

Penyelesaian *Plotting Smithchart* untuk Impedansi Input dengan Pemrograman Berbasis GUI Python

Muhammad Roby Meidiansyah, Khaila Oktavianingsih*, Aisyah Afiyanti, Viving Frendiana

Broadband Multimedia, Teknik Elektro, Politeknik Negeri Jakarta, Depok

*khaila.oktavianingsih.te22@mhs.wpnj.ac.id

Abstrak— Impedansi yang terhubung ke input rangkaian disebut dengan impedansi input atau biasa disimbolkan dengan Z_{in} . Perhitungan impedansi input dapat dilakukan dengan dua cara yaitu dengan cara matematis serta bisa dengan menggunakan smithchart. Smithchart merupakan alat bantu yang digunakan untuk menampilkan atau mengetahui karakteristik saluran transmisi seperti impedansi input. Penggunaan smithchart secara manual tentunya membutuhkan ketelitian yang baik agar hasil yang didapatkan benar dan akurat, hal tersebut tentunya membutuhkan proses serta waktu yang lebih lama. Oleh karena itu, pada project ini dirancang penyelesaian plotting smithchart untuk impedansi input dengan pemrograman berbasis GUI python yang bertujuan untuk mengembangkan sebuah program berbasis GUI Python yang dapat menyelesaikan plotting smithchart untuk menghitung nilai impedansi input. Program ini dibuat menggunakan dua modul yaitu modul plotly untuk visualisasi smithchart serta menggunakan modul tkinter untuk membuat antarmuka pengguna. Program ini dapat menampilkan hasil perhitungan nilai impedansi input serta dapat menampilkan diagram smithchart dengan cara pengguna akan menginput nilai impedansi beban, nilai impedansi karakteristik, nilai panjang kabel serta nilai frekuensi. Project ini diharapkan dapat memudahkan pengguna untuk menghitung nilai impedansi input serta menampilkan diagram smithchart tanpa harus melakukan perhitungan matematis. Selain itu, project ini juga diharapkan dapat membantu pengguna untuk mempelajari serta memahami smithchart dengan lebih efektif dan efisien.

Article History:

Received: Jan 29, 2024

Revised: July 16, 2024

Accepted: July 19, 2024

Published: Aug 9, 2024

Kata Kunci—Impedansi; Impedansi input; Pemrograman berbasis GUI python; Sistem transmisi; Smithchart;

DOI: 10.22441/jitkom.v8i2.001

I. PENDAHULUAN

Pengukuran impedansi merupakan proses dalam menentukan nilai impedansi pada suatu sistem atau komponen. Impedansi yang dilambangkan Z adalah hambatan listrik, suatu komponen terhadap arus AC atau Alternating Current (arus bolak-balik) dengan satuannya adalah ohm (Ω). Nilai impedansi yang besar menandakan resistansi yang lebih besar pada aliran listrik dan juga sebaliknya, nilai impedansi yang kecil menandakan resistansi yang lebih kecil pada aliran listrik. Pertimbangan impedansi sangat penting pada suatu rangkaian listrik. Impedansi listrik sering disebut juga sebagai jumlah hambatan listrik sebuah komponen elektronik terhadap aliran arus dalam rangkaian pada frekuensi tertentu. Rumus Impedansi adalah sebagai berikut :

$$Z = \sqrt{R^2 + (XL - XC)^2} \quad (1)$$

Penjelasan :

R : resistansi
 XL : induktansi
 XC : kapasitansi

Berdasarkan rumus tersebut, jika nilai X_L sama dengan X_C maka nilai impedansi akan sama dengan nilai resistansi ($Z = R$) dan impedansinya akan bersifat resistif [1].

Impedansi input (Z_{in}) adalah impedansi yang terhubung ke input rangkaian atau perangkat serta merupakan efek dari semua resistansi, kapasitansi, dan induktansi yang tergabung. Meskipun hanya terdapat resistansi, resistansi tersebut masih dapat dikatakan sebagai impedansi input [2].

