



AUTOMATIC DOOR OPENING SYSTEM BY SENSING THE HUMAN BODY TEMPERATURE

Dr.S.Rathinavel^{1*}, Dr.G.Balaji¹, R.Gowsalya², Anjali Kumari²

¹*Professor Department of EEE, Paavai Engineering College, Namakkal, Tamil Nadu, India.*

²*Student Department of EEE, Paavai Engineering College, Namakkal, Tamil Nadu, India.*

ABSTRACT

COVID-19 pandemic has a catastrophic consequence globally since its first case was detected in December 2019, with an aggressive spread. Safety critical mobile robotic system that provides a complete diagnostic test to check whether an individual is infected by Covid-19 or not. We introduce an affordable IoT-based solution aiming to increase COVID-19 indoor safety, covering several relevant aspects: contactless temperature sensing. Contactless temperature sensing subsystem relies on Arduino Uno and display in LCD. Those who have high temperature, Gate will not be Opening.

Keywords—*Detection, Microcontroller, IoT.*

1.1 INTRODUCTION

Since the last days of the previous year, the occurrence of novel infectious flu-alike respiratory disease COVID-19 caused by SARS-Cov-2 virus (also known as corona virus) has affected almost every aspect of people's lives globally. First, it was discovered in China, but spread quickly to other continents in just few weeks. According to, until July 11th, 2020, the total number of identified cases was 12,653,451, while taking 563,517 lives worldwide. Common symptoms of corona virus disease include fever, tiredness, sore throat, nasal congestion, loss of taste and smell. In most cases, it is transmitted directly (person to person) through respiratory droplets, but also indirectly via surfaces. Incubation period could be quite long and varies (between 14 and 27 days in extreme cases). Furthermore, even asymptomatic persons (almost 45% of cases) can spread the disease making the situation even worse. Therefore, the usage of face masks and sanitizers has shown positive results when it comes to disease spread reduction. However, the crucial problem is the lack of approved vaccine and medication. Due to these facts, many protection and safety measures were taken by governments in order to reduce the disease spread, such as obligatory indoor mask wearing, social distancing, quarantine, self-isolation, limiting citizens' movement within country borders and abroad, often together with prohibition and cancellation of huge public events and gatherings.

In this paper, cost-effective IoT-based system aiming to help organizations respect the COVID-19 safety rules and guidelines in order to reduce the disease spread is presented. We focus on most common indoor measures - people with high body temperature should stay at home, wearing mask is obligatory and distance



between persons should be at least 1.5-2 meters. For the first scenario, Arduino Uno microcontroller board with contactless temperature sensor is used, while we rely on Raspberry Pi2 single-board computer equipped with camera making use of computer vision techniques for other two scenarios. We decided to use these devices due to their small size and affordability.

1.2 LITERATURE SURVEY

Adrian Covaci et.al (2020) How can airborne transmission of COVID-19 indoors be minimized: During the rapid rise in COVID-19 illnesses and deaths globally, and notwithstanding recommended precautions, questions are voiced about routes of transmission for this pandemic disease. Inhaling small airborne droplets is probable as a third route of infection, in addition to more widely recognized transmission via larger respiratory droplets and direct contact with infected people or contaminated surfaces. While uncertainties remain regarding the relative contributions of the different transmission pathways, we argue that existing evidence is sufficiently strong to warrant engineering controls targeting airborne transmission as part of an overall strategy to limit infection risk indoors. Appropriate building engineering controls include sufficient and effective ventilation, possibly enhanced by particle filtration and air disinfection, avoiding air recirculation and avoiding overcrowding. Often, such measures can be easily implemented and without much cost, but if only they are recognized as significant in contributing to infection control goals.

John Davies et.al (2006) Semantic Web Technologies Trends and Research in Ontology-based Systems: Information technology has a surprising way of changing our culture radically—often in ways unimaginable to the inventors. When Gutenberg developed moveable type in the middle of the fifteenth century, his primary goal was to develop a mechanism to speed the printing of Bibles. Gutenberg probably never thought of his technology in terms of the general dissemination of human knowledge via printed media. He never planned explicitly for printing presses to democratize the ownership of knowledge and to take away the monopoly on the control of information that had been held previously by the Church—which initially lacked Gutenberg’s technology, but which had at its disposal the vast numbers of dedicated personnel needed to store, copy, and distribute books in a totally manual fashion. Gutenberg sought a better way to produce Bibles, and as a result changed fundamentally the control of knowledge in Western society. Within a few years, anyone who owned a printing press could distribute knowledge widely to anyone willing to read it.

Michael Jones et.al (2001) Rapid Object Detection using a Boosted Cascade of Simple Features: This paper describes a machine learning approach for visual object detection which is capable of processing images extremely rapidly and achieving high detection rates. This work is distinguished by three key contributions. The first is the introduction of a new image representation called the “Integral lineage” which allows the features used by our detector to be computed very quickly. The second is a learning algorithm, based on AdaBoost, which selects a small number of critical visual features from a larger set and yields extremely efficient classifiers. The third contribution is a method for combining increasingly more complex classifiers in a “cascade” which allows background regions of the image to be quickly discarded while spending more computation on promising object-like regions.



Nenad Petrović¹ and Đorđe Kocić (2020) Data-driven Framework for Energy-Efficient Smart Cities: Energy management is one of the greatest challenges in smart cities. Moreover, the presence of autonomous vehicles makes this task even more complex. In this paper, we propose a data-driven smart grid framework which aims to make smart cities energy-efficient focusing on two aspects: energy trading and autonomous vehicle charging. The framework leverages deep learning, linear optimization, semantic technology, domain-specific modelling notation, simulation and elements of relay protection. The evaluation of deep learning module together with code generation time and energy distribution cost reduction performed within the simulation environment also presented in this paper are given. According to the results, the achieved energy distribution cost reduction varies and depends from case to case.

