**Smoke Detection System**

Embadded Systems project by :

Crepcea Raluca-Maria

&

Maier Bianca-Anamaria

Group 2.2, 3rd year, CTI-Engleza

UPT

**Contents**

INTRODUCTION------------------------------------------------------------ 3

STATEMENT AND BRIEF EXPLANATION----------------------------------- 3

DEVELOPMENT BOARD DESCRIPTION ----------------------------------- 4

SYSTEM ARCHITECTURE -------------------------------------------------- 5

MICROCONTROLLER MODULES INVOLVED ------------------------------ 8

SENSORS AND ACTUATORS DESCRIPTION-------------------------------- 8

PROGRAMS, COMMENTS, AND DESCRIPTION---------------------------- 11

BIBLIOGRAPHY AND SOURCES OF INFORMATION------------------------ 14

**INTRODUCTION**

This documentation outlines the creation and functionality of a smoke detection system using a Raspberry Pi 3B+. The system utilizes a smoke sensor, LEDs, a buzzer, a relay, and a water pump to detect and respond to the presence of smoke, ensuring safety through early warning alerts and automatic fire suppression. Our aim is to demonstrate a seamless integration of hardware and software, allowing users to notice if smoke has been detected in their vicinity by emitting an alert sound, showing a red signal, and activating a water pump.

**STATEMENT AND BRIEF EXPLANATION**

The project involves designing a smoke detection system that monitors air quality for smoke presence and triggers alerts and responses when smoke is detected. This system is essential for early warning in case of fire, potentially saving lives and property. With the Raspberry Pi as the central control unit, we have connected an RGB LED, a buzzer, a relay, and a water pump to a breadboard. Additionally, a Smoke Detector using a Gas Sensor has been integrated into the system to enhance the interactive aspect of the project.

The system continuously monitors the environment for smoke using a smoke sensor. When smoke is detected, the system:

* Activates a red LED to visually indicate the presence of smoke.
* Generates an audible alarm using a buzzer.
* Activates a relay to start the water pump for fire suppression.
* Deactivates the alarm, switches to a green LED, and turns off the water pump when the smoke is no longer detected.

**DEVELOPMENT BOARD DESCRIPTION (components)**

The project uses a Raspberry Pi, a versatile development board based on the ARM architecture. The Raspberry Pi is known for its ease of use, extensive community support, and compatibility with various sensors and actuators.

Raspberry Pi 3B+



General Characteristics:

* CPU: 1.2 GHz 64-bit quad-core ARM Cortex-A53
* RAM: 1 GB LPDDR2
* Full-size HDMI
* Networking: Gigabit Ethernet, 2.4GHz 802.11n wireless
* Bluetooth: Bluetooth 4.2
* GPIO: 40-pin GPIO header
* 5V/2.5A micro-USB power supply

Peculiarities:

* Built-in Wi-Fi and Bluetooth for wireless connectivity.
* HDMI output for display connection.
* USB ports for peripheral devices.

Microcontroller

The Raspberry Pi is equipped with Broadcom BCM2837B0, Cortex-A53 (ARMv8) 64-bit SoC @ 1.4GHz, providing sufficient computational power for real-time sensor monitoring and control operations.

Peculiarities:

The Raspberry Pi features:

* Multiple GPIO pins for interfacing with sensors and actuators.
* Integrated support for Python programming, which simplifies the development process.
* Networking capabilities for remote monitoring and control if need

**SYSTEM ARCHITECTURE**

The hardware setup involves connecting the smoke sensor, RGB LED, relay, water pump, and buzzer to the Raspberry Pi. The schematic below illustrates the connections:

**Description**

The GPIO pins on the Raspberry Pi are utilized to interface with the smoke sensor, LEDs, and buzzer. The GPIO library in Python is used to control these pins:

