

ANALISIS EFISIENSI TRANSFER DAYA NIRKABEL BERBASIS ARDUINO UNO MELALUI ANTARMUKA KOMPUTER

Tujuan

1. Merancang dan membangun sistem transfer daya nirkabel berbasis Arduino Uno sebagai model percobaan untuk menunjukkan potensi pengisian daya *pacemaker* tanpa melakukan pengujian langsung pada perangkat medis tersebut.
2. Mengembangkan program antarmuka komputer menggunakan Arduino IDE dan Python untuk memantau, memplot, dan menyimpan data hasil transfer daya nirkabel secara *real-time*.
3. Menganalisis pengaruh variasi jarak antar kumparan, frekuensi, sudut orientasi, diameter kumparan, dan arus terhadap efisiensi transfer daya, dengan menjaga variabel lain tetap konstan selama setiap pengujian.
4. Mengidentifikasi parameter yang optimal untuk mencapai efisiensi transfer daya yang tinggi pada sistem yang dikembangkan, yang diimplementasikan pada perangkat *pacemaker* tanpa pengaplikasian langsung pada *pacemaker*.

Dasar Teori

2.1 Hukum Faraday

Induksi elektromagnetik adalah peristiwa munculnya gaya gerak listrik (GGL) pada penghantar atau kumparan akibat perubahan fluks magnetik (Firmansyah et al., 2020). Dengan kata lain, perubahan medan magnet (fluks magnet) yang melalui konduktor dapat menghasilkan arus listrik. Berdasarkan percobaan Michael Faraday, medan magnet yang berubah dapat menghasilkan arus listrik, sedangkan medan magnet konstan tidak dapat melakukannya (Firmansyah et al., 2020). Hukum Faraday menyatakan bahwa perubahan fluks

magnetik (Φ_B) pada rangkaian penghantar akan menimbulkan tegangan induksi (ϵ), yang dirumuskan sebagai:

$$\epsilon = -N \frac{d\phi_B}{dt} \quad (2.1)$$

Hukum Faraday menyatakan bahwa jika suatu konduktor dilalui oleh fluks magnet yang berubah-ubah terhadap waktu, maka akan timbul tegangan listrik pada konduktor tersebut. Besarnya tegangan yang dihasilkan sebanding dengan seberapa cepat perubahan fluks terjadi dan seberapa banyak jumlah lilitan kawat pada konduktor (Patil et al., 2017).

Tanda negatif pada persamaan menunjukkan Hukum Lenz yang menyatakan bahwa arah arus induksi selalu berlawanan dengan perubahan fluks magnetik penyebabnya (Andesta, 2018). Fluks magnetik (Φ_B) sendiri merupakan kerapatan garis-garis medan magnet (B) yang melewati luas permukaan tertentu (A), dirumuskan sebagai:

$$\phi_B = B.A.\cos\theta \quad (2.2)$$

Nilai $\cos \theta$ berperan penting karena menunjukkan seberapa besar komponen medan magnet yang benar-benar "memotong" kumparan *receiver*. Jika $\theta = 0^\circ$, medan magnet tegak lurus terhadap permukaan kumparan, sehingga $\cos \theta = 1$, dan fluks magnetik mencapai maksimum. Kondisi ini menyebabkan tegangan induksi di kumparan *receiver* meningkat, sehingga daya yang ditransfer menjadi lebih besar dan efisiensi sistem meningkat.

2.2 Hukum Ampere

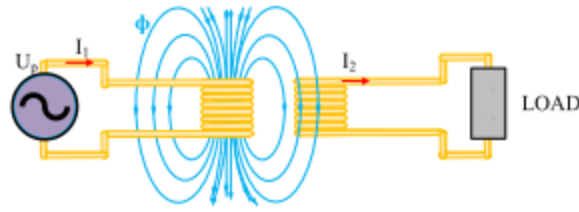
Hukum Ampere menyatakan bahwa sirkulasi medan magnet (B) di sekitar lintasan tertutup berbanding lurus dengan arus listrik total (I_{total}) yang menembus lintasan tersebut. Secara matematis, hukum Ampere dinyatakan sebagai:

$$\oint_{\partial s} \mathbf{B} \cdot d\mathbf{l} = \mu_0 I_{total} \quad (2.3)$$

Hukum Ampere menyatakan bahwa arus listrik yang mengalir melalui suatu konduktor akan menghasilkan medan magnet di sekitarnya. Besarnya medan magnet yang dihasilkan bergantung pada kuat arus listrik dan permeabilitas medium tempat arus mengalir. Dalam medium seperti ruang bebas (udara atau vakum), medan magnet ditentukan oleh permeabilitas ruang bebas (μ_0) (Patil et al., 2017). Semakin tinggi arus dan semakin baik medium dalam menghantarkan medan magnet, maka medan magnet yang terbentuk akan semakin kuat. Dalam konteks transfer daya nirkabel berbasis induksi, Hukum Ampere menjadi dasar penting karena medan magnet yang terbentuk oleh arus di kumparan pengirim akan menentukan seberapa besar energi yang dapat ditransfer ke kumparan penerima. Medan magnet yang kuat dan stabil akan menghasilkan induksi yang lebih besar sesuai Hukum Faraday, sehingga meningkatkan efisiensi transfer energi antar kumparan.

2.3 Induktansi Bersama

Induktansi bersama terjadi ketika perubahan fluks magnetik pada satu kumparan menginduksi tegangan pada kumparan lain. Dalam sistem transfer daya dekat berbasis *Inductive Power Transfer* (IPT), prinsip ini memungkinkan terjadinya transfer energi nirkabel melalui medan magnet. Induktansi bersama inilah yang menyebabkan tegangan muncul di kumparan *receiver* ketika arus pada kumparan *transmitter* berubah. Induktansi bersama dapat diartikan sebagai ukuran seberapa kuat dua kumparan saling berinteraksi melalui kopling magnetik. Induktansi bersama ini sangat dipengaruhi oleh orientasi, jarak, dan posisi antara kedua kumparan. Semakin besar nilai induktansi bersama, maka semakin besar pula fluks magnetik dari *transmitter* yang memotong kumparan *receiver*, sehingga tegangan induksi yang dihasilkan pun semakin tinggi dan efisiensi transfer daya meningkat.