Terdapat Frekuensi yang beragam dari efek kapasitansi dan induktansi, sehingga impedansi inputnya juga memiliki frekuensi yang beragam. Di antara resistansi, kapasitansi dan induktansi, pada frekuensi tinggi, efek kapasitansi dan induktansi berpengaruh paling signifikan [2].

Nilai impedansi input (masukan) setidaknya sepuluh kali impedansi output (keluaran) dari rangkaian yang mensuplai sinyal ke masukan, agar input tidak membebani sumber sinyal serta mengurangi kekuatan sinyal dalam jumlah besar maka impedansi masukan atau impedansi input harus tinggi [2].

Terdapat dua kondisi pada saluran transmisi, yaitu lossy (mengandung kerugian) dan lossless (tidak mengandung kerugian). Pada saluran transmisi yang lossy, rumus impedansi inputnya adalah sebagai berikut :

$$Z_{IN} = Z_0 \frac{Z_L + Z_0 \tanh(\gamma L)}{Z_0 + Z_L \tanh(\gamma L)} \quad (2)$$

$\tanh(\gamma L)$ dapat disederhanakan menjadi:

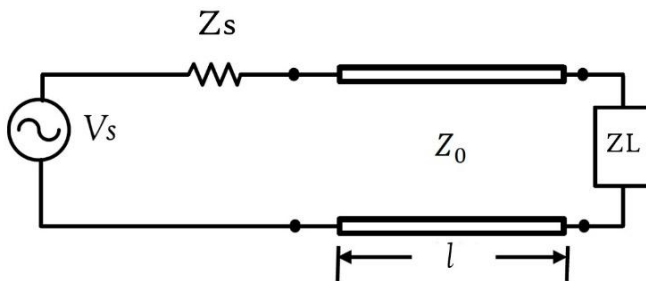
$$\tanh(\gamma L) = \frac{\tanh(\alpha L) + j \tan(\beta L)}{1 + j \tanh(\alpha L) + j \tan(\beta L)} \quad (3)$$

Sedangkan pada saluran transmisi lossless, nilai $\alpha = 0$, $\gamma = j\beta$, $\tanh(\gamma L) = \tanh(j\beta L) = j \tanh(\beta L)$, sehingga menghasilkan rumus impedansi input untuk saluran transmisi lossless adalah sebagai berikut:

$$Z_{IN} = Z_0 \frac{Z_L + j Z_0 \tan(\beta L)}{Z_0 + j Z_L \tan(\beta L)} \quad (4)$$

Penjelasan :

- γ : konstanta propagasi
- α : konstanta peredaman
- β : konstanta fasa, dimana nilai $\beta = \frac{2\pi}{\lambda}$
- Z_0 : impedansi karakteristik
- Z_L : impedansi beban
- l atau L : panjang saluran



Gambar 1. Impedansi Input [3]

Perhitungan impedansi input bisa dilakukan dengan dua cara yaitu dengan perhitungan matematis menggunakan rumus yang ada serta menggunakan bantuan *smithchart* [4]. Biasanya jika menggunakan bantuan *smithchart*, nilai impedansi input yang didapat akan bersifat aproksimasi. Dengan berkembangnya zaman, penentuan *plotting* untuk mendapatkan nilai impedansi input menggunakan *smithchart* dapat dilakukan dengan teliti karena penyelesaian *plotting smithchart* dilakukan dengan bantuan komputer. Pada penelitian pemrograman yang dilakukan ini, dirancang penyelesaian *plotting smithchart* untuk impedansi input dengan pemrograman berbasis GUI (*Graphical User Interface*) dari bahasa pemrograman Python dengan menggunakan modul *tkinter*. Sedangkan untuk menampilkan *plotting smithchart*nya digunakan modul Python yaitu *plotly*.