Pan Zhai a and Yanbing Ding (2020) The epidemiology, diagnosis and treatment of COVID-19: In December 2019, the outbreak of the novel corona virus disease (COVID-19) in China spread worldwide, becoming an emergency of major international concern. SARS-CoV-2 infection causes clusters of severe respiratory illness similar to severe acute respiratory syndrome corona virus. Human-to-human transmission via droplets, contaminated hands or surfaces has been described, with incubation times of 2-14 days. Early diagnosis, quarantine, and supportive treatments are essential to cure patients. This paper reviews the literature on all available information about the epidemiology, diagnosis, isolation and treatments of COVID-19. Treatments, including antiviral agents, chloroquine and hydroxyl chloroquine, corticosteroids, antibodies, convalescent plasma transfusion and vaccines, are discussed in this article. In addition, registered trials investigating treatment options for COVID-19 infection are listed.

Patrick Dawson and Elizabeth M. Rabold (2020) Loss of Taste and Smell as Distinguishing Symptoms of COVID-19: In a household study, loss of taste and/or smell was the fourth most reported symptom (26/42; 62%) among COVID-19 case-patients and had the highest positive predictive value (83%; 95% CI: 55–95%) among household contacts. Olfactory and taste dysfunctions should be considered for COVID-19 case identification and testing prioritization. The global corona virus disease 2019 (COVID-19) pandemic caused by severe acute respiratory syndrome corona virus 2 (SARS-CoV-2) has prompted robust public health investigations to characterize the disease course. Early published reports identified fever, cough, and shortness of breath as predominant symptoms of COVID-19 (1,2), and these classic symptoms have been used for case identification and testing prioritization.

Talib Dbouk and Dimitris Drikakis (2020) On respiratory droplets and face masks: Face mask filters—textile, surgical, or respiratory—are widely used in an effort to limit the spread of airborne viral infections. Our understanding of the droplet dynamics around a face mask filter, including the droplet containment and leakage from and passing through the cover, is incomplete. We present a fluid dynamics study of the transmission of respiratory droplets through and around a face mask filter. By employing multiphase computational fluid dynamics in a fully coupled Eulerian–Lagrangian framework, we investigate the droplet dynamics induced by a mild coughing incident and examine the fluid dynamics phenomena affecting the mask efficiency. The model takes into account turbulent dispersion forces, droplet phase-change, evaporation, and breakup in addition to the



droplet–droplet and droplet–air interactions. The model mimics real events by using data, which closely resemble cough experiments.

Thushara Galbadage1 and Brent M. Peterson1 (2020) Does COVID-19 Spread Through Droplets Alone: The world is in the middle of a historic public health crisis. As of March 30, 2020, over a third of the population in the United States were under “stay at home” orders given by state governors to protect the vulnerable and the unexposed. Unprecedented steps have been taken by governments globally to contain the novel corona virus disease 2019 (COVID-19), a rapidly spreading pandemic. This has resulted in more than 690,000 cases and over 33,000 deaths worldwide (Supplementary Table 1). The index case of the disease, caused by the Severe Acute Respiratory Syndrome Coronavirus-2 (SARS-CoV-2) was identified more than 3months ago. Since then, public health authorities worldwide have taken aggressive measures to blunt the exponential spread of this coronavirus. Furthermore, several nations, including Italy, Spain, and France, have imposed nationwide lockdown measures to enforce social distancing to further prevent the spread of COVID-19 in their respective countries.

Yang Song et.al (2020) COVID-19 Treatment: Close to a Cure? – A Rapid Review of Pharmacotherapies for the Novel Coronavirus: Currently, there is no approved therapy for COVID-19. The World Health Organization therefore endorse supportive care only. However, frontline clinicians and researchers have been experimenting with several virus-based and host-based therapeutics since the outbreak in China. China's National Health Commission has issued the first COVID-19 Treatment Guideline with therapy suggestions (7th edition attached) which inspired following clinical studies worldwide. Major therapeutics are evaluated in this review. Key evidence from in vitro researches, animal models and clinical researches in emerging coronaviruses are examined. Antiviral therapies remdesivir, lopinavir/ritonavir and umifenovir, if considered, could be initiated before the peak of viral replication for optimal outcomes. Ribavirin may be beneficial as an add-on therapy and is ineffective as a monotherapy. Corticosteroids use should be limited to indicating comorbidities. IVIG is not recommended due to lack of data in COVID-19. Xuebijing may benefit patients with complications of bacterial pneumonia or sepsis. The efficacy of interferon is unclear due to conflicting outcomes in coronavirus studies.

Zhongyuan Wang a Guangcheng Wang (2020) Masked Face Recognition Dataset and Application: In order to effectively prevent the spread of COVID- 19 virus, almost everyone wears a mask during coronavirus epidemic. This almost makes conventional facial recognition technology ineffective in many cases, such as community access control, face access control, facial attendance, facial security checks at train stations, etc. Therefore, it is very urgent to improve the recognition performance of the existing face recognition technology on the masked faces. Most current advanced face recognition approaches are designed based on deep learning, which depend on a large number of face samples. However, at present, there are no publicly available masked face recognition datasets. To this end, this work proposes three types of masked face datasets, including Masked Face Detection Dataset (MFDD), Real-world Masked Face Recognition Dataset (RMFRD) and Simulated Masked Face Recognition Dataset (SMFRD).

2.1 OVERVIEW

A dataset for masked face recognition is introduced and its application by different algorithms in context of campus and enterprise corona virus prevention discussed. Moreover, a high-accuracy method for facial mask detection using semantic segmentation based on fully convolutional networks, gradient descent and binomial cross entropy was presented. However, performance-wise, it is too heavy for low-power IoT devices, such as Raspberry Pi. On the other side, a state model-based solution for face mask detection relying on Viola-Jones algorithm in context of ATM center security was described.