Gambar 2.1: Induktansi Bersama Transfer Daya Nirkabel (Detka & Górecki, 2022)

$$M_{21} = \frac{N_2 \phi_B}{i_1} \quad (2.4)$$

$$M = \frac{\mu_0 \mu_r N_1 N_2 A}{l} \quad (2.5)$$

2.4 Kopling Magnetik

Kopling magnetik adalah interaksi medan magnet antara dua kumparan yang memungkinkan terjadinya induksi tegangan pada kumparan *receiver* akibat perubahan arus di kumparan *transmitter*. Dalam sistem *Inductive Power Transfer* (IPT), kopling magnetik menjadi mekanisme utama yang mendasari proses transfer daya nirkabel, di mana fluks magnetik yang dihasilkan oleh arus bolak-balik di kumparan *transmitter* menembus kumparan *receiver* dan membangkitkan tegangan induksi.

Tingkat kekuatan kopling magnetik ditentukan oleh seberapa besar fluks magnetik dari *transmitter* yang memotong permukaan *receiver*, yang dipengaruhi oleh jarak antar kumparan, orientasi sudutnya (θ), dan keberadaan material penghalang di antara keduanya. Kopling magnetik yang kuat akan menghasilkan nilai induktansi bersama (M) yang tinggi, sehingga efisiensi transfer daya meningkat. Sebaliknya, jika kopling lemah, hanya sedikit fluks yang ditangkap *receiver*, menyebabkan tegangan induksi dan daya yang ditransfer menurun.

Dengan kata lain, kopling magnetik menggambarkan kualitas hubungan magnetik antar kumparan, dan menjadi faktor penentu utama dalam efektivitas

sistem WPT berbasis IPT. Efisiensi transfer daya dalam sistem ini bergantung pada koefisien kopling magnetik (k), yang menunjukkan seberapa baik fluks magnetik dari kumparan pengirim diterima oleh kumparan penerima:

$$k = \frac{M}{\sqrt{L_1 L_2}} \quad (2.6)$$

Nilai k berkisar antara 0 hingga 1, dengan $k = 1$ menunjukkan kondisi ideal di mana seluruh fluks magnetik dari kumparan pengirim diterima oleh kumparan penerima. Faktor seperti jarak antara kumparan dan orientasi kumparan sangat mempengaruhi nilai ini (Diharjo, 2018).

2.5 Resistansi dan Arus Eddy

Resistansi intrinsik kawat adalah hambatan listrik yang secara alami dimiliki oleh suatu bahan penghantar (konduktor), tergantung pada jenis material, panjang kawat, dan luas penampangnya. Hambatan ini terjadi karena elektron dalam kawat bertabrakan dengan atom-atom di dalam bahan saat mengalir, sehingga menimbulkan resistansi terhadap arus listrik (Ariyanto, 2016). Resistansi ini menjadi penting dalam sistem transfer daya karena resistansi yang tinggi menyebabkan kerugian daya (*loss*) dalam bentuk panas, yang akan menurunkan efisiensi sistem secara keseluruhan.

$$R = \rho \frac{l}{A} \quad (2.7)$$

Arus eddy adalah arus listrik yang diinduksi dalam konduktor akibat perubahan medan magnet (Fadhliyansyah, 2017). Arus ini menyebabkan rugi daya dalam bentuk panas, yang dapat mengurangi efisiensi sistem WPT. Dalam konteks sistem WPT, arus eddy menjadi salah satu sumber rugi daya, karena energi magnetik sebagian diubah menjadi panas pada material di sekitar kumparan. Daya hilang akibat arus eddy dapat dihitung dengan:

$$P_e = k \cdot B^2 f^2 \quad (2.8)$$

Konstanta k yang sangat menentukan besar kecilnya rugi daya. Konstanta k ini bukan nilai tetap, melainkan bergantung pada karakteristik fisik dan material dari medium tempat arus eddy terbentuk.

2.6 Skin Depth dan Skin Effect

Skin effect adalah fenomena di mana arus listrik pada konduktor cenderung mengalir di permukaan luar kawat, terutama saat dialiri arus bolak-balik (AC) dengan frekuensi tinggi. Akibatnya, arus tidak lagi tersebar merata ke seluruh penampang kawat, melainkan terkonsentrasi di lapisan luar (permukaan), sehingga efektif luas penampang untuk penghantaran arus menjadi lebih kecil.

Skin depth adalah ukuran seberapa jauh arus listrik atau gelombang elektromagnetik dapat menembus ke dalam suatu konduktor sebelum mengalami pelemahan yang signifikan (Shukla, 2022). Besarnya *skin depth* sangat dipengaruhi oleh sifat fisis konduktor, yaitu resistivitas (ρ), permeabilitas magnetik relatif (μ_r), serta frekuensi sinyal AC yang digunakan (f). Semakin tinggi frekuensi (f), maka *skin depth* semakin kecil, sehingga arus semakin terbatas hanya di permukaan. Material dengan konduktivitas tinggi dan permeabilitas tinggi (seperti tembaga atau besi lunak) akan memiliki *skin depth* yang lebih kecil. Persamaan untuk menghitung *skin depth* diberikan oleh:

$$\delta = \sqrt{\frac{\rho}{\pi f \mu_0 \mu_r}} \quad (2.9)$$

Adapun persamaan *skin depth* yang mempertimbangkan keberadaan medium dituliskan sebagai berikut, dengan menggunakan nilai konduktivitas medium yang telah diketahui sebelumnya.

$$\delta = 503 \sqrt{\frac{1}{f \sigma}} \quad (2.10)$$

Resistansi efektif akibat *skin effect* dapat dihitung menggunakan:

$$R_{eff} \approx \frac{\rho}{2\pi r \delta} \quad (2.11)$$

2.7 Efisiensi Transfer Daya

Transfer daya nirkabel menggunakan prinsip induksi elektromagnetik, sebagaimana dijelaskan di atas. Pada transfer daya nirkabel ini, gaya gerak listrik (GGL) atau tegangan induksi (E) dihasilkan oleh perubahan fluks magnetik (Φ_B) di kumparan penerima. Tegangan induksi ini selanjutnya digunakan untuk mengalirkan arus listrik, sesuai dengan hukum Ohm:

$$V = I \cdot R \quad (2.12)$$

$$P = I \cdot V \quad (2.13)$$

Pada transfer daya nirkabel, efisiensi transfer daya dihitung sebagai rasio daya yang diterima (P_{out}) oleh kumparan penerima terhadap daya yang ditransmisikan (P_{in}) oleh kumparan pengirim:

$$\eta = \frac{P_{out}}{P_{in}} \cdot 100\% \quad (2.14)$$

Metode Percobaan

3.1 Komponen yang Digunakan

Tabel 3.1: *Komponen yang Digunakan*

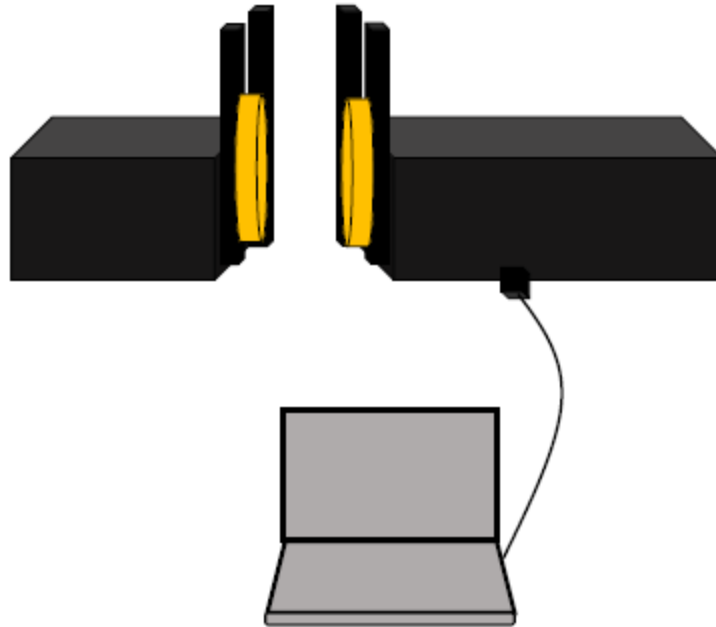
No.	Nama Komponen	Jumlah
1.	Arduino Uno + Kabel Serial	1 buah
2.	Transistor 1RFZ44N MOSFET	1 buah
3.	Resistor 10 k Ω	2 buah
4.	Resistor 1 k Ω	4 buah
5.	Resistor 47 Ω	1 buah
6.	Resistor 100 Ω	1 buah

7.	Resistor 220 Ω	1 buah
8.	Resistor 10 Ω	1 buah
9.	Trimpot 20k Ω	1 buah
10.	Trimpot 5k Ω	1 buah
11.	Kapasitor 220 μF	1 buah
12.	Kapasitor 0,1 μF	2 buah
13.	Kapasitor 0,001 μF	1 buah
14.	Kapasitor 0,0047 μF	1 buah
15.	Kapasitor 10 μF	1 buah
16.	NE555	1 buah
17.	LM317T	1 buah
18.	1N4007	1 buah
19.	Kawat Tembaga 0,5 mm	Secukupnya
20.	Kabel Jack	1 buah
21.	Female Jack	2 buah
22.	Kabel Jumper	Secukupnya
23.	Papan PCB	Secukupnya
24.	Timah	Secukupnya
25.	Pin Connector dan Inner	Secukupnya
26.	Akrilik	Secukupnya
27.	Mur dan Baut	Secukupnya
28.	NaCl	1 gram
29.	Agar-Agar	7 gram
30.	Air	100 ml

3.2 Prosedur Percobaan

Untuk melakukan analisis transfer daya nirkabel, terdapat beberapa langkah berikut:

1. Persiapkan komponen yang diperlukan seperti yang terdapat pada tabel di atas.
2. Membuat desain mekantronika sistem transfer daya nirkabel. Desain dapat dibuat dengan menggunakan software corel atau sejenisnya untuk *casing* sistem transfer daya nirkabel yang akan digunakan.

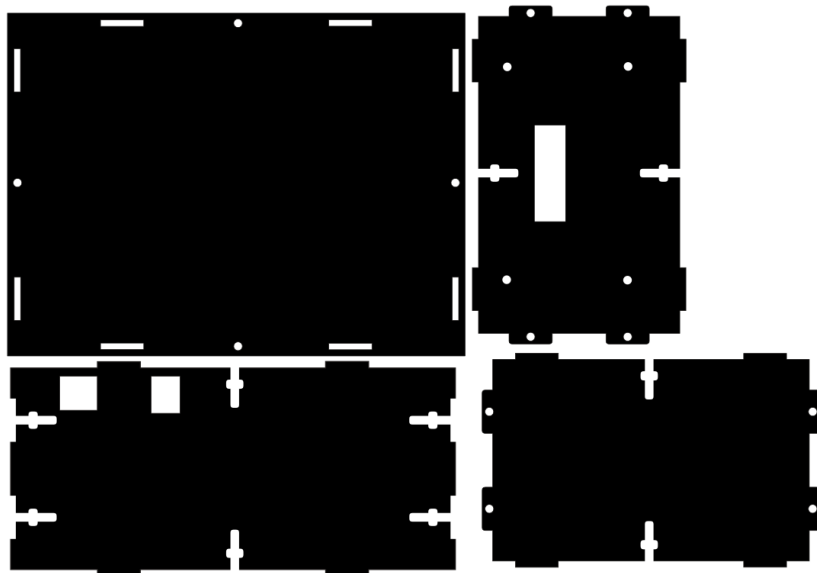


Gambar 3.1: *Prototype* Desain Sistem Transfer Daya Nirkabel





Gambar 3.2: *Casing Rangkaian Transmitter*



5. Membuat program

Gunakan Arduino IDE serta Python untuk membuat skrip program. Berikut program yang digunakan.