II. LITERATURE REVIEW

Penelitian yang diacu untuk penelitian ini, [4] merancang suatu program semacam kalkulator khusus yang dapat menghitung nilai *matching impedance* untuk saluran transmisi $\lambda/4$ (*quarterwave impedance*) dengan berbasis website. Sedangkan untuk penelitian ini, dirancang suatu program yang berisi kalkulator impedansi input dan kemudian menampilkan nilai *plotting smithchart* untuk nilai impedansi input

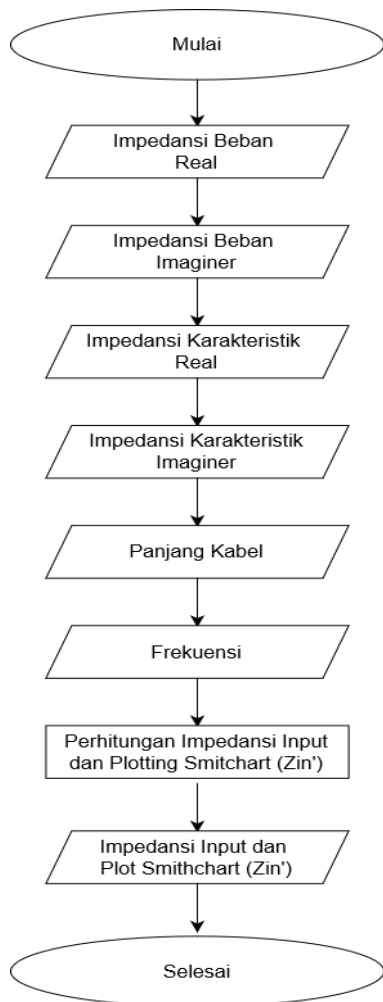
normalisasi dengan gambaran program pada [2] yang juga membahas pemrograman kalkulator impedansi input berbasis GUI Python menggunakan PyQt5. Pada penelitian [9] bertujuan untuk mengeksplorasi dan mensimulasikan penggunaan *Smith Chart* dalam penyesuaian impedansi menggunakan dua metode berbeda: trafo seperempat panjang gelombang ($\frac{1}{4} \lambda$ transformer) dan metode stub tunggal (*single stub*).

III. METODOLOGI PENELITIAN

Dalam menentukan nilai impedansi input menggunakan *smithchart* secara manual, dibutuhkan beberapa tahap sebelum akhirnya dapat dilakukan *plotting* untuk mendapatkan nilai impedansi inputnya. Secara garis besar, untuk menentukan nilai impedansi input menggunakan *smithchart*, perlu dilakukan beberapa tahapan yaitu menghitung nilai Impedansi beban normalisasi (Z_L') dengan rumus $Z_L' = \frac{Z_L}{Z_0}$ dan memplot nilainya, kemudian menghitung nilai sudut putar dengan rumus $-2\beta l = -\frac{2 \times 2\pi l}{\lambda}$ dalam satuan derajat (tanda negatif (-) menandakan perputaran sudut searah jarum jam) dan menempatkannya di *smithchart*, mencari titik impedansi input ternormalisasi (Z_{in}') dari penempatan sudut putar tersebut, dan mendenormalisasikan nilai Z_{in}' untuk mendapatkan nilai impedansi input yang asli (Z_{in}) ($Z_{in} = Z_{in}' \times Z_0$) [2]. Cara tersebut cukup sulit dilakukan dan memakan waktu yang cukup lama. Selain itu kemungkinan besar terjadi ketidakakuratan pada setiap tahapan terutama pada *plotting ZL'* dan penempatan sudut putarnya, sehingga nilai yang didapat juga tidak akurat. Dengan bantuan program kalkulator impedansi input dan plot *smithchart* ini, nilai yang didapat akan sesuai dengan perhitungan menggunakan rumus sehingga nilainya akurat.

