Brilliance
Research of Artificial Intelligence

E-ISSN : 2807-9035
Volume 3, **Number** 2, November 2023
https://doi.org/10.47709/brilliance.v3i2.3057

Design of a Microcontroller-Based Corn Dryer

Wiwik Wiharti*[1],Zulka Hendri[2],Yona Mayura[3],Ihsan Lumasa Rimra[4], Ikhwan Razikin[5]

[1,2,3,5]Department of Electrical Engineering, Electrical Engineering Study Program, Politeknik Negeri Padang

[4]Department of Electrical Engineering, Telecommunication Engineering Study Program, Politeknik Negeri Padang

[1]wiwik.rimra@gmail.com

open access

***Corresponding Author**

Article History:
Submitted: 28-10-2023
Accepted: 31-10-2023
Published: 06-11-2023

Keywords:
Drying; Arduino; DHT22 Sensor; Heat Element.

ABSTRACT

Maize drying is a crucial procedure in the agricultural and food industries, aimed at decreasing the moisture content of maize kernels to prolong their shelf life, inhibit microbial growth and maintain their quality. Two common methods for drying corn include natural techniques, which involve drying the maize in the sun, and artificial techniques, where specialized equipment such as hot air dryers, drum dryers and microwaves are employed. The author constructed a corn dryer utilizing Arduino. This device employs heat elements and fans to expedite the corn drying process. The tool can significantly quicken the corn drying process. The author utilises a DHT22 sensor to gauge temperature and humidity levels of the corn. The device is capable of generating temperatures ranging from 20°C to 90°C, thanks to the assistance of the DHT22 sensor during the drying process.

INTRODUCTION

Indonesia is predominantly an agricultural country, with the majority of its population consisting of corn farmers (Suryana, A., & Agustian, A, 2014). To ensure that post-harvest processing of agricultural products results in materials that can be processed and stored with the same quality as immediately after harvest, proper handling is necessary. One crucial step in the process is maize drying, which involves the removal of moisture to achieve the desired maize quality. The primary aims of maize drying are to preserve the material's quality, inhibit the proliferation of microorganisms, such as fungi and bacteria, which favour damp environments, and uphold the physical and chemical characteristics of the dried material. Various techniques may be used to dry maize, depending on the material type and the desired moisture level.

The quality criterion for maize as a viable feed component is set at a maximum of 14% as per (Sunarti, D & Arnol Turang, 2017). Corn kernels intended for storage should maintain a moisture content of 13% to prevent mould growth and minimize corn respiration. The quality of the corn improves with lower moisture content. Traditionally, farmers dry corn using manual methods which require significant time and energy. To address this issue, a tool that enables the drying of corn efficiently and reliably is necessary. Consequently, my research involves developing a Microcontroller-Based Corn Drying Tool Design for corn drying.

LITERATURE REVIEW

Corn

Maize (Parajuli, Rahim, 2016) is a seasonal crop that falls under the graminae class, and usually has a single stem, though in some genotypes and certain environments, tiller branches may develop. Maize stems are composed of nodes and internodes, with leaves exhibiting an alternate phyllotaxy. The male flowers appear on distinct parts of the plant, and cross-pollination is typical. Maize is a plant that requires short-day conditions. The number of leaves is determined at the beginning of male flower initiation and is influenced by genotype, length of irradiation, and temperature.

Maize is part of the food crop sub-sector, which facilitates growth in both upstream and downstream industrial sectors, contributing to overall national economic development. The advantages of maize extend beyond its role as a food source, as it is also utilised for feed and other industrial materials. According (Kasryno et al, 2007) estimation, over 55% of domestic maize demand is utilised for animal feed, while 30% is used for human consumption and the remaining percentage is allocated for various other needs and seeds. As a result, the demand for maize continues to rise (Mulyani, A., & Suwanda, M. H, 2019).

Brilliance
Research of Artificial Intelligence

E-ISSN : 2807-9035
Volume 3, **Number** 2, November 2023
https://doi.org/10.47709/brilliance.v3i2.3057

Drying

Post-harvest treatment of maize can significantly enhance its usability, thereby providing greater benefits to human welfare (Lisman, F et al ,2019). This can be achieved through ensuring the freshness of maize by preserving it in its original or processed form to maintain availability. One effective method to prolong the shelf life of agricultural produce is through drying.

Drying aims to reduce the water content to a level where biological reactions cease, preventing the growth of microorganisms and insects (Taufiq, M, 2004). The process involves extracting water from the material until it reaches a safe moisture content for processing and storage. The SNI 01-4483-1998 standard stipulates that maize used as feed raw materials must have a moisture content of no more than 14%.

