }
  else{Serial.println("It's calm!!!");
  digitalWrite(yellow, LOW);
  digitalWrite(green, HIGH);}
  lcd.setCursor(0, 0);
  lcd.print(" It's calm");
  noTone(bazer);
  // put your main code here, to run repeatedly:
}
```

- **Fourth stage: practical implementation of the project, experiment**

If the software code is successfully written on the ARDUINO UNO, the electronic scheme begins to function in accordance with the project algorithm. The Arduino IDE work window's Serial Monitor displays a corresponding illustration of the procedure on the computer screen. The built circuit model can be connected to a power source (up to 12 V adapter or specific batteries are allowed), and can be used to measure air quality/pollution for safety reasons. (See Fig. N14; N15)

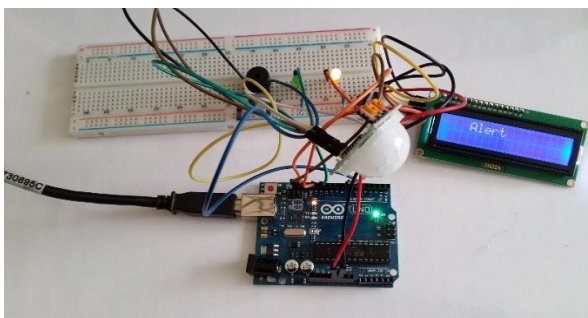


Fig. N15 Pir Motion: Alert!!!

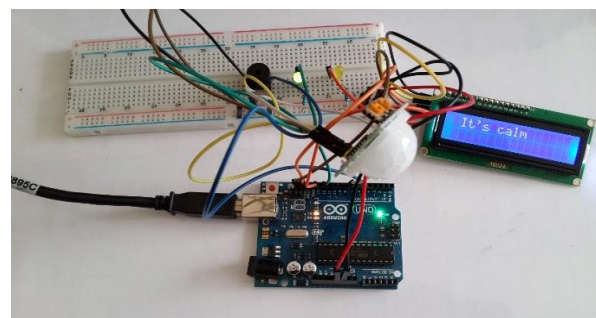


Fig. N16 Pir Motion: It's Calme

With the help of the link below, you can watch a video of the project being implemented in practice in a real-world setting. <https://drive.google.com/file/d/1Y3iIIBiP82Cr1BSk6I9s2S89BkE61AMD/view?usp=sharing>

### 3.2 Interfacing Arduino uno with AIR Quality Semiconductor Sensor - MQ135

**Project name:** AIR Quality Sensor MQ135 - Security Alarm

**Project goal:** Building a security model/automated control using the AIR sensor on the Arduino UNO interface.

**The project model's operational algorithm is described as follows:** The AIR Quality Sensor MQ135 on the Arduino UNO interface detects the level of air pollution. Additionally, it may find airborne contaminants including ammonia,



nitrogen oxides, alcohol, benzene, smoke, carbon dioxide, etc. The red warning lighting turns on in line with the software code, and a message (warning) regarding air pollution in the specified area is shown on the monitor. In other situations, the alert is silenced, the lamp turns green, and the monitor shows details on the air's non-toxic status.