Program Arduino IDE	
	<pre>// Sensor tegangan int Voltmeter_Receiver = A0; // pin tegangan receiver int Amperemeter_Receiver = A1; // pin arus receiver float R1 = 10000; // Resistor atas float R2 = 1000; // Resistor bawah float R4 = 10; // Resistor untuk pengukuran arus receiver bool isMeasuring = false; // Status pengukuran void setup() { Serial.begin(9600); pinMode(Voltmeter_Receiver, INPUT); pinMode(Amperemeter_Receiver, INPUT); } void loop() { // Cek apakah ada data dari Serial if (Serial.available() > 0) { int Marissa = Serial.read(); if (Marissa == 'M' Marissa == 'm') { isMeasuring = true; } else if (Marissa == 'B' Marissa == 'b') { isMeasuring = false; } } // Jika pengukuran sedang aktif if (isMeasuring) { // Nilai tetap untuk Transmitter float V_Transmitter = 5.0; // Tegangan transmitter tetap 5V float I_Transmitter = 0.5; // Arus transmitter tetap 0.5A // Mengukur tegangan Receiver float adc3 = analogRead(Voltmeter_Receiver);</pre>

```

float Vadc3 = 5.0 * adc3 / 1023.0;
float V_Receiver = Vadc3 * (R1 / R2 + 1);

// Mengukur arus Receiver
float adc4 = analogRead(Amperemeter_Receiver);
float Vadc4 = 5.0 * adc4 / 1023.0;
float I_Receiver = Vadc4 / R4;

// Menampilkan hasil pengukuran
Serial.println(V_Transmitter, 6); // Tegangan transmitter
Serial.println(I_Transmitter, 6); // Arus transmitter
Serial.println(V_Receiver, 6);    // Tegangan receiver
Serial.println(I_Receiver, 6);    // Arus receiver

delay(1000); // Delay 1 detik
}
}

```

Program Python

```

from matplotlib.backends.backend_tkagg import FigureCanvasTkAgg
from matplotlib.figure import Figure
import tkinter as tk
import serial as sr
import collections
from matplotlib.lines import Line2D
import time
import csv

# Mendapatkan timestamp awal dalam detik
start_time = None

#-----Memulai setting port serial-----
serial_port = sr.Serial('COM7', 9600, timeout=1) # Pastikan port sesuai
serial_port.reset_input_buffer()

Samples = 50 # Jumlah sampel yang muncul pada grafik
numData = 4 # Jumlah data tiap pengiriman dari Arduino (tegangan dan
            arus transmitter dan receiver)
cond = False

# Variabel untuk menyimpan total daya, tegangan, arus, dan efisiensi
total_power_transmitter = 0
total_power_receiver = 0
total_efficiency = 0
total_voltage_transmitter = 0

```

```

total_current_transmitter = 0
total_voltage_receiver = 0
total_current_receiver = 0
measurement_count = 0

# Membuat file CSV untuk menyimpan data
t = time.localtime()
nama_file = time.strftime("%Y_%m_%d_%H_%M_%S", t)
csv_file_path = f"{nama_file}_(Implementasi Pacemaker_125k Hz, 1cm,
0').csv"

with open(csv_file_path, 'w', newline='') as csvfile:
    fieldnames = ['Time (s)', 'V_Transmitter', 'I_Transmitter',
'V_Receiver', 'I_Receiver', 'Power_Transmitter', 'Power_Receiver',
'Efficiency']
    writer = csv.DictWriter(csvfile, fieldnames=fieldnames,
delimiter=';')
    writer.writeheader()

#-----Fungsi plotting data-----
def plot_data():
    global cond, data, lines, total_power_transmitter,
total_power_receiver, total_efficiency, measurement_count
    global total_voltage_transmitter, total_current_transmitter,
total_voltage_receiver, total_current_receiver

    if cond:
        waktu = get_current_time() # Menghitung waktu dalam detik

        try:
            # Membaca data dari serial
            serial_data = [float(serial_port.readline().decode('utf-
8')).strip()) for _ in range(numData)]

            for i in range(numData):
                data[i].append(serial_data[i]) # Simpan data di posisi
akhir
                lines[i].set_data(range(Samples), data[i]) # Perbarui
garis

            # Menghitung daya transmitter dan receiver
            V_Transmitter = serial_data[0]
            I_Transmitter = serial_data[1]
            V_Receiver = serial_data[2]
            I_Receiver = serial_data[3]

```

```

        Power_Transmitter = V_Transmitter * I_Transmitter
        Power_Receiver = V_Receiver * I_Receiver

        # Menghitung efisiensi
        if Power_Transmitter != 0:
            Efficiency = (Power_Receiver / Power_Transmitter) * 100
        else:
            Efficiency = 0

        # Menyimpan total daya, tegangan, arus, dan efisiensi untuk
        perhitungan rata-rata
        total_voltage_transmitter += V_Transmitter
        total_current_transmitter += I_Transmitter
        total_voltage_receiver += V_Receiver
        total_current_receiver += I_Receiver
        total_power_transmitter += Power_Transmitter
        total_power_receiver += Power_Receiver
        total_efficiency += Efficiency
        measurement_count += 1

        # Menyimpan data ke file CSV
        with open(csv_file_path, 'a', newline='') as csvfile:
            writer = csv.writer(csvfile, delimiter=';')
            writer.writerow([f"{waktu:.1f}", V_Transmitter,
I_Transmitter, V_Receiver, I_Receiver, Power_Transmitter,
Power_Receiver, Efficiency])

        # Hitung dan tampilkan rata-rata tegangan dan arus
        calculate_averages()

    except Exception as e:
        print(f"Error reading data: {e}")

    # Perbarui batasan sumbu
    for ax in axes:
        ax.relim()
        ax.autoscale_view()

    # Gambar ulang canvas
    canvas.draw()

    root.after(1000, plot_data) # Perbarui setiap 1 detik

# Fungsi untuk memulai pengukuran

```

```

def start_measurement():
    global cond, start_time
    if not cond:
        serial_port.write(b'M') # Kirim perintah 'M' ke Arduino
        start_time = time.time()
        cond = True
        serial_port.reset_input_buffer()

# Fungsi untuk menghentikan pengukuran dan menghitung rata-rata daya dan efisiensi
def stop_measurement():
    global cond, total_power_transmitter, total_power_receiver,
    total_efficiency, measurement_count
    global total_voltage_transmitter, total_current_transmitter,
    total_voltage_receiver, total_current_receiver

    # Menghitung rata-rata daya dan efisiensi
    if measurement_count > 0:
        avg_power_transmitter = total_power_transmitter /
measurement_count
        avg_power_receiver = total_power_receiver / measurement_count
        avg_efficiency = total_efficiency / measurement_count
    else:
        avg_power_transmitter = avg_power_receiver = avg_efficiency = 0

    # Tampilkan hasil rata-rata di GUI
    avg_power_transmitter_label.config(text=f"Avg Power Transmitter:
{avg_power_transmitter:.4f} W")
    avg_power_receiver_label.config(text=f"Avg Power Receiver:
{avg_power_receiver:.4f} W")
    avg_efficiency_label.config(text=f"Avg Efficiency:
{avg_efficiency:.2f}%")

    # Hitung dan tampilkan rata-rata tegangan dan arus setelah
penghentian pengukuran
    calculate_averages()

    # Kirim perintah 'B' untuk menghentikan pengukuran pada Arduino
    serial_port.write(b'B')
    cond = False

# Fungsi untuk mendapatkan waktu dalam detik
def get_current_time():
    return time.time() - start_time if start_time else 0