The drying process is the only way to reduce the moisture content of maize to the standard, therefore drying is the most important part in the production of quality maize. According to (Parajuli, Rahim, 2016), the average moisture content of maize kernels circulating in the community still has a high moisture content, so it cannot be exported outside the country and cannot be stored for a long time, especially for industrial needs, the maximum moisture content of maize kernels is 14% (Resmisari, A. ,2006).

Understanding Arduino Software IDE.

The term IDE stands for Integrated Development Environment, which refers to software used for development purposes. It is named as such because through this software, Arduino programming is executed to carry out functions embedded through programming syntax. Arduino employs its own programming language similar to C language. The Arduino programming language (Sketch) has been modified to enhance its accessibility for novice programmers, diverging from its original format. Prior to its release, the Arduino microcontroller IC has been fitted with a programme called Bootloader, which acts as an interface between the Arduino compiler and the microcontroller.

The Arduino IDE is built using the Java programming language and comes with a C/C++ library known as Wiring. The Arduino IDE is built using the Java programming language and comes with a C/C++ library known as Wiring. This library simplifies input and output operations. Originally, the Arduino IDE was adapted from the Processing software for the purpose of programming with Arduino.

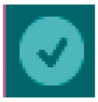
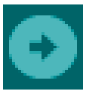
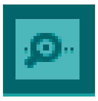
Sketch Writing

Programmes created on the Arduino Software (IDE) are known as sketches. These sketches are composed in a text editor and stored in a document with the .ino extension. The text editor provided in the Arduino Software includes helpful features such as cut, paste, cache, and replace functions that simplify the programming process. Additionally, the Arduino IDE Software displays critical messages in a black message box, including errors, compilation details, and programme uploading status. At the bottom right-hand side of the Arduino IDE software, the configured board and the used COM Ports are displayed.

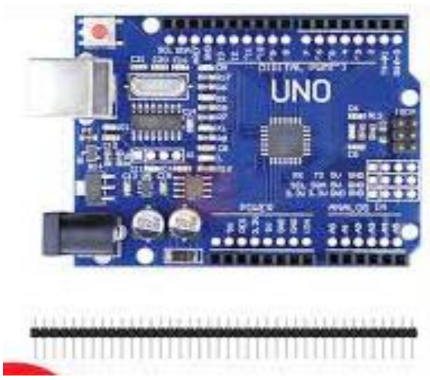
Figure 1. Arduino Integrated Development Environment Display

Brilliance
Research of Artificial Intelligence

E-ISSN : 2807-9035
Volume 3, **Number** 2, November 2023
https://doi.org/10.47709/brilliance.v3i2.3057

	**Verify** Serves to verify whether your code conforms to existing programming regulations.
	**Upload** Compiling code creates a language that can be understood by the machine, namely the Arduino.
	**New** Function to Create a New Sketch
	**Open** Serves to open the sketch that you have created and reopen for editing or just re-uploading to Arduino.
	**Save** Function to save the created sketch.
	**Serial Monitor** This section facilitates the opening of the serial monitor. The serial monitor is a window that presents data transmissions between Arduino and the sketch served through the serial port. The Serial Monitor proves beneficial for programming and debugging, bypassing the need for the LCD on Arduino. This window can display process values, readings, and error messages.

Arduino Uno



Sumber : electronicshub.org
Figure 2. Arduino Uno

The Arduino Uno microcontroller board is founded on the ATmega328 (datasheet) and features 14 digital input pins, of which 6 can be employed as PWM output, and 6 analogue input pins (Louis, L. ,2016). The Arduino Uno microcontroller board is founded on the ATmega328 (datasheet) and features 14 digital input pins, of which 6 can be employed as PWM output, and 6 analogue input pins. The Arduino Uno microcontroller board is founded on the ATmega328 (datasheet) and features 14 digital input pins, of which 6 can be employed as PWM output, and 6 analogue input pins. It also includes a 16 MHz crystal oscillator, USB connectivity, a power jack, an ICSP header, and a reset button. To operate, easily link the Arduino Uno Board to a computer via a USB cable or furnish electricity with an AC-to-DC adapter or battery.

All 14 digital pins on the Arduino Uno have dual functionality as inputs and outputs, enabled through the pinMode(), digitalWrite(), and digitalRead() functions. These functions operate at a voltage of 5 volts. Each pin can provide or receive a maximum current of 40 mA while featuring a default disconnected pull-up resistor of 20-50 kOhm (David, N. et al, 2015).

Brilliance
Research of Artificial Intelligence

E-ISSN : 2807-9035
Volume 3, **Number** 2, November 2023
https://doi.org/10.47709/brilliance.v3i2.3057

Sensor DHT 22

Figure 3. DHT22 sensor

The DHT22 sensor is capable of measuring temperature and humidity and has a digital output signal. It possesses a high level of accuracy since it takes into account the room temperature that is set using the stored value in its integrated OTP memory. The DHT22 sensor has a broad temperature and humidity range. Additionally, it can distribute output signals via cables up to 20 metres long, making it a suitable choice, even for distant placements. Common applications for this sensor include temperature and humidity monitoring in various settings, including, but not limited to, animal cages, residential and commercial buildings, and warehouses. In addition to reading the temperature and humidity of the room, the DHT22 sensor featured in this Garbage Dryer can also measure these factors outside the room. In addition to reading the temperature and humidity of the room, the DHT22 sensor featured in this Garbage Dryer can also measure these factors outside the room. Providing accurate readings, it serves as a highly effective temperature and humidity detector.