**Components needed to build the project model:** Arduino UNO; Breadboard; Arduino USB 2.0.Cable; MQ135 Semiconductor Sensor for Air Quality; Piezo; LCD 16x2; Potentiometer; Resistors; LED RGB; Jumper wires. (See Table N1)

**Table N2- Project Components -Security Alarm**

Name	Quantity	Component
U2	1	Arduino Uno R3
U3	1	MCP23008-based, 32 LCD 16 x 2 (I2C)
GAS2	1	Gas Sensor
R3	1	4 kΩ Resistor
D1	1	LED RGB
R1, R2, R4	3	1 kΩ Resistor
PIEZO2	1	Piezo

**Resource:** [https://create.arduino.cc/projecthub/m\\_karim02/arduino-and-mq-135-gas-sensor-with-arduino-code-a8c1c6](https://create.arduino.cc/projecthub/m_karim02/arduino-and-mq-135-gas-sensor-with-arduino-code-a8c1c6)

**Applications and online services:** <https://www.tinkercad.com/> / [Arduino Software IDE](#)

**The essence of functioning of MQ135 Semiconductor AIR Quality Sensor**

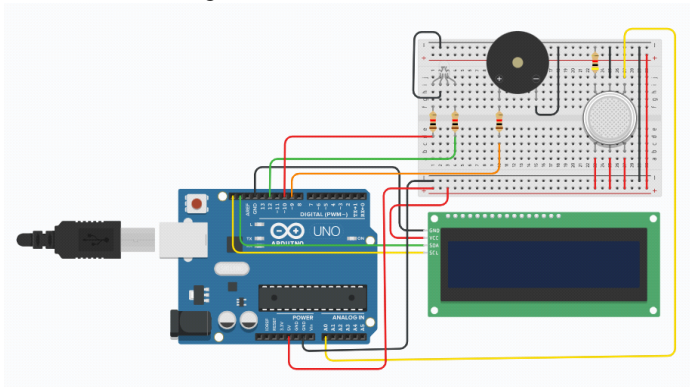
An MQ135 Air Quality Sensor is one type of MQ gas sensor used to detect, measure, and monitor a wide range of gases present in air like ammonia (NH3), Nitrogen oxides (NOx) alcohol, benzene, smoke, carbon dioxide (CO2), etc. It operates at a 5V supply with 150mA consumption. (See Fig.N17)



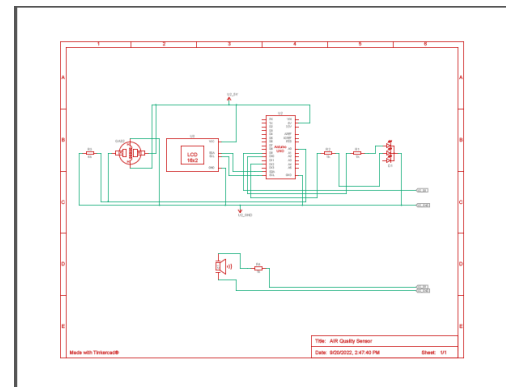
**Fig.17 Air Quality Sensor**

**The first stage: tinkercad.com virtual environment**

Project circuit diagram in tinkercad.com online virtual environment (See Fig.N18;N19)

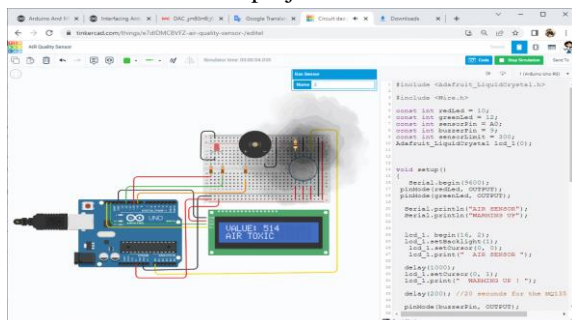


**Fig. 18 Virtual circuit**

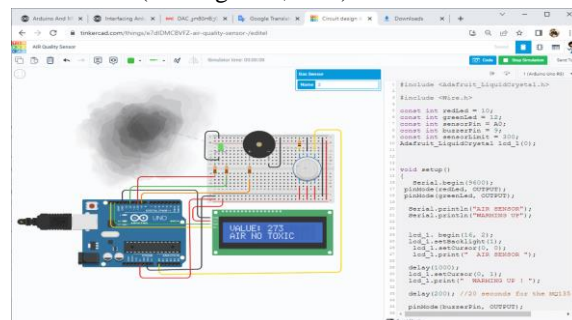


**Fig. N19 Virtual diagram**

Virtual simulation of the project in tinkercad.com online environment (See Fig.N20; N21)



**Fig. N20 Virtual simulation: Air Toxic**



**Fig. N21 Virtual simulation: Air No Toxic**



<http://www.atbp.ontu.edu.ua/>

The following link will allow you to view the project model, software code, and simulation in Tinkercad.com's virtual online environment: <https://www.tinkercad.com/things/e7dIDMCBVFZ>

- **The second stage: Arduino UNO and AIR Quality Sensor;**  
Actual circuit diagram of the project (See Fig. N 22)
- **The third stage: Arduino IDE, Arduino Code;**

Placing the project software code, saving it in the Arduino IDE environment, uploading it from the computer to the ARDUINO UNO, experiment.