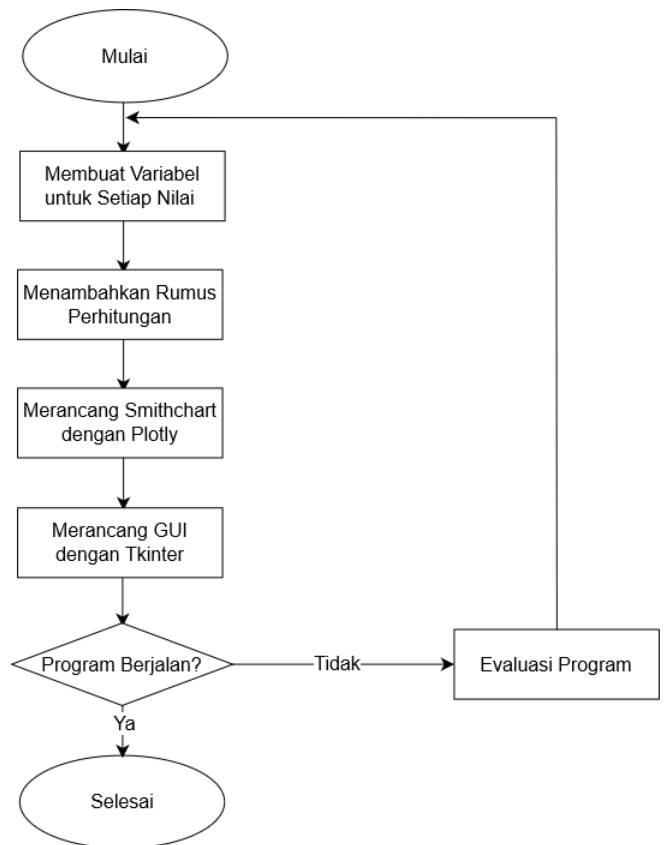
Perancangan program yang dilakukan didasari pada perhitungan secara matematis menggunakan rumus perhitungan, dimana rumus yang digunakan adalah rumus impedansi input untuk saluran lossless. Pada saluran lossy, terdapat nilai konstanta peredamannya (α), meskipun demikian, nilai α yang didapat akan cenderung mendekati 0 ($\alpha \approx 0$). Sehingga rumus perhitungan yang ditetapkan untuk digunakan pada program ini adalah rumus Z_{in} lossless.

Gambar 2 menampilkan flowchart perhitungan impedansi input yang kemudian akan menampilkan plot *smithchart* untuk impedansi input normalisasinya. Pada tahap pertama adalah menginput nilai-nilai yang dibutuhkan berdasarkan rumus Z_{in} lossless yaitu impedansi beban real dan imajiner, impedansi karakteristik real dan imajiner, panjang kabel, dan frekuensi. Dengan rumus $Z_{IN} = Z_0 \frac{Z_L + j Z_0 \tan(\beta L)}{Z_0 + j Z_L \tan(\beta L)}$, dimana nilai $\beta = \frac{2\pi}{\lambda}$ dan nilai $\lambda = \frac{c}{f}$ dengan c adalah kecepatan cahaya dengan besar $c = 3 \times 10^8$ m/s dan f adalah frekuensi yang digunakan. Sedangkan untuk nilai Z_{in}' digunakan rumus $Z_{IN}' = \frac{Z_{IN}}{Z_0}$ sehingga rumusnya akan menjadi $Z_{IN}' = \frac{Z_L + j Z_0 \tan(\beta L)}{Z_0 + j Z_L \tan(\beta L)}$. Dari hasil perhitungan tersebut akan didapatkan nilai impedansi input dan plot *smithchart* untuk impedansi input normalisasinya.



