

Microcontroller-Based Automatic Treatment Skin Treater Design

Rama Okta Wiyagi^{*1}, Anna Nur Nazilah Chamim¹, Muhamad Yusvin Mustar¹, Adi Nugroho¹, Yessi Jusman¹

¹Department of Electrical Engineering, Faculty of Engineering, Universitas Muhammadiyah Yogyakarta

Bantul 55183 Daerah Istimewa Yogyakarta, Indonesia

^{*}Corresponding author, e-mail: ramaoktawiyagi@umy.ac.id

Abstract –Rubber or rubber trees (*Hevea brasiliensis*) is one of the export commodities that become the source of income for the state income. From year to year, the increasing demand for rubber in various areas is very significant. Dusun IV Bandar Sidomulyo is a hamlet located in Lempuyang Bandar Village, Way Pengubuan Subdistrict, Central Lampung Regency with ± 200 houses where source of livelihood is still depend on rubber plantation. So the residents must be willing to peel the skin of rubber plants at early days to get maximum rubber yield. The prototype of Automatic Rubber Tree Tool based on Microcontroller that has been designed works peeling the bark of a tree with depth 1,5 mm with Duwet or Jamblang (*Syzygium cumini*) trees as an object, so that the tool can be used to peel rubber automatically on rubber trees generally.

Keywords: Arduino, Control, Rubber

I. Introduction

Rubber tree or rubber plant (*Hevea brasiliensis*) is one of the export crops which is a source of state income and world rubber demand is increasingly increasing. The Indonesian state has many opportunities to exploit the potential of the market [1].

Most of the regions in Indonesia depend on rubber plants for their income. But to produce good rubber must peel the rubber at dawn. Hamlet IV Bandar Sidomulyo is a hamlet located in Lempuyang Bandar Village, Way Pengubuan District, Central Lampung Regency with ± 200 heads of households where the source of livelihood is still dependent on rubber plants. So residents must be willing to peel the skin of rubber plants in the early hours of the morning to get maximum rubber.

Advances in science and technology encourage people to try to overcome all problems that arise around them and optimize existing work. With an automatic Arduino-based rubber tree bark peeler control, it is easy to peel the rubber tree bark without having someone come to peel in the early

hours of the morning. The way the tool works is to use Arduino which will be used as an output data processor. The data will be processed by Arduino in accordance with the program to be made. By using RTC (Real Time Clock), it will determine the tool is working [2].

In the Final Project by Agung Setiawan, Faculty of Electronics and Instrumentation Students, Gajah Mada University (2014), with the title Automatic Watering and Fertilizing System Using RTC (Real Time Clock) and Soil Moisture Sensor Based on Arduino Uno R3. In its working system, this automatic watering and fertilizing system uses Arduino UNO, a soil moisture sensor that is used to measure soil moisture levels in plant pots, RTC (Real Time Clock) to set the time for watering liquid fertilizer periodically in brackets per week, a series relay to turn on and turn off the pump, and a 16x2 LCD that functions as a display of the value of the soil moisture sensor and RTC (Real Time Clock). After testing as a whole the system can water the soil in plants in pots. At pump 1 it is regulated using a soil moisture sensor, and pump 2 is regulated using RTC (Real Time Clock). When the ADC value is above 420, pump 1 will light up watering plants flowing through the pipe to the pot,

and when the ADC value is below 420, pump 1 will stop watering the plants. Furthermore, pump 2 is set to spray liquid fertilizer using RTC once a week on plants [3].

The purpose of this journal is to design a rubber tree peeler that will be controlled using Arduino Atmega328 to give commands to the DC motor so that the DC motor will pull the blade that is connected to the DC motor so that the blade will peel the rubber tree bark automatically in Hamlet IV Bandar Sidomulyo, Way Pengubuan Subdistrict, Central Lampung Regency so that residents do not need to wake up in the early hours of the morning to get maximum rubber latex.

II. Methods

Flowchart for the research is presented in Fig. 1.