Step Down Regulator

The Step-Down Voltage Regulator, also known as a DC Buck Converter, is an efficient electronic module that converts high voltage input from a power source to a lower output voltage. Technical term abbreviations will be stated upon initial usage. The logical flow of information will be ensured with a clear and objective language, correct grammatical structure, and consistent technical terms. The use of colloquial words, figurative expressions, and emotional language will be avoided. Citation and footnote styles will follow convention, maintaining regular format and structure used by the author and institution. This module is powered by the LM2596 integrated circuit, which is capable of regulating a 3A Step-Down.

Source : techdelivers.com
Figure 3. Step Down Regulator DC

The LM2596 chip operates at a switching frequency of 150 kHz, enabling smaller filter components compared to those typically required by low-frequency switching regulators. The IC manufacturer guarantees an output voltage difference tolerance of only ±4% at specified input voltage and output load conditions, and ±15% tolerance at oscillator frequency. The IC can enter sleep mode externally, consuming only 80µA power in standby. Protection features consist of a two-stage current limiting mechanism that reduces frequency for the switch output, and an automatic chip shutdown function in the event of over-temperature conditions.

Heater Element

Electric heating elements are extensively used in everyday life, in both domestic and industrial settings. The shape and type of these elements vary depending on their function, installation location, and the medium to be heated. Technical abbreviations will be explained upon first use. Objective evaluations will be clearly marked as such, with clear language avoiding biased or emotional expressions. The text will be free of grammatical, spelling and punctuation errors. The language will be formal, avoiding contractions, colloquialisms, and jargon while being comprehensible and logically structured. Sections will be well-defined and will follow commonly established conventions. The language will be clear, value-neutral, and consistent with technical terminologies. Furthermore, quotes will be set apart, and filler words will be avoided. Finally, the text will employ a logical flow of information with causal connections between statements.

E-ISSN : 2807-9035
Volume 3, **Number** 2, November 2023
https://doi.org/10.47709/brilliance.v3i2.3057

Figure 4. Heater Element

Buzzer

The buzzer is an electronic device that converts electrical vibrations into sound vibrations. It comprises an electromagnetic coil linked to a diaphragm. The coil produces a magnetic field when energised, causing the coil to move in or out, depending on the current direction and magnet polarity. As the coil is connected to the diaphragm, any motion of the coil will cause the diaphragm to oscillate, resulting in sound production through the movement of air. Buzzers are a widely used means of indicating the completion of a process or alerting of an error in a device such as an alarm. Buzzers are classed as either active or passive. Active buzzers can produce sound immediately when given voltage, whereas passive buzzers can only sound when there is a change in voltage frequency. The buzzer can emit a tone-like sound by altering the frequency of the voltage it receives, similar to how sounds with varying frequencies operate. The LED buzzer may be illuminated through either an on-off (high-low) signal or via a PWM signal.

METHOD

Working Principle of Dryer

The microcontroller-based corn dryer operates by receiving a voltage supply from the PLN, which is first converted from 220VAC to 12VDC using a Power Supply. Technical abbreviations will be explained upon their first use. The 12VDC is further reduced to 5VDC with the use of a Stepdown device before it is inputted into the Arduino Uno. Avoid using ornate language or ambiguous terms. The DHT22 sensor in this device plays a crucial role in detecting humidity and temperature levels. Additionally, adhere to conventional formatting and structure guidelines, eliminate biased language, and correct any grammatical and punctuation errors. Additionally, the LCD screen will display the obtained temperature and humidity values, and the relay will activate if the temperature shown on the screen falls below 60^{o}C, turning off when it is above 60^{o}C. Technical abbreviations will be explained when first introduced. The structure of the system follows a logical progression with causal connections between statements, and all sections adhere to conventional academic layout and formatting requirements. Citations and footnotes are provided in a consistent style, avoiding filler words, grammatical errors, and punctuation mistakes. The buzzer will sound only if the humidity is less than 14%, remaining silent if the humidity is above 14%. Careful attention is given to precision in word choice and avoiding subjective evaluations. The language used is formal, employing a high-level of standard terminology, passive tone, and consistent technical terms.

System Block Diagram

When planning tools to aid and streamline the tool-making process, the author begins by creating a block diagram that elucidates the tool system. Each block is connected to form a comprehensive system of tools. The image below displays the block diagram crafted by the author.