Gambar 2. Flowchart perhitungan impedansi input dan plotting smithchart (Zin')

Gambar 3 menampilkan flowchart alur pembuatan program dari kalkulator impedansi input dan plot smithchart untuk Zin' dengan bahasa pemrograman Python. Tahap awal pembuatan adalah membuat variabel untuk setiap nilai yang yang dibutuhkan berdasarkan rumus Zin lossless yaitu impedansi beban real dan imaginer, impedansi karakteristik real dan imaginer, panjang kabel, dan frekuensi, selanjutnya memasukkan rumus perhitungan ke program untuk mendapatkan nilai yang dicari yaitu Zin, dan Zin', kemudian merancang smithchart di Python dengan modul Plotly [5], dan yang terakhir membuat GUI Python dengan modul tkinter [6]. Jika program sudah berjalan dengan baik maka pembuatan program dianggap selesai, sedangkan jika program tidak berhasil atau program tidak berjalan semestinya maka perlu dilakukan evaluasi program.

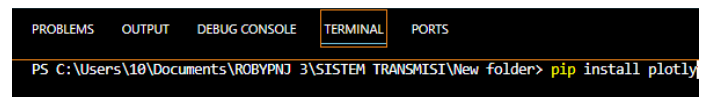


Gambar 3. Flowchart Pembuatan program Python kalkulator Zin dan plot smithchart (Zin')

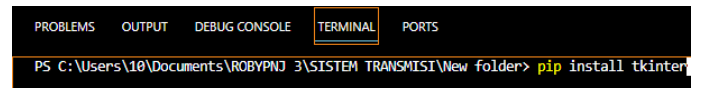
IV. HASIL DAN ANALISA

Program ini memerlukan *software code editor* untuk Python seperti Visual Studio Code, Jupyter Notebook, Sublime Text, Thonny, dan software code editor untuk Python lainnya. Sebelum menjalankan programnya perlu dipastikan lebih dulu bahwa modul-modul yang digunakan yaitu Plotly dan tkinter telah diinstall pada software code editor yang digunakan. Caranya yaitu dengan menggunakan *package manager* Python yaitu PIP kemudian pada terminal code editor ketik :

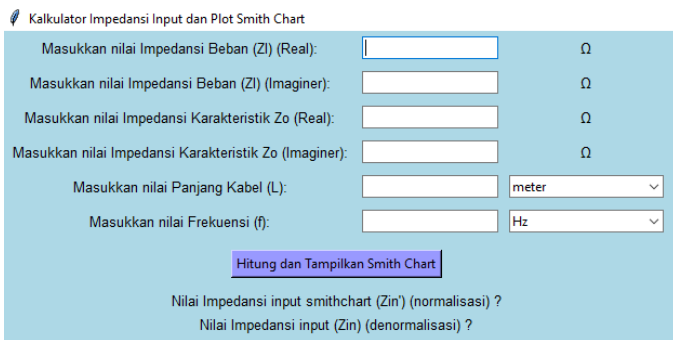
pip install (modul yang ingin diinstal) seperti berikut :