2.1.1 DISADVANTAGE

- No symptoms persons also sometimes affected by virus.
- Identification of that person is too difficult.

2.2 EXISTING BLOCK DIAGRAM

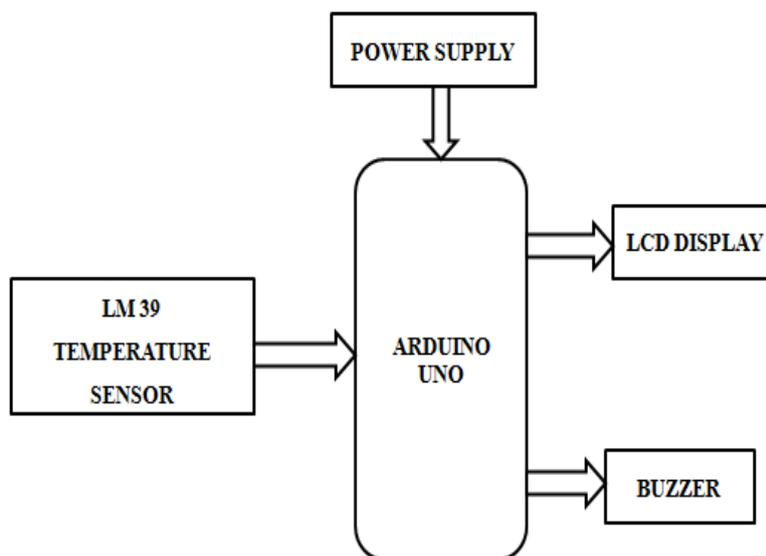


Figure 2.1 Block Diagram of Existing System

3.1 INTRODUCTION

We focus on most common indoor measures school, College, office people with high body temperature should stay at home, and otherwise the gate will open. Automatically check the body temperature through contactless temperature sensing, if temperature high gate will not open with help of servo motor. At the same time Buzzer will be ON. All the parameters of data will displayed through lcd display.

3.2 PROPOSED BLOCK DIAGRAM

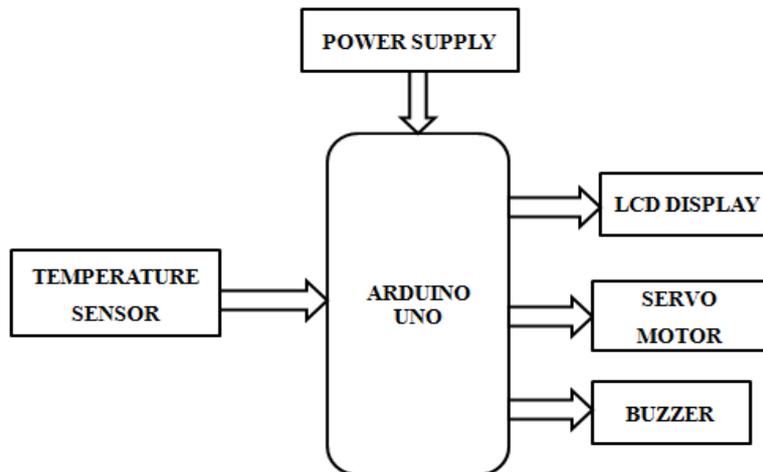


Figure 3.1 Block Diagram of Proposed System

3.3 ADVANTAGE

- Person identification above normal temperature.
- To reduce the disease spread is presented.

3.4 APPLICATION

- Safety critical mobile robotic system that provides a complete diagnostic test to check whether an individual is infected by Covid-19 or not.
- Secure from the virus us wearing masks and communication through social media.

4.1 ARDUINO IDE

The Arduino integrated development environment (IDE) is a cross-platform application (for Windows, macOS, Linux) that is written in the programming language Java. It is used to write and upload programs to Arduino board.

The source code for the IDE is released under the GNU General Public License, version 2. The Arduino IDE supports the languages C and C++ using special rules of code structuring.^[4] The Arduino IDE supplies a software library from the Wiring project, which provides many common input and output procedures. User-written code only requires two basic functions, for starting the sketch and the main program loop, that are compiled and linked with a program stub main() into an executable cyclic executive program with the GNU tool chain, also included with the IDE distribution.^[5] The Arduino IDE employs the program avrdude to convert the executable code into a text file in hexadecimal encoding that is loaded into the Arduino board by a loader program in the board's firmware.

Arduino is an open-source electronics platform based on easy-to-use hardware and software. Arduino boards are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an



output - activating a motor, turning on an LED, publishing something online. You can tell your board what to do by sending a set of instructions to the microcontroller on the board

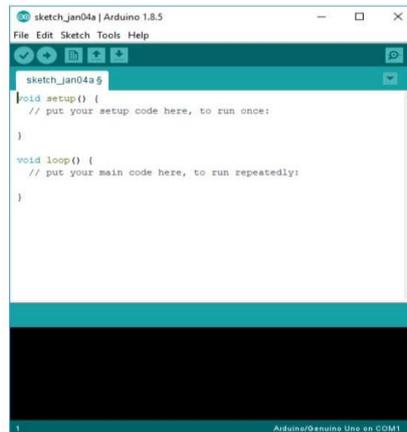


Figure 4.1 Arduino IDE

The Arduino IDE is incredibly minimalistic, yet it provides a near-complete environment for most Arduino-based projects. The top menu bar has the standard options, including “File” (new, load save, etc.), “Edit” (font, copy, paste, etc.), “Sketch” (for compiling and programming), “Tools” (useful options for testing projects), and “Help”. The middle section of the IDE is a simple text editor that where you can enter the program code. The bottom section of the IDE is dedicated to an output window that is used to see the status of the compilation, how much memory has been used, any errors that were found in the program, and various other useful messages.