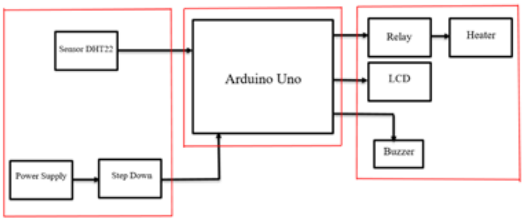


Figure 5. System Design Block Diagram

The diagram depicts the workflow of the system within the final project tool, specifically focusing on the use of Arduino Uno. The system incorporates a logical flow of information utilizing connected lines between the input, process, and output modules. Technical terms are judiciously explained to ensure ease of understanding for readers. The components illustrated include Power Supply, Stepdown, and DHT22 sensors as inputs, Arduino Uno functioning as a processing element, and LCD K, Relay and Buzzer serving as outputs.

E-ISSN : 2807-9035
Volume 3, **Number** 2, November 2023
https://doi.org/10.47709/brilliance.v3i2.3057

The power supply is utilised to convert the voltage from 220V AC to 12V DC before passing through the Stepdown, which further reduces it to 5V DC for use with Arduino Uno. The DHT22 sensor is responsible for providing temperature and humidity readings within the room, which are then relayed to Arduino Uno. After receiving readings from the DHT22 sensor, the Arduino Uno sends a signal to both the Relay and LCD to display the number of readings taken. Once the DHT22 sensor has taken the required readings, the buzzer will activate if the predetermined humidity level has been reached. Technical abbreviations, such as 'DHT22', should always be explained upon first use.

Working Principle of Arduino Uno with DHT22 Sensor on Corn Dryer Tool

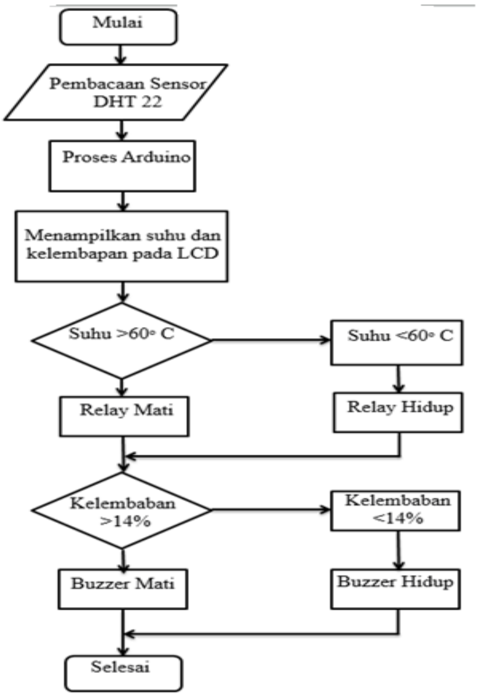


Fig. 6. Arduino Uno with DHT22 Sensor on Corn Dryer Tool Flowchart

Before addressing the problem, the author conducted research on how to dry corn. Before addressing the problem, the author conducted research on how to dry corn. Once a solution was found, several steps were taken to resolve the issue. The problem-solving stages were executed to ensure a proper and structured resolution. The final project report outlines a systematic approach comprising the stages presented in the flowchart.

Hardware Design

Hardware design for mechanical manufacturing of tools comprises stages that will be utilized for the design of the hardware. The following are the stages of the hardware design for the tool.

Table 1. Materials used.

NO	Material Name	Total
1	Arduino Uno	1
2	Solid State Relay (SSR 25A)	1
3	Power Supply 12V	1
4	DHT22 sensor	1
5	LCD Keypad Shield	1
6	Step Down	1
7	Buzzer	1
8	12V Fan	1
9	Heater	2
10	Indicator Light	1
11	Switch	1
12	Light	1

E-ISSN : 2807-9035
Volume 3, Number 2, November 2023
https://doi.org/10.47709/brilliance.v3i2.3057

Assembling Tools and Materials

Once the tool frame has been manufactured, the components are assembled and organized by the author. The microcontroller-based garbage dryer arranges these components in an orderly fashion on the control panel of the tool. During the drying process, the Arduino Uno controls this tool.

The tool's circuit comprises a switch that automatically activates it, and the heater and fan function to dry the trash. Control of this automatic system is performed by an Arduino Uno.

Figure 7. Tool Control Panel

This garbage dryer features indicator lights to signal when the heater is on. Technical abbreviations are explained upon first use. The machine utilises a DHT22 sensor which measures temperature and humidity levels in the rice. An Arduino Uno controls the DHT22 sensor, and the temperature sensor readings are displayed on the LCD Keypad Shield. The structure is clear, logical, and concise, formatted following standard academic guidelines. Objective and formal language is favoured, with grammatical accuracy ensured.