```



```

# Fungsi untuk menghitung rata-rata tegangan dan arus
def calculate_averages():
    global total_voltage_transmitter, total_current_transmitter,
    total_voltage_receiver, total_current_receiver, measurement_count

    if measurement_count > 0:
        avg_voltage_transmitter = total_voltage_transmitter /
measurement_count
        avg_current_transmitter = total_current_transmitter /
measurement_count
        avg_voltage_receiver = total_voltage_receiver /
measurement_count
        avg_current_receiver = total_current_receiver /
measurement_count
    else:
        avg_voltage_transmitter = avg_current_transmitter =
avg_voltage_receiver = avg_current_receiver = 0

    # Tampilkan rata-rata pada GUI
    avg_voltage_transmitter_label.config(text=f"Avg Voltage Transmitter:
{avg_voltage_transmitter:.4f} V")
    avg_current_transmitter_label.config(text=f"Avg Current Transmitter:
{avg_current_transmitter:.4f} A")
    avg_voltage_receiver_label.config(text=f"Avg Voltage Receiver:
{avg_voltage_receiver:.4f} V")
    avg_current_receiver_label.config(text=f"Avg Current Receiver:
{avg_current_receiver:.4f} A")

#-----Kode utama GUI-----
root = tk.Tk()
root.title('Real Time Plot')
root.configure(background='blue')
root.geometry("900x700") # Memperbesar tampilan untuk ruang antar
subplot

# Batas grafik
xmin = 0
xmax = Samples
ymin = [0, 0, 0, 0]
ymax = [10, 1, 5, 1] # Disesuaikan untuk tegangan dan arus

lines = []
data = []

# Buat deque untuk menyimpan data

```

```

for _ in range(numData):
    data.append(collections.deque([0] * Samples, maxlen=Samples))
    lines.append(Line2D([], []))

# Membuat plot
fig = Figure(figsize=(8, 10)) # Memperbesar ukuran figure
fig.subplots_adjust(hspace=0.8) # Menambahkan lebih banyak jarak antar
subplot
axes = []

titles = ['Tegangan Transmitter (V)', 'Arus Transmitter (A)', 'Tegangan
Receiver (V)', 'Arus Receiver (A)']
for i in range(numData):
    ax = fig.add_subplot(4, 1, i + 1, xlim=(xmin, xmax), ylim=(ymin[i],
ymax[i]))
    ax.title.set_text(titles[i])
    ax.set_xlabel("Waktu (s)")
    ax.set_ylabel("Nilai")
    ax.add_line(lines[i])
    axes.append(ax)

canvas = FigureCanvasTkAgg(fig, master=root)
canvas.get_tk_widget().pack(side="left", padx=10, pady=10)

# Membuat Frame untuk tombol di sebelah kanan
button_frame = tk.Frame(root)
button_frame.pack(side="right", padx=10, pady=10)

# Tombol Start untuk memulai pengukuran
start_button = tk.Button(button_frame, text="Start",
command=start_measurement, bg="green", fg="white", width=20)
start_button.pack(side="top", pady=10)

# Tombol Stop untuk menghentikan pengukuran
stop_button = tk.Button(button_frame, text="Stop",
command=stop_measurement, bg="red", fg="white", width=20)
stop_button.pack(side="top", pady=10)

# Label untuk menampilkan rata-rata daya dan efisiensi
avg_power_transmitter_label = tk.Label(button_frame, text="Avg Power
Transmitter: N/A", bg="lightgrey", width=30)
avg_power_transmitter_label.pack(pady=5)

avg_power_receiver_label = tk.Label(button_frame, text="Avg Power
Receiver: N/A", bg="lightgrey", width=30)

```

```

avg_power_receiver_label.pack(pady=5)

avg_efficiency_label = tk.Label(button_frame, text="Avg Efficiency:
N/A", bg="lightgrey", width=30)
avg_efficiency_label.pack(pady=5)

# Label untuk menampilkan rata-rata tegangan dan arus
avg_voltage_transmitter_label = tk.Label(button_frame, text="Avg Voltage
Transmitter: N/A", bg="lightgrey", width=30)
avg_voltage_transmitter_label.pack(pady=5)

avg_current_transmitter_label = tk.Label(button_frame, text="Avg Current
Transmitter: N/A", bg="lightgrey", width=30)
avg_current_transmitter_label.pack(pady=5)

avg_voltage_receiver_label = tk.Label(button_frame, text="Avg Voltage
Receiver: N/A", bg="lightgrey", width=30)
avg_voltage_receiver_label.pack(pady=5)

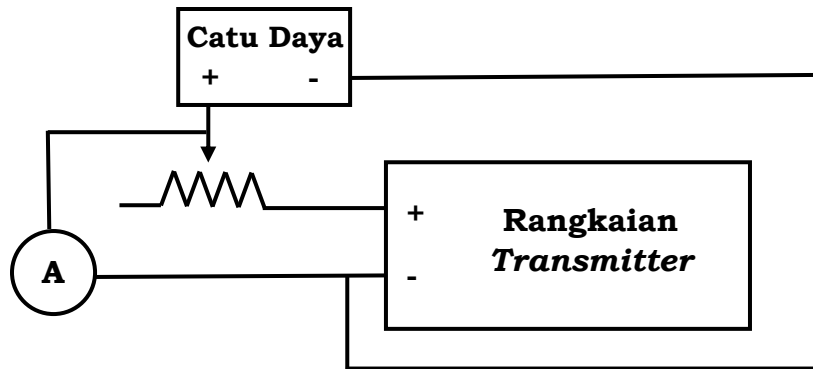
avg_current_receiver_label = tk.Label(button_frame, text="Avg Current
Receiver: N/A", bg="lightgrey", width=30)
avg_current_receiver_label.pack(pady=5)

root.after(1000, plot_data) # Memulai plotting data
root.mainloop()

```

6. Menjalankan Program dan Pengujian:

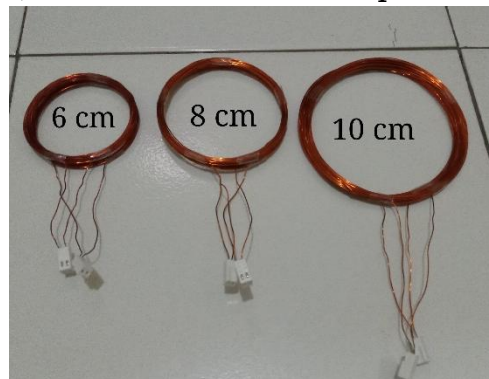
- Masukkan kode program Arduin IDE di atas dan running kode program yang telah terhubung pada rangkaian.
- Masukkan kode program Python diatas yang telah terintegrasi dengan program Arduino IDE sebelumnya pada software Python, lakukan running kode program yang telah terhubung dengan rangkaian dan lakukan pengujian.
- Pengujian dilakukan dengan beberapa variasi diantaranya:
 - Variasi Arus
Gunakan resistor variabel diantara sumber daya dengan rangkain *transmitter* seperti pada gambar di bawah ini.