Projects made using the Arduino are called sketches, and such sketches are usually written in a cut-down version of C++ (a number of C++ features are not included). Because programming a microcontroller is somewhat different from programming a computer, there are a number of device-specific libraries (e.g., changing pin modes, output data on pins, reading analog values, and timers). This sometimes confuses users who think Arduino is programmed in an “Arduino language.” However, the Arduino is, in fact, programmed in C++. It just uses unique libraries for the device.

The Arduino Integrated Development Environment - or Arduino Software (IDE) - contains a text editor for writing code, a message area, a text console, a toolbar with buttons for common functions and a series of menus. It connects to the Arduino and Genuino hardware to upload programs and communicate with them.

Programs written using Arduino Software (IDE) are called **sketches**. These sketches are written in the text editor and are saved with the file extension `.ino`. The editor has features for cutting/pasting and for searching/replacing text. The message area gives feedback while saving and exporting and also displays errors. The console displays text output by the Arduino Software (IDE), including complete error messages and other information. The bottom righthand corner of the window displays the configured board and serial port. The toolbar buttons allow you to verify and upload programs, create, open, and save sketches, and open the serial monitor.



4.2 LIBRARIES

Libraries provide extra functionality for use in sketches, e.g. working with hardware or manipulating data. To use a library in a sketch, select it from the **Sketch > Import Library** menu. This will insert one or more **#include** statements at the top of the sketch and compile the library with your sketch. Because libraries are uploaded to the board with your sketch, they increase the amount of space it takes up. If a sketch no longer needs a library, simply delete its **#include** statements from the top of your code.

There is a list of libraries in the reference. Some libraries are included with the Arduino software. Others can be downloaded from a variety of sources or through the Library Manager. Starting with version 1.0.5 of the IDE, you do can import a library from a zip file and use it in an open sketch.

4.3 CONNECTING THE ARDUINO

Connecting an Arduino board to your PC is quite simple. On Windows:

1. Plug in the USB cable - one end to the PC, and one end to the Arduino board.
2. When prompted, select "Browse my computer for driver" and then select the folder to which you extracted your original Arduino IDE download.
3. You may receive an error that the board is not a Microsoft certified device - select "Install anyway."
4. Your board should now be ready for programming.

When programming your Arduino board it is important to know what COM port the Arduino is using on your PC. On Windows, navigate to Start->Devices and Printers, and look for the Arduino. The COM port will be displayed underneath.

Alternatively, the message telling you that the Arduino has been connected successfully in the lower-left hand corner of your screen usually specifies the COM port is it using.

4.4 PREPARING THE BOARD

Before loading any code to your Arduino board, you must first open the IDE. Double click the Arduino .exe file that you downloaded earlier. A blank program, or "sketch," should open.

The Blink example is the easiest way to test any Arduino board. Within the Arduino window, it can be found under File->Examples->Basics->Blink.

Before the code can be uploaded to your board, two important steps are required.

1. Select your Arduino from the list under Tools->Board. The standard board used in RBE 1001, 2001, and 2002 is the Arduino Mega 2560, so select the "Arduino Mega 2560 or Mega ADK" option in the dropdown.
2. Select the communication port, or COM port, by going to Tools->Serial Port.

If you noted the COM port your Arduino board is using, it should be listed in the dropdown menu. If not, your board has not finished installing or needs to be reconnected.



4.5 LOADING CODE

The upper left of the Arduino window has two buttons: A checkmark to Verify your code, and a right-facing arrow to Upload it. Press the right arrow button to compile and upload the Blink example to your Arduino board.

The black bar at the bottom of the Arduino window is reserved for messages indicating the success or failure of code uploading. A "Completed Successfully" message should appear once the code is done uploading to your board. If an error message appears instead, check that you selected the correct board and COM port in the Tools menu, and check your physical connections.

If uploaded successfully, the LED on your board should blink on/off once every second. Most Arduino boards have an LED prewired to pin 13.

<https://forms.gle/3Q143c3nx9vyMbW79t> is very important that you do not use pins 0 or 1 while loading code. It is recommended that you do not use those pins ever.

Arduino code is loaded over a serial port to the controller. Older models use an FTDI chip which deals with all the USB specifics. Newer models have either a small AVR that mimics the FTDI chip or a built-in USB-to-serial port on the AVR micro-controller itself.

4.6 PRODUCT MODULES

The Proteus Design Suite is a Windows application for schematic capture, simulation, and PCB (Printed Circuit Board) layout design. It can be purchased in many configurations, depending on the size of designs being produced and the requirements for microcontroller simulation. All PCB Design products include an auto-router and basic mixed mode SPICE simulation capabilities.

4.6.1 Schematic Capture

Schematic capture in the Proteus Design Suite is used for both the simulation of designs and as the design phase of a PCB layout project. It is therefore a core component and is included with all product configurations.

4.6.2 Microcontroller Simulation

The micro-controller simulation in Proteus works by applying either a hex file or a debug file to the microcontroller part on the schematic. It is then co-simulated along with any analog and digital electronics connected to it. This enables its use in a broad spectrum of project prototyping in areas such as motor control, temperature control and user interface design.^[6] It also finds use in the general hobbyist community and, since no hardware is required, is convenient to use as a training or teaching tool. Support is available for co-simulation of:

- Microchip Technologies PIC10, PIC12, PIC16, PIC18, PIC24, dsPIC33 Microcontrollers.
- Atmel AVR (and Arduino), 8051 and ARM Cortex-M3 Microcontrollers



- NXP 8051, ARM7, ARM Cortex-M0 and ARM Cortex-M3 Microcontrollers.
- Texas Instruments MSP430, PICCOLO DSP and ARM Cortex-M3 Microcontrollers.
- Parallax Basic Stamp, Freescale HC11, 8086 Microcontrollers.