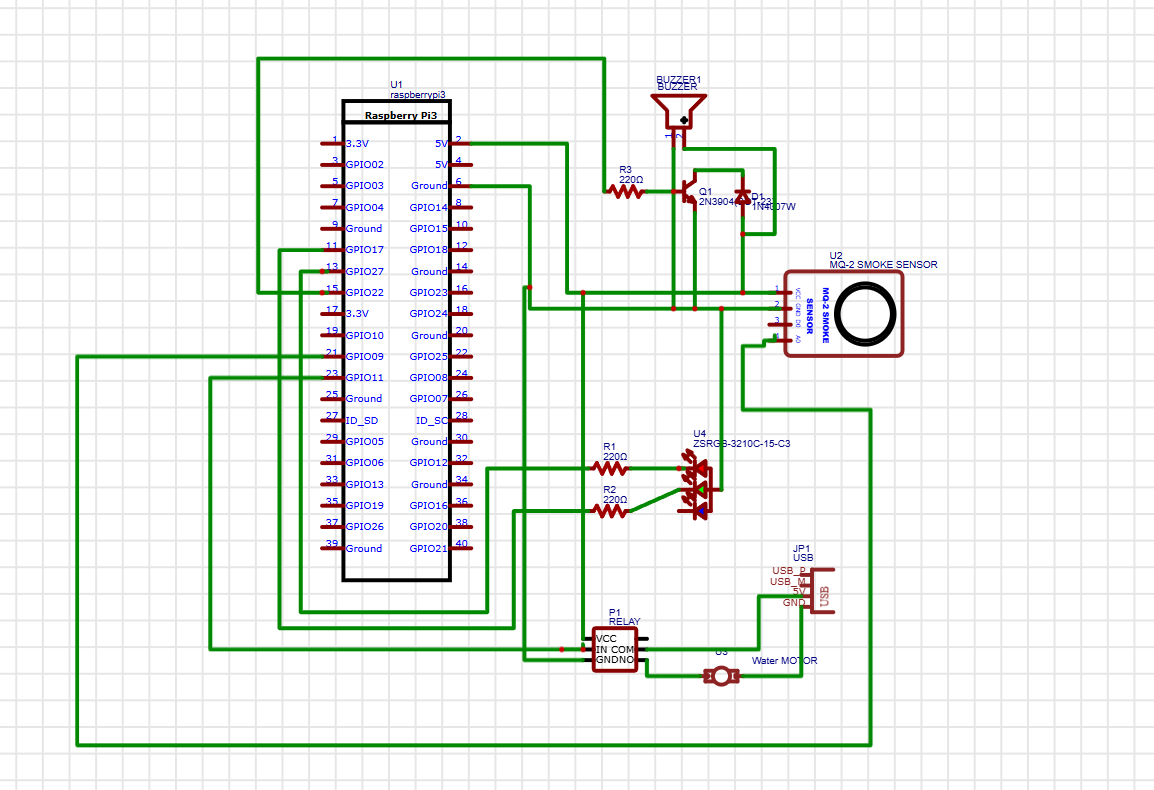
**GPIO 21**: Reads smoke sensor data.

**GPIO 11**: Controls red LED.

**GPIO 13**: Controls green LED.

**GPIO 15**: Controls buzzer.

**GPIO 23**: Controls relay.



**For Raspberry Pi 3B+:**

**We connected:**

* GPIO 21 (BOARD pin 40): Connected to the data pin of the Smoke Sensor.
* GPIO 11 (BOARD pin 11): Connected to the Red pin of the RGB LED (with a current-limiting resistor).
* GPIO 13 (BOARD pin 33): Connected to the Green pin of the RGB LED (with a current-limiting resistor).
* GPIO 15 (BOARD pin 10): Connected to the positive terminal of the Buzzer.
* GPIO 23 (BOARD pin 16): Connected to the control pin of the Relay.
* 5V Power Pin: Provides power to the VCC rail on the breadboard.
* GND Pin: Connected to the ground rail on the breadboard.

**Breadboard:**

Provides a platform for connecting components. Connects to the Raspberry Pi through various GPIO pins and power rails.



For the Breadboard we connected.

* VCC Rail: Connected to the 5V power pin of the Raspberry Pi.
* GND Rail: Connected to the GND pin of the Raspberry Pi.
* Smoke Sensor: VCC and GND connected to the corresponding rails on the breadboard. Data pin connected to GPIO 21 (BOARD pin 40) on the Raspberry Pi.
* RGB LED:
  + Red pin connected to GPIO 11 (BOARD pin 11) on the Raspberry Pi through a current-limiting resistor.
  + Green pin connected to GPIO 13 on the Raspberry Pi through a current-limiting resistor.
  + Common cathode connected to the GND rail.
* Buzzer: Positive terminal connected to GPIO 15 (BOARD pin 10) on the Raspberry Pi. Negative terminal connected to the GND rail.
* Relay: Control pin connected to GPIO 23 (BOARD pin 16) on the Raspberry Pi. VCC and GND connected to the corresponding rails on the breadboard.
* Water Pump: Connected to the normally open (NO) contacts of the relay.