Gambar 3.5: Skematik Pengujian Variasi Arus

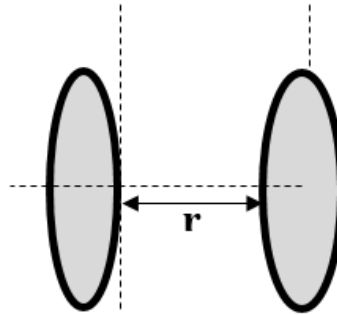
Arus yang digunakan pada sistem mulai dari 0.1 A, 0.3 A, dan 0.5 A pada jarak 1 cm, frekuensi sistem 100 kHz, sudut antar kumparan 0° dan diameter kumparan 6 cm.

- Variasi diameter kumparan
Gunakan diameter kumparan 6 cm, 8 cm dan 10 cm pada kumparan primer maupun sekunder dengan jumlah lilitan yang sama. Arus yang digunakan 0.5 A, jarak 1 cm, frekuensi sistem 100 kHz, dan sudut antar kumparan 0° .



Gambar 3.6: Diameter Kumparan

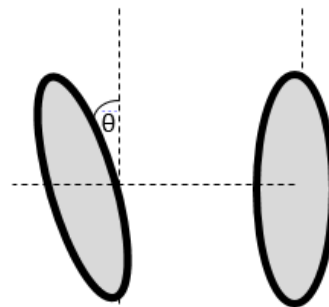
- Variasi Jarak
Pada variasi ini digunakan arus yang digunakan 0.5 A, sudut antar kumparan 0° , dan diameter kumparan 6 cm karna paling stabil nilai efisiensinya. Variasi jarak dilakukan mulai dari jarak terdekat hingga terjauh energi dapat ditransfer. Pada variasi ini digunakan frekuensi sistem 100 kHz, 125 kHz, dan 150 kHz.



Gambar 3.7: Ilustrasi Jarak Antar Kumparan

- Variasi Sudut

Pada variasi ini digunakan arus yang digunakan 0.5 A, jarak antar kumparan 1 cm, dan diameter kumparan 6 cm. Variasi sudut dilakukan mulai dari 0° hingga 90°. Pada variasi ini digunakan frekuensi sistem 100 kHz, 125 kHz, dan 150 kHz.



Gambar 3.8: Ilustrasi Sudut Antar Kumparan

- Implementasi sitem pada *pacemaker*

Gunakan data yang masuk pada spesifikasi dari *pacemaker* yakni 2.9 volt dengan kapasitan 1.0 Ah. Maka tegangan *receiver* yang memiliki nilai minimum 2.9 volt dapat dilakukan pengujian lanjutan dengan keberadaan medium antar kumparannya. Medium yang digunakan yakni 1 gram NaCl dan 7 gram agar-agar yang dilarutkan pada 100 mL air.

Ukur resistansi medium dengan multimeter dan hasilnya dimasukkan pada persamaan berikut untuk menghitung konduktivitas medium yang digunakan.

$$\sigma = \frac{1}{\rho} = \frac{L}{R \cdot \pi r^2} \quad (3.1)$$

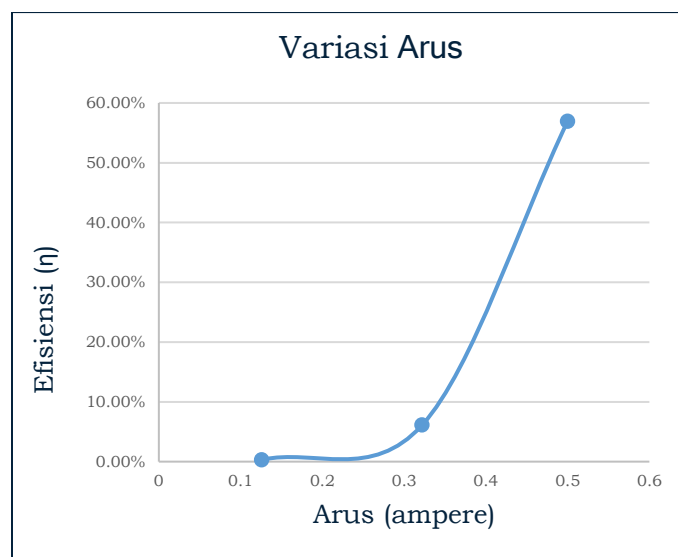
Dengan L jarak antar *probenya* (dalam meter), r jari-jari ujung *probe* multimeter, dan R resistansi yang terbaca oleh multimeter untuk medium.

Hitung *skin depth* karena adanya medium dengan persamaan (2.10) untuk memastikan arus dapat sampai pada kumparan penerima.

Data dan Analisis Pengujian

Tabel 4.1: Data Variasi Arus

d	f	θ	r	V_T	I_T	P_T	V_R	I_R	P_R	η
6 cm	100 kHz	0°	1 cm	5 V	0,5 A	2,5 W	3,7130 V	0,3833 A	1,4237 W	56,95%
6 cm	100 kHz	0°	1 cm	5 V	0,322 A	1,61 W	0,9333 V	0,1022 A	0,0984 W	6,11%
6 cm	100 kHz	0°	1 cm	5 V	0,126 A	0,63 W	0,1086 V	0,0182 A	0,0020 W	0,32%



Gambar 4.1: Grafik Perubahan Arus Terhadap Efisiensi

Dari data pada Tabel 4.1, dapat dianalisis bahwa variasi arus pada rangkaian pengirim memiliki dampak signifikan terhadap daya yang diterima oleh rangkaian penerima dan efisiensinya. Semakin kecil arus yang mengalir, semakin rendah efisiensi yang dihasilkan.