Arduino Uno programme design for DHT22 Sensor

In this tool, the DHT22 sensor is controlled by an Arduino Uno to regulate the temperature and humidity in a corn dryer. Alongside the DHT22 sensor, the author incorporates the LCD Keypad Shield to display readings of temperature and humidity. Further, this LCD Keypad functions to set required temperature and humidity. The subsequent program is designed accordingly.

```
#include <LiquidCrystal.h>
#include <EEPROMex.h>
#include <Arduino.h>
#include <Adafruit_Sensor.h>
#include <DHT.h>
#include <DHT_U.h>
//#include <PID_v1.h>
#define DHTPIN 12
#define DHTTYPE    DHT22
DHT_Unified dht(DHTPIN, DHTTYPE);

LiquidCrystal lcd(8, 0, 9, 4, 5, 6, 7);

long timer[7] = {0, 25200, 46800, 64800, 0, 0, 0};
long timerSt[7] = {0, 0, 0, 0, 0, 0, 0};

byte relayPin = A6;
float temp, rh;
uint32_t delayMS;
byte setSt = 0;
int setSuhu = 60;
int setRh = 50;
float Op;
```

Brilliance
Research of Artificial Intelligence

E-ISSN : 2807-9035
Volume 3, **Number** 2, November 2023
https://doi.org/10.47709/brilliance.v3i2.3057

```
void setup() {
  Serial.begin(9600);
  delay(100);
  //saveData();
  openData();
  while (!Serial) ; // wait for serial
  lcd.begin(16, 2);
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("IKHWAN RAZIQIN");
  lcd.setCursor(0, 1);
  lcd.print("2001032065");
  delay(2000);
  pinMode(11, OUTPUT);
  digitalWrite(11, LOW);
  lcd.clear();
  setupDHT();

  beep();
  beep();
}

void loop() {
  tombol();
  if (setSt == 0) {
    if (millis() - 1000 > oldMillis) {
      oldMillis = millis();
      loopDHT();
      //lcd.clear();
      lcd.setCursor(0, 0);
      lcd.print("T  = "); lcd.print(temp, 1); lcd.print(" C,  "); lcd.print(setSuhu);

      lcd.setCursor(0, 1);
      lcd.print("Rh = "); lcd.print(rh, 1); lcd.print(" %,  "); lcd.print(setRh);
      countt++;
      if (countt >= 8) countt = 0;
      Serial.println(countt);
      pid(setSuhu, temp, 5, 0, 0);
      Serial.print("OP ");
      Serial.println(Op);
      if (Op < 0) Op = 0;
      ope = Op / 8;
      if (ope >= 6) ope = 5;
      if (ope <= 0) ope = 0;
      if (countt < ope) digitalWrite(11, HIGH); else digitalWrite(11, LOW);
      Serial.print("ope ");
      Serial.println(ope);
      Serial.println(countt);
void setupDHT() {
  Serial.begin(9600);
  // Initialize device.
  dht.begin();
  Serial.println(F("DHTxx Unified Sensor Example"));
  // Print temperature sensor details.
  sensor_t sensor;
  dht.temperature().getSensor(&sensor);
  Serial.println(F("------------------------------------"));
  Serial.println(F("Temperature Sensor"));
  Serial.print  (F("Sensor Type: ")); Serial.println(sensor.name);
  Serial.print  (F("Driver Ver:  ")); Serial.println(sensor.version);
  Serial.print  (F("Unique ID:   ")); Serial.println(sensor.sensor_id);
  Serial.print  (F("Max Value:   ")); Serial.print(sensor.max_value); Serial.println(F("°C"));
  Serial.print  (F("Min Value:   ")); Serial.print(sensor.min_value); Serial.println(F("°C"));
  Serial.print  (F("Resolution:  ")); Serial.print(sensor.resolution); Serial.println(F("°C"));
  Serial.println(F("------------------------------------"));
  // Print humidity sensor details.
  dht.humidity().getSensor(&sensor);
  Serial.println(F("Humidity Sensor"));
  Serial.print  (F("Sensor Type: ")); Serial.println(sensor.name);
  Serial.print  (F("Driver Ver:  ")); Serial.println(sensor.version);
  Serial.print  (F("Unique ID:   ")); Serial.println(sensor.sensor_id);
  Serial.print  (F("Max Value:   ")); Serial.print(sensor.max_value); Serial.println(F("%"));
  Serial.print  (F("Min Value:   ")); Serial.print(sensor.min_value); Serial.println(F("%"));
  Serial.print  (F("Resolution:  ")); Serial.print(sensor.resolution); Serial.println(F("%"));
```

Brilliance
Research of Artificial Intelligence

E-ISSN : 2807-9035
Volume 3, **Number** 2, November 2023
https://doi.org/10.47709/brilliance.v3i2.3057

The presented software is developed for monitoring corn dryers. Reading from the DHT22 sensors is facilitated by the LCD Keypad Shield, while relays serve as switches to enable disconnection and connection of various components. The program utilizes a number of libraries, including Liquid Crsytal.h, DHT.h, EEPROMex.h, and Adafruit_Sensor.h. The control mechanism is operated by Arduino Uno, while the system employs DHT22 sensors to detect temperature and humidity levels. Please see the wiring circuit for the Arduino Uno below.