PCB Design

The PCB Layout module is automatically given connectivity information in the form of a netlist from the schematic capture module. It applies this information, together with the user specified design rules and various design automation tools, to assist with error free board design. PCB's of up to 16 copper layers can be produced with design size limited by product configuration.

4.7 PROTEUS SIMULATIONS

Proteus's simulation feature. Many of the components in Proteus can be simulated. There are two options for simulating: Run simulator and advance frame by frame. The "Run simulator" option simulates the circuit in a normal speed (If the circuit is not heavy). "Advance frame by frame" option advances to next frame and waits till you click this button for the next time. This can be useful for debugging digital circuits. You can also simulate microcontrollers. The microcontrollers which can be simulated include PIC24, dsPIC33, 8051, Arduino, ARM7 based microcontrollers. You can download the compilers for Proteus or use different compiler and dump the hex files in the microcontroller in Proteus. You can even interact in real-time with the simulation using switches, resistors, LDRs, etc. There are even virtual voltmeter, ammeter, oscilloscope, logic analyzer, etc.

5.1 POWER SUPPLY CIRCUIT

Power supply is a reference to a source of electrical power. A device or system that supplies electrical or other types of energy to an output load or group of loads is called a power supply unit or PSU. The term is most commonly applied to electrical energy supplies, less often to mechanical ones, and rarely to others.

Power supplies for electronic devices can be broadly divided into linear and switching power supplies. The linear supply is a relatively simple design that becomes increasingly bulky and heavy for high current devices; voltage regulation in a linear supply can result in low efficiency. A switched-mode supply of the same rating as a linear supply will be smaller, is usually more efficient, but will be more complex.

5.2 LINEAR POWER SUPPLY

An AC powered linear power supply usually uses a transformer to convert the voltage from the wall outlet (mains) to a different, usually a lower voltage. If it is used to produce DC, a rectifier is used. A capacitor is used to smooth the pulsating current from the rectifier. Some small periodic deviations from smooth direct current will remain, which is known as ripple. These pulsations occur at a frequency related to the AC power frequency (for example, a multiple of 50 or 60 Hz).

5.3 TRANSFORMER

Transformers convert AC electricity from one voltage to another with little loss of power. Transformers work only with AC and this is one of the reasons why mains electricity is AC.

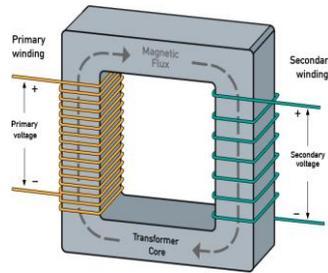


Figure 5.1 Transformer

Step-up transformers increase voltage, step-down transformers reduce voltage. Most power supplies use a step-down transformer to reduce the dangerously high mains voltage (230V in UK) to a safer low voltage.

The input coil is called the primary and the output coil is called the secondary. There is no electrical connection between the two coils; instead they are linked by an alternating magnetic field created in the soft-iron core of the transformer. The two lines in the middle of the circuit symbol represent the core.

Transformers waste very little power so the power out is (almost) equal to the power in. Note that as voltage is stepped down current is stepped up.

The ratio of the number of turns on each coil, called the turn's ratio, determines the ratio of the voltages. A step-down transformer has a large number of turns on its primary (input) coil which is connected to the high voltage mains supply, and a small number of turns on its secondary (output) coil to give a low output voltage.

5.4 BRIDGE RECTIFIER

A bridge rectifier can be made using four individual diodes, but it is also available in special packages containing the four diodes required. It is called a full-wave rectifier because it uses the entire AC wave (both positive and negative sections). 1.4V is used up in the bridge rectifier because each diode uses 0.7V when conducting and there are always two diodes conducting, as shown in the diagram below. Bridge rectifiers are rated by the maximum current they can pass and the maximum reverse voltage they can withstand (this must be at least three times the supply RMS voltage so the rectifier can withstand the peak voltages). Please see the DIODES page for more details, including pictures of bridge rectifiers.

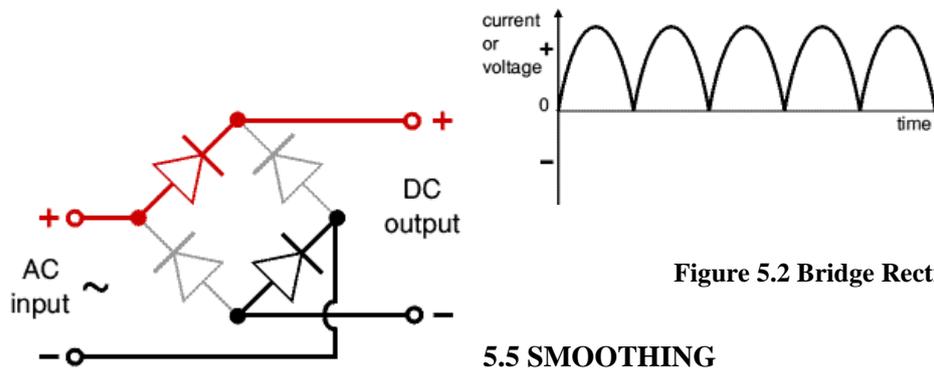


Figure 5.2 Bridge Rectifier

5.5 SMOOTHING

Smoothing is performed by a large value electrolytic capacitor connected across the DC supply to act as a reservoir, supplying current to the output when the varying DC voltage from the rectifier is falling. The diagram shows the unsmoothed varying DC (dotted line) and the smoothed DC (solid line). The capacitor charges quickly near the peak of the varying DC, and then discharges as it supplies current to the output.