Fenomena ini dapat dijelaskan dengan konsep induksi elektromagnetik, di mana medan magnet yang dihasilkan oleh kumparan pengirim berbanding lurus dengan arus yang mengalir, seperti pada persamaan (2.4). Saat arus yang diberikan ke kumparan pengirim lebih besar, medan magnet yang dihasilkan lebih kuat, sehingga lebih banyak daya dapat diinduksikan ke kumparan penerima. Sebaliknya, saat arus menurun, medan magnet yang terbentuk semakin lemah, mengurangi jumlah fluks magnet yang memotong kumparan penerima, sehingga daya yang dapat ditransfer berkurang secara signifikan.

Pada sistem WPT berbasis induksi magnetik, daya yang ditransfer ke kumparan penerima (P_{out}) berasal dari interaksi antara medan magnet yang dihasilkan oleh kumparan pengirim dan arus yang mengalir pada kumparan penerima. Proses ini bisa dijelaskan melalui dasar hukum elektromagnetisme, yaitu Hukum Ampere dan Hukum Faraday. Menurut Hukum Ampere, arus listrik (I) yang mengalir pada kumparan pengirim akan menghasilkan medan magnet (B) yang besarnya berbanding lurus dengan arus tersebut, sehingga $B \propto I$. Selanjutnya, menurut Hukum Faraday, perubahan fluks magnetik pada kumparan penerima akan menginduksi tegangan (V_{ind}) pada kumparan tersebut, dengan besarnya tegangan terinduksi bergantung pada medan magnet yang dihasilkan oleh kumparan pengirim. Karena medan magnet (B) bergantung pada arus (I), maka tegangan yang terinduksi juga berbanding lurus dengan arus. Daya yang diterima oleh kumparan penerima merupakan hasil kali tegangan induksi dan arus yang dihasilkan pada kumparan penerima seperti yang terdapat pada persamaan (2.16). Karena $V_{ind} \propto B \propto I$, maka daya yang dihasilkan menjadi $P_{out} \propto B \cdot I \propto I \cdot I = I^2$. Inilah alasan mengapa grafik daya yang dihasilkan terhadap arus membentuk kurva kuadratik, karena hubungan antara arus dan medan magnet yang dihasilkan bersifat linier, sehingga daya yang dihasilkan akan mengikuti pola kuadratik terhadap arus pengirim. Oleh karena itu, semakin besar arus pada kumparan pengirim, medan magnet yang dihasilkan akan semakin kuat, menyebabkan daya yang dihasilkan meningkat secara kuadratik.

Tabel 4.9: Data Variasi Sebelum Terdapat Medium

d	f	θ	r	V_T	I_T	P_T	V_R	I_R	P_R	η
6 cm	100 kHz	0°	1 cm	5 V	0,5 A	2,5 W	3,7130 V	0,3833 A	1,4237 W	56,95%
6 cm	100 kHz	0°	1,2 cm	5 V	0,5 A	2,5 W	3,5572 V	0,3644 A	1,2964 W	51,86%
6 cm	100 kHz	0°	1,4 cm	5 V	0,5 A	2,5 W	3,4826 V	0,3586 A	1,2490 W	49,96%

6 cm	100 kHz	0°	1,6 cm	5 V	0,5 A	2,5 W	3,2247 V	0,3308 A	1,0666 W	42,67%
6 cm	100 kHz	0°	1,8 cm	5 V	0,5 A	2,5 W	3,1468 V	0,3220 A	1,0131 W	40,52%
6 cm	100 kHz	0°	2 cm	5 V	0,5 A	2,5 W	3,0261 V	0,3109 A	0,9409 W	37,64%
6 cm	100 kHz	2°	1 cm	5 V	0,5 A	2,5 W	3,4387 V	0,3496 A	1,2023 W	48,09%
6 cm	100 kHz	4°	1 cm	5 V	0,5 A	2,5 W	3,1709 V	0,3275 A	1,0383 W	41,53%
6 cm	100 kHz	6°	1 cm	5 V	0,5 A	2,5 W	3,0963 V	0,3194 A	0,9888 W	39,55%
6 cm	100 kHz	8°	1 cm	5 V	0,5 A	2,5 W	2,9098 V	0,3005 A	0,8744 W	34,98%
6 cm	125 kHz	0°	1 cm	5 V	0,5 A	2,5 W	2,9658 V	0,3037 A	0,9006 W	36,03%

Tabel 4.10: Data Variasi Setelah Terdapat Medium

d	f	θ	r	V_T	I_T	P_T	V_R	I_R	P_R	η
6 cm	100 kHz	0°	1 cm	5 V	0,5 A	2,5 W	3,4409 V	0,3505 A	1,2060 W	48,24%
6 cm	100 kHz	0°	1,2 cm	5 V	0,5 A	2,5 W	3,3432 V	0,3442 A	1,1508 W	46,03%
6 cm	100 kHz	0°	1,4 cm	5 V	0,5 A	2,5 W	3,2247 V	0,3340 A	1,0770 W	43,08%
6 cm	100 kHz	0°	1,6 cm	5 V	0,5 A	2,5 W	3,1216 V	0,3194 A	0,9970 W	39,88%
6 cm	100 kHz	0°	1,8 cm	5 V	0,5 A	2,5 W	3,0963 V	0,3169 A	0,9811 W	39,24%
6 cm	100 kHz	0°	2 cm	5 V	0,5 A	2,5 W	2,7573 V	0,2860 A	0,7884 W	31,54%
6 cm	100 kHz	2°	1 cm	5 V	0,5 A	2,5 W	3,1644 V	0,3210 A	1,0159 W	40,64%
6 cm	100 kHz	4°	1 cm	5 V	0,5 A	2,5 W	3,0843 V	0,3161 A	0,9751 W	39,00%
6 cm	100 kHz	6°	1 cm	5 V	0,5 A	2,5 W	2,8714 V	0,2951 A	0,8475 W	33,90%
6 cm	100 kHz	8°	1 cm	5 V	0,5 A	2,5 W	2,5151 V	0,2605 A	0,6557 W	26,23%

6 cm	125 kHz	0°	1 cm	5 V	0,5 A	2,5 W	2,4215 V	0,2541 A	0,6153 W	24,61%
------	---------	----	------	-----	-------	-------	----------	----------	----------	--------