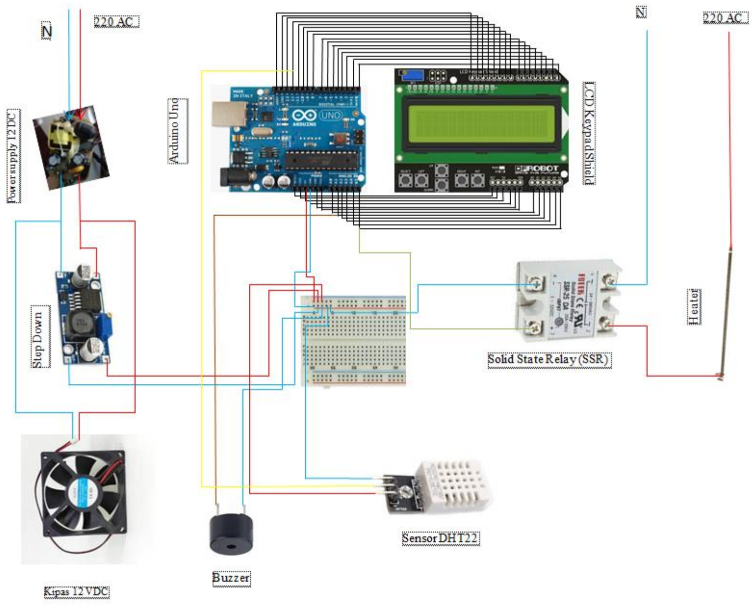


Figure 8. Arduino Wiring Diagram

Testing Methodology

Testing methodology is a procedural process used to retrieve data for analysis. The tool implemented by the author employs qualitative methods, a research strategy that involves observation and analysis of data. During the development of the final project tool, the author encountered obstacles which prevented timely completion. Objective analysis indicates that the author's ability to complete the project was impeded by unforeseen technical challenges. It is important to consider these issues when assessing the success of the project.

1. When testing without a switched on circuit source.
2. Initially, the relay functions properly. However, after some time has elapsed, the relay ceases to function, therefore, the heater remains on, leading to an increase in temperature, thereby posing a potential risk of damaging the DHT22 sensor.

Problem Solving

From the issues encountered, the author has devised solutions for each problem in their final project tool. The following presents an explanation of how to address the problems.

1. When testing the circuit source, it failed to activate. The authors subsequently reconnected the cables, resulting in the circuit being restored, and the device operating correctly. Each circuit on the control panel was inspected, and it was discovered that a pair of wires had become loose due to poor coupling. The authors subsequently reconnected the cables, resulting in the circuit being restored, and the device operating correctly. The authors subsequently reconnected the cables, resulting in the circuit being restored, and the device operating correctly.
2. Initially, the relay functions correctly, but eventually it ceases to function, resulting in the continuous heating of the heater. This overheating nearly causes damage to the DHT22 sensor. Upon investigating the issue, it was discovered that the relay was unable to endure the high levels of power being supplied to the heater. To address the problem, the author has replaced the old relay with a 25A Solid State Relay. After the circuit was replaced, the tool resumed working without any further issues.

E-ISSN : 2807-9035
Volume 3, **Number** 2, November 2023
https://doi.org/10.47709/brilliance.v3i2.3057

```
#include <LiquidCrystal.h>
#include <EEPROMex.h>
#include <Arduino.h>
#include <Adafruit_Sensor.h>
#include <DHT.h>
#include <DHT_U.h>
//#include <PID_v1.h>
#define DHTPIN 12
#define DHTTYPE    DHT22
DHT_Unified dht(DHTPIN, DHTTYPE);

LiquidCrystal lcd(8, 0, 9, 4, 5, 6, 7);

long timer[7] = {0, 25200, 46800, 64800, 0, 0, 0};
long timerSt[7] = {0, 0, 0, 0, 0, 0, 0};

byte relayPin = A6;
float temp, rh;
uint32_t delayMS;
byte setSt = 0;
int setSuhu = 60;
int setRh = 50;
float Op;
```

RESULT AND DISCUSSION

Monitoring Programme Using Arduino Uno with DHT22 Sensor as Temperature and Humidity Detector