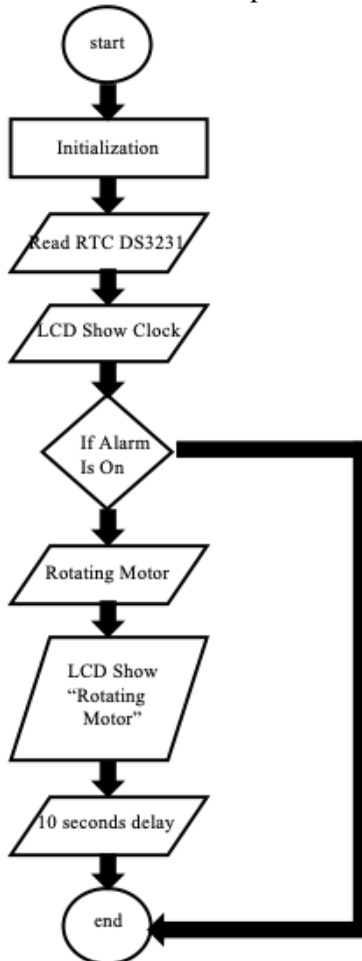


Fig. 1. Overall system flowchart

The design of the tool is done with the aim to get the tool with the minimum percentage of errors. In this case the authors use the Fritzing Software and Proteus 7.5 Professional to design and simulate the tool. The design of the tool includes the selection of

components to be used, namely by considering whether the components are easily available in the market, the amount of voltage supply, and the script program that will be included in the Arduino microcontroller. The tool will be simulated first, if the simulation gets results that are appropriate or close, then the design of the tool has been successful and complete.

III. Results

III.1. Power Supply Circuit Testing

The power supply is used to convert the 12V voltage into the 5V voltage required by the microcontroller circuit. IC 7805 is an IC specifically designed as a voltage regulator. Input a varying DC voltage you will get a stable 5V voltage. Figure 2 is the power supply circuit used for the Arduino nano microcontroller voltage.

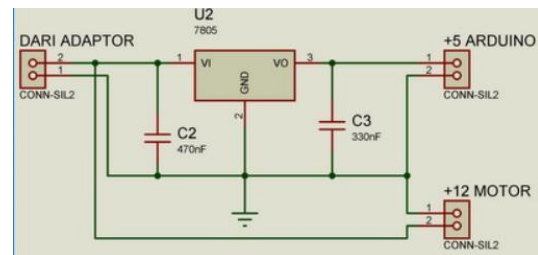


Fig. 2. Power supply circuit testing

According to Figure 2, the test results are obtained as presented in Table I.

TABLE I
POWER SUPPLY CIRCUIT TESTING

Testing	Voltage
1	4.89 V
2	4.90 V
3	4.86 V

The number shown on the moving multimeter shows the voltage that is used is close to 5V, then the power supply circuit is ready to use.

III.2. DC Motor Testing

The motor driver circuit has an IC VN2SP30 that functions as a DC motor driver and a DC motor that functions to move the pulling knife automatically.

By giving a HIGH or LOW input to the VN2SP30 IC the results are as shown in Table II.

TABLE II
DC MOTOR TESTING

Port 11	Port 12	DC Motor
HIGH	HIGH	STOP
LOW	HIGH	CW
HIGH	LOW	CCW
LOW	LOW	STOP

The number shown on the moving multimeter shows the voltage that is used is close to 5V, then the power supply circuit is ready to use.

Looking at the results from Table II shows that the DC motor circuit can function properly and is ready to use.

III.3. Programming Tools

The programming process is done after this tool is finished. The whole series is tested whether it is suitable and there are no errors in the circuit. Then the program is inserted into the Arduino Nano Atmega328 microcontroller and the tool can display the results, the tool is in good condition.

To download the program to the Arduino Nano Atmega328 microcontroller, arduino software is needed which is installed into a laptop or PC. In this final project uses Arduino IDE 1.81 software, which after the installation is complete, additional software is needed, namely CH341SER so that the USB laptop or PC can detect the USB connection from Arduino.

1. Source Code on Arduino Nano Atmega328

Arduino Nano Atmega328 acts as a receiver (receiver) of data from a laptop or sender (transmitter). To control the overall components of both input and output as required in this study, Arduino Nano ATmega328 in the program using the software Arduino IDE is based on the source code below:

a. Settings or void setup ()

The void setup () function is used to initialize variables, pin mode, start using the library, start serial communication, etc. This setup function will only work once, i.e. every powerup or restart of the Arduino board. Settings or void setup () is presented in Figure 3.

b. Execute the Program command or void loop ()