**MICROCONTROLLER MODULES INVOLVED**

**GPIO Pins**

The Raspberry Pi's GPIO pins are used to interface with the smoke sensor, LEDs, and buzzer. The following pins are configured:

* Output Pins: Red LED (11), Green LED (13), Buzzer (15), Relay (23)
* Input Pin: Smoke Sensor (21)

**Interrupt System**

An interrupt system is implemented to detect changes in the smoke sensor's state, triggering the appropriate response (LEDs, buzzer and water pump) promptly.

**SENSORS AND ACTUATORS DESCRIPTION**

Buzzer



Operating Principle: Generates sound through rapid toggling of the GPIO pin.

Interface Type: Digital output (controlled by GPIO pin).

Smoke Sensor



* Type: MQ-2
* Analog output voltage: 0-5V.
* Block Diagram:

A black screen with white text

Description automatically generated

* D0 (Digital Output): Outputs a digital signal when the gas concentration exceeds a certain threshold. Connects to a GPIO pin on the microcontroller.

The MQ-2 sensor operates on the principle of surface adsorption and resistance variation. When the sensor is exposed to smoke or gas, the adsorbed gases cause a change in the resistance of the sensor material. This change in resistance is then converted into a corresponding voltage signal, which can be read by the microcontroller.

RBG LED



An RGB LED is a compact electronic component that combines red, green, and blue light-emitting diodes into a single package. It enables the generation of a broad spectrum of colors by independently adjusting the intensity of each primary color.

* Block Diagram:

A black screen with white text

Description automatically generated

WATER PUMP



* Operating Principle: Pumps water when powered.
* Interface Type: Controlled via the relay.
* Working Voltage: 4v-12V
* Working Current: 0.8A
* Motor Diameter: 27mm

In our project the pump is powered externally by a USB cable.

RELAY



* **Operating Principle:** Acts as a switch to control high-power devices using a low-power signal from the Raspberry Pi.
* **Interface Type:** Digital output (controlled by GPIO pin).
* **Purpose:** Used to control the water pump.

**PROGRAMS, COMMENTS, AND DESCRIPTION**

A screenshot of a computer

Description automatically generated

A screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated

**Justification for Programming Features**

* Loops:

The while True loop keeps the program running indefinitely, which is necessary for continuous monitoring of the flame sensor.

The for loop in the ‘generate\_sound’ function toggles the buzzer pin to produce sound.

* Conditional Branches:

The if-else statements in the callback function check the status of the flame sensor and respond by turning LEDs on/off and generating sound if a flame is detected.

Event Detection:

* GPIO.add\_event\_detect and GPIO.add\_event\_callback are used for efficient and responsive detection of changes in the flame sensor's output, ensuring timely execution of the callback function.

**Level of Involvement in Modifications**

* RPi.GPIO Library:

Obtained From: The RPi.GPIO library is an external library available through the Python Package Index (PyPI). It can be installed using:

**sudo pip install RPi.GPIO**

Modifications: No modifications were made to the RPi.GPIO library itself. The library was used as-is to control the GPIO pins on the Raspberry Pi.

* time Library:

Obtained From: The time module is part of Python’s standard library and does not require separate installation.

Modifications: No modifications were made to the time library. The library was used as-is for handling delays and sleep functionality in the code.

**BIBLIOGRAPHY AND SOURCES OF INFORMATION**

* [raspberry-gpio-python / Wiki / BasicUsage (sourceforge.net)](https://sourceforge.net/p/raspberry-gpio-python/wiki/BasicUsage/)
* [Raspberry Pi Documentation](https://www.raspberrypi.com/documentation/)
* [time — Time access and conversions — Python 3.12.3 documentation](https://docs.python.org/3/library/time.html)
* [Smoke Detector using Gas Sensor | Arduino Project Hub](https://projecthub.arduino.cc/srajandikshit147/smoke-detector-using-gas-sensor-ecdee6)
* [Interfacing RGB Led with Arduino | Arduino Project Hub](https://projecthub.arduino.cc/semsemharaz/interfacing-rgb-led-with-arduino-b59902)
* [USE a BUZZER MODULE (PIEZO SPEAKER) USING ARDUINO UNO | Arduino Project Hub](https://projecthub.arduino.cc/SURYATEJA/use-a-buzzer-module-piezo-speaker-using-arduino-uno-cf4191)