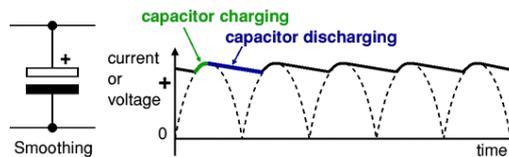


Figure 5.3 Smoothing

Note that smoothing significantly increases the average DC voltage to almost the peak value ($1.4 \times$ RMS value). For example 6V RMS AC is rectified to full wave DC of about 4.6V RMS (1.4V is lost in the bridge rectifier), with smoothing this increases to almost the peak value giving $1.4 \times 4.6 = 6.4V$ smooth DC.

Smoothing is not perfect due to the capacitor voltage falling a little as it discharges, giving a small ripple voltage. For many circuits a ripple which is 10% of the supply voltage is satisfactory and the equation below gives the required value for the smoothing capacitor. A larger capacitor will give fewer ripples. The capacitor value must be doubled when smoothing half-wave DC.

Smoothing Capacitor for 10% ripple, $C = 5 \cdot I_o / v_s \cdot f$

C = smoothing capacitance in farads (F)

I_o = output current from the supply in amps (A)

V_s = supply voltage in volts (V), this is the peak value of the unsmoothed DC

f = frequency of the AC supply in hertz (Hz), 50Hz in the UK.

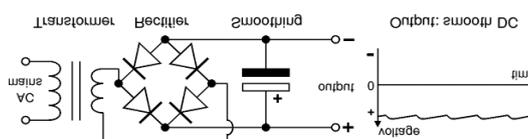


Figure 5.4 power supply circuit

The smooth DC output has a small ripple. It is suitable for most electronic circuits.

5.6 REGULATOR

Voltage regulator ICs are available with fixed (typically 5, 12 and 15V) or variable output voltages. They are also rated by the maximum current they can pass. Negative voltage regulators are available, mainly for use in dual supplies. Most regulators include some automatic protection from excessive current ('overload protection') and overheating ('thermal protection').

The LM78XX series of three terminal regulators is available with several fixed output voltages making them useful in a wide range of applications. One of these is local on card regulation, eliminating the distribution problems associated with single point regulation. The voltages available allow these regulators to be used in logic systems, instrumentation, Hi-Fi, and other solid state electronic equipment. Although designed primarily as fixed voltage regulators these devices can be used with external components to obtain adjustable voltages and current.

Many of the fixed voltage regulator ICs has 3 leads and look like power transistors, such as the 7805 +5V 1A regulator shown on the right. They include a hole for attaching a heat sink if necessary.

1. Positive regulator
 1. input pin
 2. ground pin
 3. output pin
2. Negative regulator
 1. ground pin
 2. input pin
 3. output pin

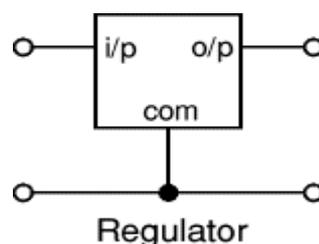


Figure 5.5 Regulator

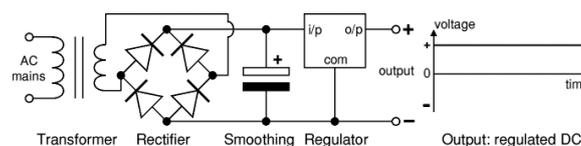


Figure 5.6 Transformer Rectifier Smoothing Regulator



The regulated DC output is very smooth with no ripple. It is suitable for all electronic circuits.

5.7 ARDUINO UNO

The Arduino UNO is an open-source microcontroller board based on the Microchip ATmega328P microcontroller developed by Arduino.cc. The board is equipped with sets of digital and analog input/output (I/O) pins that may be interfaced to various expansion boards (shields) and other circuits. The board has 14 Digital pins, 6 Analog pins, and programmable with the Arduino IDE (Integrated Development Environment) via a type B USB cable. It can be powered by a USB cable or by an external 9 volt battery, though it accepts voltages between 7 and 20 volts. It is also similar to the Arduino Nano and Leonardo. The hardware reference design is distributed under a Creative Commons Attribution Share-Alike 2.5 license and is available on the Arduino website. Layout and production files for some versions of the hardware are also available. "Uno" means one in Italian and was chosen to mark the release of Arduino Software (IDE) 1.0. The Uno board and version 1.0 of Arduino Software (IDE) were the reference versions of Arduino, now evolved to newer releases. The Uno board is the first in a series of USB Arduino boards, and the reference model for the Arduino platform. The ATmega328 on the Arduino Uno comes preprogrammed with a boot loader that allows uploading new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol. The Uno also differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it uses the Atmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter

The Arduino project started at the Interaction Design Institute Ivrea (IDII) in Ivrea, Italy. At that time, the students used a BASIC Stamp microcontroller at a cost of \$100, a considerable expense for many students. In 2003 Hernando Barragán created the development platform Wiring as a Master's thesis project at IDII, under the supervision of Massimo Banzi and Casey Reas, who are known for work on the Processing language. The project goal was to create simple, low-cost tools for creating digital projects by non-engineers. The Wiring platform consisted of a printed circuit board (PCB) with an ATmega168 microcontroller, an IDE based on Processing and library functions to easily program the microcontroller. In 2003, Massimo Banzi, with David Mellis, another IDII student, and David Cuartielles, added support for the cheaper ATmega8 microcontroller to Wiring. But instead of continuing the work on Wiring, they forked the project and renamed it Arduino. Early arduino boards used the FTDI USB-to-serial driver chip and an ATmega168. The Uno differed from all preceding boards by featuring the ATmega328P microcontroller and an ATmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter.