Program Command Execution or void loop () functions to execute program commands that have been made. This function will actively control the Arduino board to either read the input or change the output. Execute the Program command or void loop () is presented in Figure 4.

```
void setup() {
  Serial.begin(9600);
  lcd.begin(16, 2);
  pinMode(MOTOR_A1_PIN, OUTPUT);
  pinMode(MOTOR_B1_PIN, OUTPUT);
  pinMode(PWM_MOTOR_1, OUTPUT);
  digitalWrite (PWM_MOTOR_1, HIGH);

  // Set an alarm
  RTC.read(tm);
  setTime(tm.Hour, tm.Minute, tm.Second, tm.Day,
tm.Month,
tm.YearToCalendar(tm.Year));
  Alarm.alarmRepeat(02, 00, 00, motorAlarm);
}
```

Fig. 3. Settings or void setup ()

```
void loop () //arduino main program
{
  showHours();
  Alarm.delay(1000);
}
```

Fig. 4. Execute the program command or void loop ()

```
void showHours()
{
  RTC.read(tm);
  lcd.setCursor(0, 0);
  lcd.print("Time");
  lcd.setCursor(5, 0);
  lcd.print('=');
  lcd.setCursor(7, 0);
  print2digits(tm.Hour);
  lcd.setCursor(9, 0);
  lcd.print(':');
  lcd.setCursor(10, 0);
  print2digits(tm.Minute);
  lcd.setCursor(12, 0);
  lcd.print(':');
  lcd.setCursor(13, 0);
  print2digits(tm.Second);
  lcd.setCursor(0, 1);
  lcd.print("Date");
  lcd.setCursor(5, 1);
  lcd.print('=');
  lcd.setCursor(6, 1);
  print2digits(tm.Day);
  lcd.setCursor(8, 1);
  lcd.print('/');
  lcd.setCursor(9, 1);
  print2digits(tm.Month);
  lcd.setCursor(11, 1);
  lcd.print('/');
  lcd.setCursor(12, 1);
  print2digits(tm.YearToCalendar(tm.Year));
}
}

void print2digits(int number) {
  if (number >= 0 && number < 10) {
    lcd.print('0');
  }
  lcd.print(number);
}
```

Fig. 5. Real time program

2. Timer and Alarm Program

a. Real Time Program

The Real Time Program is a program that is structured so that the time traveled is according to the actual time. The arrangement of the Real Time program is presented in Figure 5.

The Real Time shown on the LCD on the microcontroller is presented in Figure 6.



Fig. 6. Real time on the LCD

b. Automatic DC Motor Program

Automatic DC Motor Program is a program that is arranged so that the blade can be pulled automatically based on input from RTC DS3231. Testing of the DC motor is obtained when port 6 logic high with a duration of 10 seconds, and after a logic high for 10 seconds then the port 6 to logic Low or DC motor will stop. The DC motor will rotate clockwise (CW) when port 6 has HIGH logic [4]. The information is produced by the program that has been entered into the board of Arduino Nano which is listed in Figure 7.

```
void motorRotates()
{
  if (condition == true)
  {
    digitalWrite (MOTOR_A1_PIN, HIGH);
    digitalWrite (MOTOR_B1_PIN, LOW);
  }
  condition = false;
  return condition;
}
void motorStop()
{
  if (condition == false)
  {
    digitalWrite (MOTOR_A1_PIN, LOW);
    digitalWrite (MOTOR_B1_PIN, LOW);
  }
  condition = true;
  return condition;
}
```

Fig. 7. Automatic DC motor program

c. Alarm Program

Alarm Program is a program that is arranged so that the desired time can adjust the DC motor or knife to work with the desired time. The program was compiled based on input from RTC

DS3231. RTC DS3231 works to calculate the time where when the time shows the desired time, the DC motor pulls the blade and peels the prototype tree bark automatically. After the desired time runs, the DC motor will automatically stop automatically with the time lag given. The use of this alarm is intended to control the blade or DC motor driven. An alarm program is presented in Figure 8.

```
// Set an alarm
RTC.read(tm);
setTime(tm.Hour, tm.Minute, tm.Second, tm.Day,
tm.Month,
tm.YearToCalendar(tm.Year));
Alarm.alarmRepeat(02, 00, 00, motorAlarm);
void motorAlarm()
{
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Motor bergerak");
  motorBergerak();
  delay(5000);
  motorStop();
  lcd.clear();
}
```

Fig. 8. Alarm program

The alarm that is printed on the LCD on the microcontroller is presented in Figure 9.