The software code of the project built in the tinkercad.com online space will be placed on the Arduino IDE working board. It will be saved as "\*.ino" file and uploaded to the board of the actual built circuit -Arduino that is connected to the computer using the Arduino UNO's unique USB cable. Arduino IDE environment (See Fig. N23 )

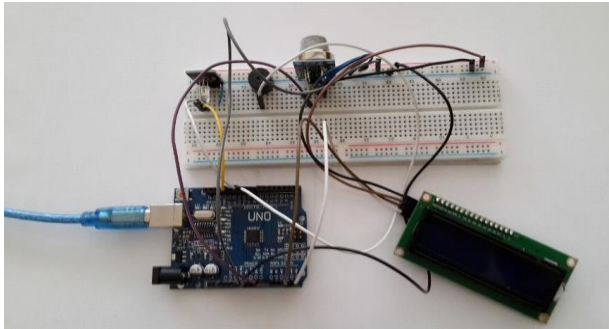


Fig. N22 Actual circuit

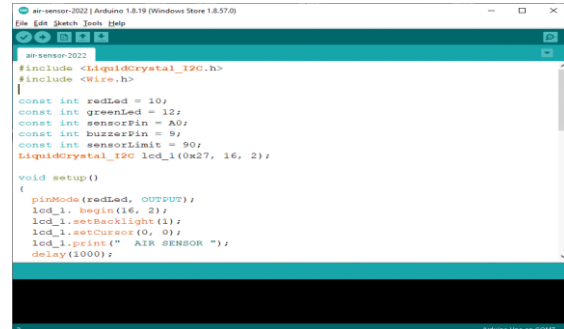


Fig. N23 Arduino IDE environment

### Software code

```
#include <LiquidCrystal_I2C.h>
#include <Wire.h>
const int redLed = 10;
const int greenLed = 12;
const int sensorPin = A0;
const int buzzerPin = 9;
const int sensorLimit = 90;
LiquidCrystal_I2C lcd_1(0x27, 16, 2);
void setup()
{
    pinMode(redLed, OUTPUT);
    lcd_1.begin(16, 2);
    lcd_1.setBacklight(1);
    lcd_1.setCursor(0, 0);
    lcd_1.print(" AIR SENSOR ");
    delay(1000);
    lcd_1.setCursor(0, 1);
    lcd_1.print(" WARMING UP ! ");
    delay(20000); //20 seconds for the MQ135 to
warm up
    pinMode(buzzerPin, OUTPUT);
    Serial.begin(9600);
}
void loop()
{
    lcd_1.clear();
    int sensorValue = analogRead(sensorPin);
    Serial.print("Pin A0: ");
    Serial.println(sensorValue);
    lcd_1.setCursor(0, 0);
    lcd_1.print("VALUE: ");
    lcd_1.print(sensorValue);
    lcd_1.print("ppm");
    if (sensorValue > sensorLimit)
    {
```

```
digitalWrite(redLed, HIGH);
    lcd_1.setCursor(0, 1);
    lcd_1.print("AIR CONTEMINETED");
    tone(buzzerPin,500, 2000 );
    }
else
    {
    digitalWrite(redLed, LOW);
    digitalWrite(greenLed, HIGH);
    lcd_1.setCursor(0, 1);
    lcd_1.print("NO CONTEMINETED");
    noTone(buzzerPin);
    }
    delay(500);
    }
}
```



• **Fourth stage: practical implementation of the project, experiment**

If the software code is successfully written on the ARDUINO UNO, the electronic scheme begins to function in accordance with the project algorithm. The Arduino IDE work window's Serial Monitor displays a corresponding illustration of the procedure on the computer screen. The built circuit model can be connected to a power source (up to 12 V adapter or specific batteries are allowed), and can be used to measure air quality/pollution for safety reasons. (See Fig. N25; N26)

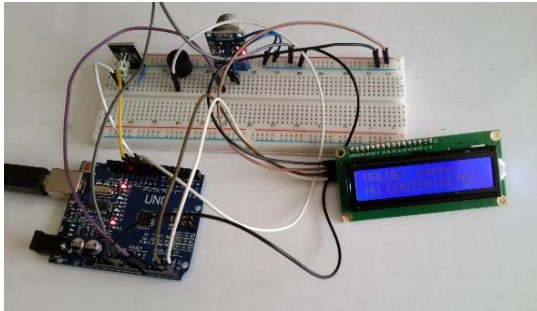


Fig.N25 AIR NO Contemeneted

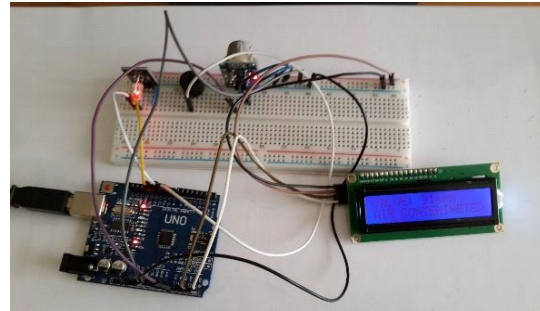


Figure N26 AIR Contemeneted

With the help of the link below, you can watch a video of the project being implemented in practice in a real-world setting.  