5.8 SPECIFICATION

- Microcontroller: Microchip ATmega328P
- Operating Voltage: 5 Volt
- Input Voltage: 7 to 20 Volts
- Digital I/O Pins: 14 (of which 6 provide PWM output)



- Analog Input Pins: 6
- DC Current per I/O Pin: 20 mA
- DC Current for 3.3V Pin: 50 mA
- Flash Memory: 32 KB of which 0.5 KB used by boot loader
- SRAM: 2 KB
- EEPROM: 1 KB
- Clock Speed: 16 MHz
- Length: 68.6 mm
- Width: 53.4 mm
- Weight: 25 g

5.9 ARDUINO/GENUINO

The Arduino/Genuino Uno has a number of facilities for communicating with a computer, another Arduino/Genuino board, or other microcontrollers. The ATmega328 provides UART TTL (5V) serial communication, which is available on digital pins 0 (RX) and 1 (TX). An ATmega16U2 on the board channels this serial communication over USB and appears as a virtual com port to software on the computer. The 16U2 firmware uses the standard USB COM drivers, and no external driver is needed. However, on Windows, a .inf file is required. The Arduino Software (IDE) includes a serial monitor which allows simple textual data to be sent to and from the board. The RX and TX LEDs on the board will flash when data is being transmitted via the USB-to-serial chip and USB connection to the computer (but not for serial communication on pins 0 and 1). A Software Serial library allows serial communication on any of the Uno's digital pins.



Figure 5.7

5.10 PINS GENERAL PIN FUNCTIONS

- **LED:** There is a built-in LED driven by digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.
- **VIN:** The input voltage to the Arduino/Genuino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.



- **5V:** This pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack (7 - 20V), the USB connector (5V), or the VIN pin of the board (7-20V). Supplying voltage via the 5V or 3.3V pins bypasses the regulator, and can damage the board.
- **3V3:** A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- **GND:** Ground pins.
- **IOREF:** This pin on the Arduino/Genuino board provides the voltage reference with which the microcontroller operates. A properly configured shield can read the IOREF pin voltage and select the appropriate power source or enable voltage translators on the outputs to work with the 5V or 3.3V.
- **Reset:** Typically used to add a reset button to shields which block the one on the board.

5.11 SPECIAL PIN FUNCTIONS

Each of the 14 digital pins and 6 Analog pins on the Uno can be used as an input or output, using `pinMode()`, `digitalWrite()`, and `digitalRead()` functions. They operate at 5 volts. Each pin can provide or receive 20 mA as recommended operating condition and has an internal pull-up resistor (disconnected by default) of 20-50k ohm. A maximum of 40mA is the value that must not be exceeded on any I/O pin to avoid permanent damage to the microcontroller. The Uno has 6 analog inputs, labeled A0 through A5, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though it is possible to change the upper end of their range using the AREF pin and the `analogReference()` function.

In addition, some pins have specialized functions:

- **Serial / UART:** pins 0 (RX) and 1 (TX). Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.
- **External Interrupts:** pins 2 and 3. These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value.
- **PWM (Pulse Width Modulation):** 3, 5, 6, 9, 10, and 11 Can provide 8-bit PWM output with the `analogWrite()` function.
- **SPI (Serial Peripheral Interface):** 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). These pins support SPI communication using the SPI library.
- **TWI (Two Wire Interface) / I²C:** A4 or SDA pin and A5 or SCL pin. Support TWI communication using the Wire library.
- **AREF (Analog REFERENCE):** Reference voltage for the analog inputs.

5.12 LIQUID CRYSTAL DISPLAY

A liquid crystal display (LCD) is a flat panel display, electronic visual display, or video display that uses the light modulating properties of liquid crystals. Liquid crystals do not emit light directly. LCDs are available to display arbitrary images (as in a general-purpose computer display) or fixed images which can be displayed or hidden, such as preset words, digits, and 7-segment displays as in a digital clock. They use the

same basic technology, except that arbitrary images are made up of a large number of small pixels, while other displays have larger elements. An LCD is a small low cost display. It is easy to interface with a micro-controller because of an embedded controller (the black blob on the back of the board). This controller is standard across many displays (HD 44780) which means many micro-controllers (including the Arduino) have libraries that make displaying messages as easy as a single line of code.



Figure 5.8 LCD display unit

LCDs are used in a wide range of applications including computer monitors, televisions, instrument panels, aircraft cockpit displays, and signage. They are common in consumer devices such as video players, gaming devices, clocks, watches, calculators, and telephones, and have replaced cathode ray tube (CRT) displays in most applications. They are available in a wider range of screen sizes than CRT and plasma displays, and since they do not use phosphors, they do not suffer image burn-in. LCDs are, however, susceptible to image persistence.

5.13 MLX90614 TEMPERATURE SENSOR

The MLX90614 is a Contactless Infrared (IR) Digital Temperature Sensor that can be used to measure the temperature of a particular object ranging from -70°C to 382.2°C . The sensor uses IR rays to measure the temperature of the object without any physical contact and communicates to the microcontroller using the I2C protocol.

The key feature of MLX90614 is that it is a contactless IR temperature sensor with high accuracy. So it can be used in industries to measure the temperature of moving objects like a rotating motor shaft. Due to its high accuracy and precision, it is also used in a wide range of commercial, health care, and household applications like room temperature monitoring, body temperature measurement, etc.