Fig. 9. Alarm on the LCD

III.4. Overall Tool Testing

This tool is designed using RTC DS3231 as a timer that aims to allow the knife to peel the prototype tree bark automatically with the desired time. To drive a DC motor used Monster Moto Shield VNH2SP30 as a motor driver. This driver serves to control the rotation of the DC motor so that it can pull the knife and peel the tree bark. Arduino Nano Atmega328 microcontroller is used as the brain of this tool. The circuit is connected to a 5V power supply.

The initial condition is that the RTC DS3231 calculates the time and the blade or DC motor is stationary, where port 6 is Low logic. After Arduino nano detects the chill time program, the DC motor will pull the knife and peel the tree bark. The DC

motor works for 10 seconds, after reaching 10 seconds the DC motor or knife stops working and port 6 will have logic Low. Block diagram is presented in Figure 10 and set of tools is presented in Figure 11.

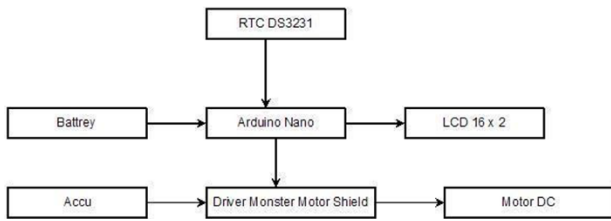


Fig. 10. Block diagram

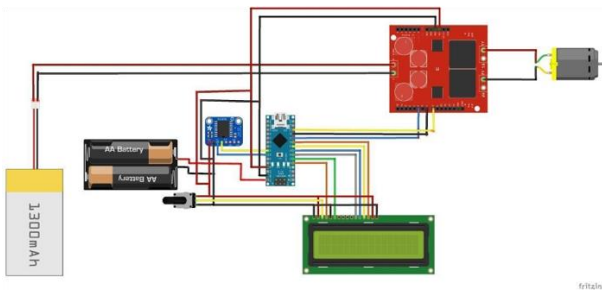


Fig. 11. Set of tools

The overall physical form of the tool is presented in Figures 12, 13, 14, 15 and 16.

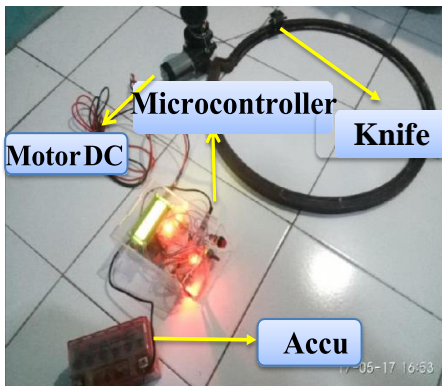


Fig. 12. Overall shape of the device with a microcontroller

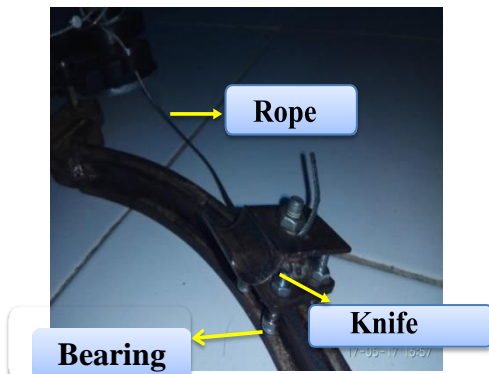


Fig. 13. Physical shape of a paring knife

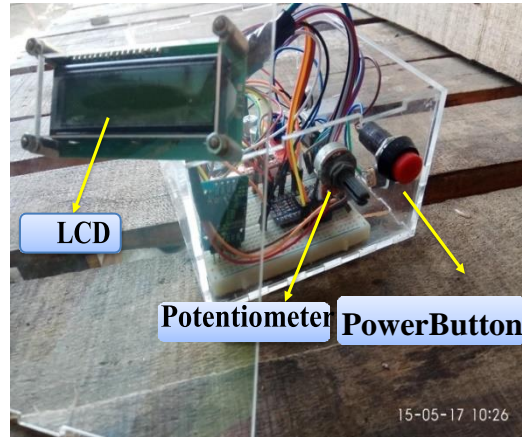


Fig. 14. Physical form of microcontroller looks from the front



Fig. 15. Shape of a knife on a prototype tree



Fig. 16. The results of a knife incision on a tree

After testing the rubber tree peeler, the results of a knife incision on the Duwet (*Syzygium cumini*) tree along 15 cm. The results of the data taken in the testing tool is presented in Table III.