The provided program for operating the Corn dryer integrates several libraries, such as (Liquid Crystal.h), (DHT.h), (EEPROMex.h), and (Adafruit_Sensor.h). In this program, the author utilises pin A5 on the Arduino board for Relay, as well as the LCD Keypad Shield. The author employs a DHT22 sensor to detect temperature and humidity levels within the garbage dryer.

```
void setup() {
  Serial.begin(9600);
  delay(100);
  //saveData();
  openData();
  while (!Serial) ; // wait for serial
  lcd.begin(16, 2);
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("IKHWAN RAZIQIN");
  lcd.setCursor(0, 1);
  lcd.print("2001032065");
  delay(2000);
  pinMode(11, OUTPUT);
  digitalWrite(11, LOW);
  lcd.clear();
  setupDHT();

  beep();
  beep();
}

void loop() {
  tombol();
  if (setSt == 0) {
    if (millis() - 1000 > oldMillis) {
      oldMillis = millis();
      loopDHT();
      //lcd.clear();
      lcd.setCursor(0, 0);
      lcd.print("T  = "); lcd.print(temp, 1); lcd.print(" C,  "); lcd.print(setSuhu);

      lcd.setCursor(0, 1);
      lcd.print("Rh = "); lcd.print(rh, 1); lcd.print(" %,  "); lcd.print(setRh);
      countt++;
      if (countt >= 8) countt = 0;
      Serial.println(countt);
      pid(setSuhu, temp, 5, 0, 0);
      Serial.print("OP ");
      Serial.println(Op);
      if (Op < 0) Op = 0;
      ope = Op / 8;
      if (ope >= 6) ope = 5;
      if (ope <= 0) ope = 0;
      if (countt < ope) digitalWrite(11, HIGH); else digitalWrite(11, LOW);
      Serial.print("ope ");
      Serial.println(ope);
      Serial.println(countt);
```

Brilliance
Research of Artificial Intelligence

E-ISSN : 2807-9035
Volume 3, **Number** 2, November 2023
https://doi.org/10.47709/brilliance.v3i2.3057

```
void setupDHT() {
  Serial.begin(9600);
  // Initialize device.
  dht.begin();
  Serial.println(F("DHTxx Unified Sensor Example"));
  // Print temperature sensor details.
  sensor_t sensor;
  dht.temperature().getSensor(&sensor);
  Serial.println(F("------------------------------------"));
  Serial.println(F("Temperature Sensor"));
  Serial.print  (F("Sensor Type: ")); Serial.println(sensor.name);
  Serial.print  (F("Driver Ver:  ")); Serial.println(sensor.version);
  Serial.print  (F("Unique ID:   ")); Serial.println(sensor.sensor_id);
  Serial.print  (F("Max Value:   ")); Serial.print(sensor.max_value); Serial.println(F("°C"));
  Serial.print  (F("Min Value:   ")); Serial.print(sensor.min_value); Serial.println(F("°C"));
  Serial.print  (F("Resolution:  ")); Serial.print(sensor.resolution); Serial.println(F("°C"));
  Serial.println(F("------------------------------------"));
  // Print humidity sensor details.
  dht.humidity().getSensor(&sensor);
  Serial.println(F("Humidity Sensor"));
  Serial.print  (F("Sensor Type: ")); Serial.println(sensor.name);
  Serial.print  (F("Driver Ver:  ")); Serial.println(sensor.version);
  Serial.print  (F("Unique ID:   ")); Serial.println(sensor.sensor_id);
  Serial.print  (F("Max Value:   ")); Serial.print(sensor.max_value); Serial.println(F("%"));
  Serial.print  (F("Min Value:   ")); Serial.print(sensor.min_value); Serial.println(F("%"));
  Serial.print  (F("Resolution:  ")); Serial.print(sensor.resolution); Serial.println(F("%"));
```

The Arduino program is intended to operate the corn dryer. The author employs various libraries, such as Liquid Crystal.h, DHT.h, EEPROMex.h, and Adafruit_Sensor.h in the above-mentioned program. For the Relay in the Arduino, the author employs pin A5. The LCD Keypad Shield is utilized by the author. For temperature and humidity monitoring, the DHT22 sensor is used in this corn dryer.

Analysis

1. Using 250 grams of corn at 60°C, the humidity of the corn decreased by 34% in 22 minutes and the weight of the corn decreased from 250 grams to 231 grams. Subsequently, the author conducted a follow-up experiment with the same temperature and weight as the previous experiment, resulting in a decrease in corn humidity by 4% from the previous experiment, namely a total decrease of 30% within 60 minutes (1 hour).
2. Using 400 grams of corn at a temperature of 60°C, the humidity of the corn decreased by 34% within 48 minutes, and its weight reduced from 400 grams to 373 grams. Subsequently, the author repeated the experiment with the same quantity and temperature of corn, achieving a 4% reduction in humidity compared to the earlier results, ultimately reaching 30% humidity reduction within 80 minutes (1 hour 20 minutes).
3. Using 500 grams of corn at 60°C, the humidity of the corn decreased by 34% within 67 minutes and the weight of the corn decreased from 500 grams to 467 grams. A subsequent re-experiment was conducted with the same temperature and weight, resulting in a 4% decrease in humidity from the previous experiment, reaching a total reduction of 30% within 105 minutes (1 hour 45 minutes).
4. Humidity refers to the amount of water vapour in the air, which can lead to a moist environment. The level of saturation is mainly determined by temperature. When the partial vapour pressure reaches the saturated vapour pressure of water, the air becomes compressed. Relative humidity (RH) is the proportion of partial water vapour pressure to saturated water vapour pressure, expressed as a percentage [5].