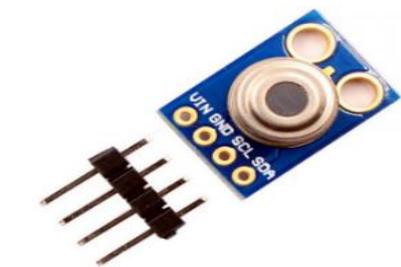


Figure 5.9 MLX90614 Temperature Sensor

The MLX90614 Temperature sensor is manufactured by a company called Melexis. The sensor is factory calibrated and hence it acts like a plug and play sensor module for speeding up development processes.

The MLX90614 consists of two devices embedded as a single sensor, one device acts as a sensing unit and the other device acts as a processing unit. The sensing unit is an Infrared Thermopile Detector called MLX81101 which senses the temperature and the processing unit is a Single Conditioning ASSP called MLX90302 which converts the signal from the sensor to digital value and communicates using I2C protocol. The MLX90302 has a low noise amplifier, 17-bit ADC and a powerful DSP which helps the sensor to have high accuracy and resolution.

The sensor requires no external components and can be directly interfaced with a microcontroller like Arduino. As you can see above the power pins (Vdd and Gnd) can be directly used to power the sensor, typically 5V can be used, but there are other versions of this sensor which can operate on 3.3V and 7V as well. The capacitor C1 is optional and is used to filter noise and provide optimum EMC. The signal pins (SCL and SDA) are used for I2C communication and can be connected directly to microcontroller operating on 5V logic.

The sensor is also sold as a module as you can see in the pinout image. But the sensor module is very similar to the sensor itself and does not have any additional components other than the sensor itself.

5.14 SPECIFICATION

- Operating Voltage: 3.6V to 5V (available in 3V and 5V version)
- Supply Current: 1.5mA
- Object Temperature Range: -70° C to 382.2°C
- Ambient Temperature Range: -40° C to 125°C
- Accuracy: 0.02°C
- Field of View: 80°
- Distance between object and sensor: 2cm-5cm (approx.)

5.15 SERVO MOTOR

The servo motor is most commonly used for high technology devices in the industrial applications like automation technology. It is a self-contained electrical device that rotates parts of machine with high efficiency and great precision. Moreover the output shaft of this motor can be moved to a particular angle. Servo motors are mainly used in home electronics, toys, cars, airplanes and many more devices. Thus this blog discusses the definition, types, mechanism, principle, working, controlling, and lastly the applications of a servo machine.



Figure 5.10 Servo Motor

A servo motor is a rotary actuator or a motor that allows for a precise control in terms of the angular position, acceleration, and velocity. Basically it has certain capabilities that a regular motor does not have. Consequently it makes use of a regular motor and pairs it with a sensor for position feedback.

5.16 BUZZER

A buzzer or beeper is a signalling device, usually electronic, typically used in automobiles, household appliances such as a microwave oven, or game shows. It most commonly consists of a number of switches or sensors connected to a control unit that determines if and which button was pushed or a preset time has lapsed, and usually illuminates a light on the appropriate button or control panel, and sounds a warning in the form of a continuous or intermittent buzzing or beeping sound. Initially this device was based on an electromechanical system which was identical to an electric bell without the metal gong (which makes the ringing noise). Often these units were anchored to a wall or ceiling and used the ceiling or wall as a sounding board.



Figure 5.11 Buzzer

Another implementation with some AC-connected devices was to implement a circuit to make the AC current into a noise loud enough to drive a loudspeaker and hook this circuit up to a cheap 8-ohm speaker. Nowadays, it is more popular to use a ceramic-based piezoelectric sounder like a Sonalert which makes a high-pitched tone. Usually these were hooked up to "driver" circuits which varied the pitch of the sound or pulsed the sound on and off. In game shows it is also known as a "lockout system," because when one person signals ("buzzes in"), all others are locked out from signalling. Several game shows have large buzzer buttons which are identified as "plungers". The word "buzzer" comes from the rasping noise that buzzers made when they were electromechanical devices, operated from stepped-down AC line voltage at 50 or 60 cycles. Other sounds commonly used to indicate that a button has been pressed are a ring or a beep.

CONCLUSION

According to the achieved results, the proposed solution is usable for its purpose under certain performance limitations (such as number of processed frames or measurements per second). Moreover, it relies on both open hardware and free software, being definite and desirable advantage for such systems. In future, it is planned to experiment with various deep learning and computer vision frameworks for object detection on Arduino in order to achieve higher frame rate. Moreover, we would like to extend this solution with environment sensing mechanisms for adaptive building air conditioning and ventilation airborne protection in order to reduce the spread of corona virus indoors especially during summer. Finally, the ultimate goal is to integrate the system presented in this paper with our framework for efficient resource planning during pandemic



crisis in order to enable efficient security personnel scheduling, together with risk assessment based on statistics about respecting the safety guidelines and air quality.

REFERENCES

- [1] Corona virus Update (Live) [online]. 2020.
- [2] P. Zhai et al., “The epidemiology, diagnosis and treatment of COVID-19”, 2020.
- [3] P. Dawson et al., “Loss of Taste and Smell as Distinguishing Symptoms of COVID-19”, 2020.
- [4] L. Morawska, “How can airborne transmission of COVID-19 indoors be minimised?”, 2020.
- [5] T. Galbadage, B. Peterson, R. Gunasekera, “Does COVID-19 Spread Through Droplets Alone?”2020
- [6] Coronavirus Incubation Period [online]. 2020.
- [7] D. Oran, E. Topol, Prevalence of Asymptomatic SARS-CoV-2 Infection: 2020.
- [8] T. Dbouk, D. Drikakis, “On respiratory droplets and face masks”, 2020.
- [9] Y. Song et al., “COVID-19 Treatment: Close to a Cure? – A Rapid Review of Pharmacotherapies for the Novel Coronavirus” 2020.
- [10] V. Balachandar et al., “COVID-19: emerging protective measures”, 2020.