TABLE III
TESTING OVERALL TOOL

Motor DC	No burden	Motor Load	Full Load
Voltage on Battery (V)	13.33 V	13.33 V	13.33 V
Voltage at output (V)	13.27 V	13.27 V	13.27 V
Current (A)	0 A	1.23 A	7.22 A
Power (W)	0 W	16.3221 W	95.8094 W

Based on the test Table III, testing using Motor Load shows a power of 16.3221 W. The equation is as presented in equation 1.

$$\begin{aligned} P &= V \cdot A \\ P &= 13.27 \times 1.23 \\ P &= 16.3221 \text{ W} \end{aligned} \quad (1)$$

IV. Conclusion

Based on research that has been done, conclusions can be drawn namely:

1. A prototype of Arduino Nano Atmega328 microcontroller based automatic rubber tree peeler using RTC DS3231.
2. The prototype of the automatic rubber tree peeler only works at predetermined hours.
3. RTC DS3231 works when the time shows the desired time, the DC motor pulls the blade and peels the prototype tree bark automatically.
4. The DC motor works according to the desired time and peels the Prototype bark according to the standard rubber tree with a depth of 1.5 mm.
5. The DC motor works for 3 seconds on the Prototype tree and produces a knife incision with a 15 cm long incision.

Acknowledgements

This work was supported by Universitas Muhammadiyah Yogyakarta.

References

- [1] S. Damanik, *Budidaya dan Pasca Panen KARET*. Bogor: Pusat Penelitian dan Pengembangan Perkebunan, 2010.
- [2] I. Maulana, "RTC (Real Time Clock) DS1307 (Code Vision AVR) – proyekrumahan.id," 2013. [Online]. Available: <https://proyekrumahan.id/2013/06/rtc-real-time-clock-ds1307-code-vision-avr/>. [Accessed: 02-Apr-2017].
- [3] A. Setiawan, "Sistem Penyiraman dan Pemupukan Otomatis menggunakan RTC (Real Time Clock) dan Sensor Kelembababan Tanah Berbasis Arduino Uno R3," 2014.
- [4] R. Antoni, "Perancangan Sistem Pengaturan Kecepatan Motor DC," *Univ. Marit. Raja Ali Haji*, vol. 1, no. 1, pp. 1–11, 2008.

Authors' information



Chamim Rama Okta Wiyagi obtained his B. Eng in Electrical Engineering from Universitas Muhammadiyah Yogyakarta, Indonesia in 2009. His Master study was done at 2015 at the Electrical Engineering, Universitas Gadjah Mada, Indonesia. He currently is a lecture in department of electrical engineering, Universitas Muhammadiyah Yogyakarta



Anna Nur Nazilah obtained her B. Eng in Electrical Engineering from Universitas Muhammadiyah Yogyakarta, Indonesia. Her Master study was done at 2015 at the Electrical Engineering, Universitas Gadjah Mada, Indonesia. She currently is a lecture in department of electrical engineering, Universitas Muhammadiyah Yogyakarta.



Muhamad Yusvin Mustar obtained his B. Eng in Electrical Engineering from Universitas Muhammadiyah Yogyakarta, Indonesia in 2011. His Master study was done at 2015 at the Electrical Engineering, Universitas Gadjah Mada, Indonesia. He currently is a lecture in department of electrical engineering, Universitas Muhammadiyah Yogyakarta.

Adi Nugroho obtained his B. Eng in Electrical Engineering from Universitas Muhammadiyah Yogyakarta, Indonesia in 2017.



Yessi Jusman obtained her B. Eng in Electrical and Electronic Engineering from Andalas University, Indonesia in 2007. She worked as a Research Assistant started in July 2008 until November 2009 in Universiti Sains Malaysia. Her Master study was done at 2012 at the School of Electrical and Electronic Engineering, USM Engineering Campus in Nibong Tebal, Penang, Malaysia. She was finished her PhD degree at 2016 in University of Malaya with specializes in Image, Signal Processing, and algorithms. She currently is a lecture in department of electrical engineering, Universitas Muhammadiyah Yogyakarta.