<https://drive.google.com/file/d/1BsiLVdjXCSoTpijdss2bljTT3ouO0Bw6/view?usp=sharing>

**IV. Conclusions**

The article covers the processes of modeling and building surveillance and security systems based on the Arduino UNO interface in virtual and real environments. <https://www.tinkercad.com/>; Arduino Software IDE (Integrated Development Environment) local programming editor on the computer and Arduino UNO electronic constructor were the tools utilized to implement the projects. The AIR Quality Semiconductor Sensor - MQ135 and PIR Motion Sensor were utilized as samples.

The projects' building phase was divided into four stages:

- ✓ First stage: virtual simulation on Tinkercad.com;
- ✓ Second stage: construction of a real circuit using the Arduino UNO interface;
- ✓ Third stage: integration of the Arduino IDE, Arduino Code, and Arduino UNO;
- ✓ Fourth stage: actual implementation of the project and experiment.

The experiments carried out under actual settings attested to the projects' successful implementation. Using the links below, you may watch project videos:

PIR Motion Sensor

<https://drive.google.com/file/d/1Y3iIBiP82Cr1BSk6I9s2S89BkE61AMD/view?usp=sharing>

AIR Quality Sensor MQ135

<https://drive.google.com/file/d/1BsiLVdjXCSoTpijdss2bljTT3ouO0Bw6/view?usp=sharing>

The projects demonstrated two things to us: 1) the levels of air pollution, and 2) instances of movement in a small area, which were captured by turning on the appropriate colored lights, by signaling and displaying messages on the monitor.

The projects discussed in the paper will assist interested readers in carrying out research projects, creating models of various sensor-oriented "smart" devices, and using their prototypes in actual settings for process automation.

**V. References**

- [1]. Zakariashvili M., (7.10. 2022). *ARDUINO - a tool for modeling and constructing STEM-type research projects*. IV International Scientific Conference "Education, Research, Practice". Iakob Gogebashvili Telavi State University. Faculty of Educational Science. Conference Proceedings. Pg. 95-98 UDK (733) 378.4 (479.221.2) (063); ISSN 2449-2337; 9772449233004; 04-44 . Published „Meridiani“. Tbilisi.
- [2]. Mosashvili I., Oniani S. (2016) *Fundamentals OF ARDUINO Programming*. Georgian Technical University. TEMPUS-JPCR-544091 Tbilisi.
- [3]. Tabatadze Z., Todua T. (2019) *Arduino. A Practical Guide for Beginners*. Georgian Technical University. Tbilisi.
- [4]. Nancy N. Heilbronner (2014) *International STEM Achievement: Not a Zero-Sum Game*. *Global Education Review*, 1 (4). 7-14. <https://files.eric.ed.gov/fulltext/EJ1055227.pdf> Retrieved 24/03/2023
- [5]. Geddes, Mark. (2016) *Arduino project handbook : 25 practical projects to get you started /* San Francisco. <http://lcn.loc.gov/2015033781>; <https://ss-valpovo.hr/wp-ontent/uploads/2020/01/arduinoprojecthandbook.pdf> Retrieved 24/03/2023
- [6]. STEM (Science, Technology, Engineering, Math ) Global Education Innovation Initiative . <https://globaled.gse.harvard.edu/stem-science-technology-engineering-math> retrieved 24/03/2023

Отримана в редакції 10.01.2023. Прийнята до друку 11.08.2023. Received 08 July 2023. Approved 15 August 2023. Available in Internet 12 September 2023