Berdasarkan data pada Tabel 4.9, hasil pengukuran menunjukkan bahwa nilai yang diperoleh masih sesuai dengan spesifikasi perangkat *pacemaker*. Perangkat *pacemaker* sendiri memiliki kapasitas 2.9 Volt dan 1.0 Ah. Pada implementasinya tentunya ada penghalang antara kumparan pengirim dan penerima. Dalam hal ini meliputi kulit, lemak dan otot. Namun, untuk melakukan implementasi secara langsung pada tubuh tentunya sangat kompleks. Oleh karena itu, dilakukan percobaan lanjutan dengan menambahkan media penghalang di antara kumparan dengan menggunakan campuran larutan NaCl 1% dengan agar-agar untuk mensimulasikan jaringan manusia dengan ketebalam 10 mm yang merepresentasikan kulit (3 mm), lemak (2 mm) dan otot (5 mm). Hasil pengujian setelah penambahan media ditunjukkan pada Tabel 4.10, di mana terjadi penurunan efisiensi transfer daya.

Penurunan efisiensi transfer daya pada sistem WPT yang diimplementasikan menggunakan medium penghalang berupa larutan NaCl dan agar-agar setebal 10 mm menunjukkan kompleksitas interaksi antara konduktivitas medium, orientasi sudut kumparan, jarak antar kumparan, dan frekuensi kerja. Konduktivitas medium memengaruhi penyebaran medan magnet dan meningkatkan pembentukan arus eddy (arus pusar). Arus eddy ini menyebabkan rugi-rugi energi dalam bentuk panas, yang pada akhirnya mengurangi efisiensi transfer daya. Selain itu, medium dengan konduktivitas lebih tinggi juga dapat meningkatkan efek pelemahan medan magnet, sehingga menghambat proses induksi pada kumparan penerima.

Dalam kondisi frekuensi tetap, perubahan sudut orientasi antara kumparan *transmitter* dan *receiver* menyebabkan penurunan efisiensi yang lebih signifikan dibandingkan saat sudut 0°. Hal ini terjadi karena fluks magnet yang dihasilkan tidak lagi memotong kumparan penerima secara optimal, sehingga GGL yang diinduksi berkurang.

Begitu pula saat jarak antara kumparan diperbesar, efisiensi transfer daya juga menurun secara signifikan dibandingkan saat kedua kumparan berada pada jarak 1 cm. Hal ini disebabkan oleh lemahnya kopling magnetik seiring bertambahnya jarak, ditambah lagi dengan redaman medan magnet oleh medium penghalang yang konduktif.

Sementara itu, saat sudut dan jarak dijaga tetap, namun frekuensi diubah ke nilai yang lebih tinggi, efisiensi juga mengalami penurunan. Hal ini menunjukkan bahwa medium penghalang yang memiliki konduktivitas tinggi seperti larutan NaCl lebih merespon terhadap frekuensi tinggi dengan meningkatkan rugi-rugi akibat arus eddy dan skin effect. Akibatnya, daya yang

berhasil ditransfer ke kumparan penerima menjadi lebih kecil dibandingkan saat sistem bekerja pada frekuensi yang lebih rendah.

Secara keseluruhan, hasil implementasi ini menunjukkan bahwa keberadaan medium penghalang tidak hanya menimbulkan rugi-rugi akibat konduktivitasnya, tetapi juga memperparah dampak perubahan orientasi sudut, jarak, dan frekuensi terhadap efisiensi sistem. Oleh karena itu, desain sistem WPT yang melibatkan medium biologis atau konduktif seperti jaringan tubuh perlu mempertimbangkan ketiga faktor tersebut untuk meminimalkan penurunan efisiensi daya yang ditransfer.

Hasil penelitian yang diimplementasikan dengan medium memiliki efisiensi terbesar 48.24% dengan tegangan 3,4409 Volt dan arusnya 0,3505 Ampere, nilai ini sudah bisa diimplementasikan untuk pengisian daya baterai *pacemaker*. Namun, arus yang dihasilkan masih kecil yang berdampak pada durasi pengisian baterai yang cukup lama. Sehingga diperlukan optimasi lebih lanjut untuk menghasilkan arus yang lebih besar.

Tabel 4.4: Data Variasi Jarak

d	f	θ	r	V_T	I_T	P_T	V_R	I_R	P_R	η

Referensi

- Andesta, R. (2018). *Rancang Bangun Prototipe Wireless Power Transfer (WPT) Menggunakan Induktor Planar Untuk Peralatan Elektronika Berdaya Rendah*.
- Ariyanto, A. (2016). *Studi Pengaruh Tensile Stress Terhadap Nilai Hambatan Kawat Penghantar*.
- Detka, K., & Górecki, K. (2022). Wireless Power Transfer—A Review. *Energies*, 15(19), 1–21. <https://doi.org/10.3390/en15197236>
- Diharjo, S. J. (2018). *Rancang Bangun Transfer Daya Listrik Nirkabel Beban DC*

Menggunakan Induksi Elektromagnetik. In *Universitas Islam Indonesia, Yogyakarta*.

Fadhliyansyah, M. (2017). Analisis Perhitungan Rugi-Rugi Transformator Akibat Harmonisa (Studi Kasus Gardu Distribusi Smti Pontianak). *Trafodaya.*, 0(6), 1–7.

<http://jurnal.untan.ac.id/index.php/jteuntan/article/viewFile/30300/75676579524>

Firmansyah, D., Kusumawardani, M., & Heru, Y. (2020). Analisis Pengaruh Jumlah Lilitan pada Wireless Charger Smartphone Terhadap Daya Output Wireless Charger. *Jurnal JARTEL*, 1(1), 20–25.
<https://jartel.polinema.ac.id/index.php/jartel/article/view/162%0Ahttps://jartel.polinema.ac.id/index.php/jartel/article/download/162/78>

Patil, D., McDonough, M. K., Miller, J. M., Fahimi, B., & Balsara, P. T. (2017). Wireless Power Transfer for Vehicular Applications: Overview and Challenges. *IEEE Transactions on Transportation Electrification*, 4(1), 3–37.
<https://doi.org/10.1109/TTE.2017.2780627>

Shukla, D. A. (2022). Design and construction of a reliable wireless power transfer system for an embedded device. In *KTH Royal Institute of Technology*.