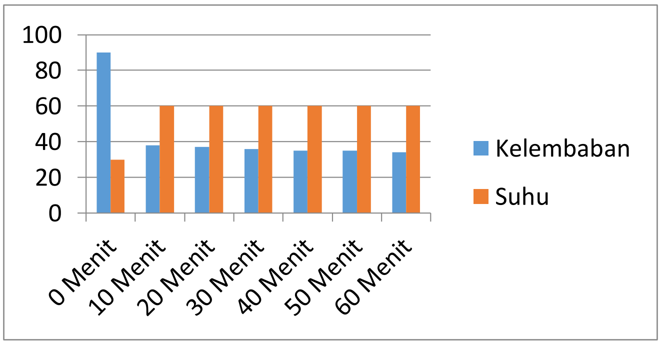


Fig. 8. Temperature and humidity graph of Maize drying process

E-ISSN : 2807-9035
Volume 3, **Number** 2, November 2023
https://doi.org/10.47709/brilliance.v3i2.3057

CONCLUSION

1. The programming utilises various libraries, specifically (Liquid Crystal.h), (DHT.h), (EEPROMex.h), and (Adafruit_Sensor.h). The program comprises multiple operational commands, including the turning off of the relay when the temperature surpasses 60° C and turning it on when the temperature falls below 60° C. Additionally, the buzzer switches off when the humidity level is above 14% and turns on when it is below 14%, signalling its target has been attained.
2. During the installation of an Arduino Uno with a DHT22 sensor on the Corn Dryer Tool, various components were implemented, including the Arduino Uno R3, Solid State Relay (SSR), LCD Keypad Shield, and DHT22 Sensor.
3. The outcomes of the drying experiments conducted with this tool manifest superior drying speed compared to manual methods. Moreover, the usage of this tool in the rainy season eliminates concerns over corn spoilage due to mould growth.

REFERENCES

David, N., Chima, A., Ugochukwu, A., & Obinna, E. (2015). Design of a home automation system using arduino. *International Journal of Scientific & Engineering Research*, *6*(6), 795-801.

Kasryno, F., Effendi P., Suyamto., Adnyana, MO., 2007. *Gambaran Umum Jagung Indonesia*. Pusat Penelitian dan Pengembangan Tanaman Pangan: Bogor.

Lagiyono. (2012). *Pengaruh udara masuk terhadap suhu air conditioner (ac) kapasitas 1 pk pada ruang instalasi uji.* Jurnal Fakultas Teknik Universitas Negeri Semarang.

Louis, L. (2016). working principle of Arduino and u sing it. *International Journal of Control, Automation, Communication and Systems (IJCACS)*, *1*(2), 21-29.

Lisman, F., Syahriza, R., Nurtam, M. R., & Djinis, M. E. (2019). Rancang bangun alat pemipil jagung sederhana. *Agroteknika*, *2*(1), 11-19.

Mulyani, A., & Suwanda, M. H. (2019). Pengelolaan lahan kering beriklim kering untuk pengembangan jagung di Nusa Tenggara. *Jurnal sumberdaya lahan*, *13*(1), 41-52.

Parajuli, Rahim. (2016)**.** *Pengaruh waktu Pengeringan terhadap Laju Penurunan Kadar Air dan Berat Jagung Hibrida (Zea mays l.)*. Skripsi Universtas Ichsan Gorontalo. Gorontalo.

Resmisari, A. (2006). Tepung jagung komposit, pembuatan dan pengolahannya. In *Prosiding Seminar Nasional Teknologi Inovatif Pascapanen Pengembangan Pertanian* (Vol. 28, No. 2, pp. 63-71).

Sunarti, D & Arnol Turang.(2017). *Penanganan Panen dan Pasca Panen Jagung untuk Tingkat Mutu Jagung, Balai Pengkajian Teknologi Pertanian Sulawesi Utara*.

Taufiq, M. 2004. *Pengaruh Temperatur Terhadap Pengeringan Jagung Pada Pengeringan Konvensional* Fakultas Teknik. Universitas Sebelas Maret.

Suryana, A., & Agustian, A. (2014). Analisis daya saing usaha tani jagung di Indonesia. Analisis Kebijakan Pertanian, 12(2), 143-156.