Gambar 4. Cara menginstall modul Plotly

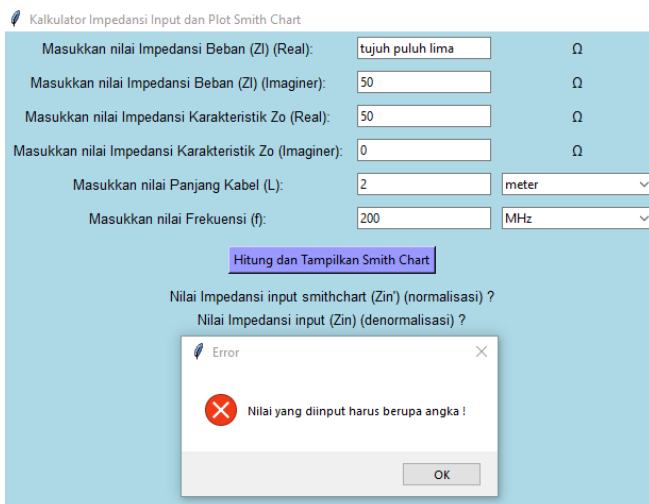


Gambar 5. Cara menginstall modul tkinter

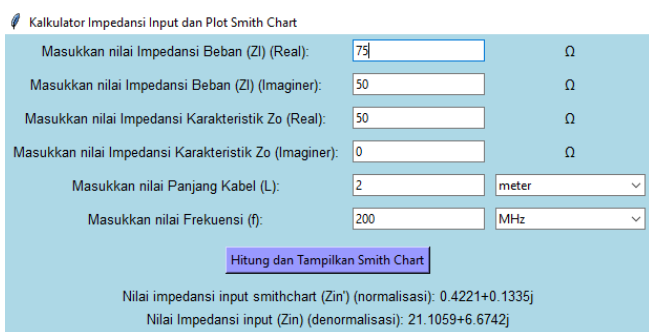


Gambar 6. Tampilan GUI Kalkulator Impedansi Input dan Plot Smithchart

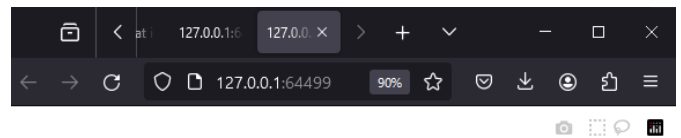
Gambar 6 menampilkan GUI Kalkulator Impedansi Input dan Plot Smithchart untuk Z_{in}' . Nilai-nilai yang diinput diketik pada label harus berupa angka (*float*). Jika *user* menginput selain angka pada nilai-nilai yang dibutuhkan kemudian mengklik tombol “Hitung dan Tampilkan Smithchart”, maka akan muncul pesan *error* yang mengharuskan *user* menginput angka seperti pada Gambar 7. Jika *user* telah menginput setiap nilai dengan angka seperti pada Gambar 8, maka ketika mengklik tombol “Hitung dan Tampilkan Smithchart”, akan muncul hasil impedansi inputnya (Z_{in} dan Z_{in}') serta ditampilkan plot smithchartnya di *default browser* pada perangkat yang digunakan seperti pada Gambar 9.



Gambar 7. Pesan Error yang muncul ketika *user* menginput nilai selain angka

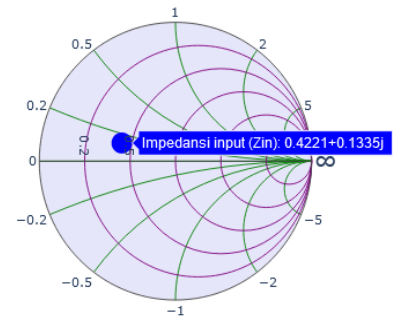


Gambar 8. Nilai Z_{in} dan Z_{in}' muncul ketika *user* menginput nilai dengan benar



Plot Smith Chart Impedansi Input

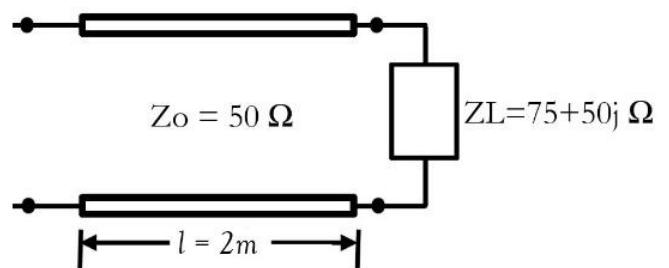
Impedansi Input Smithchart (Z_{in}') (normalisasi) = $0.4221+0.1335j$
 Impedansi Input ($Z_{in} = Z_{in}' \times Z_0$) (denormalisasi) = $21.1059+6.6742j$



Gambar 9. Tampilan plot smithchart di browser Mozilla Firefox

Pada Gambar 9 titik biru merupakan plot smithchart untuk nilai impedansi input normalisasinya (Z_{in}'). *Icon* berbentuk kamera pada Gambar 9 dapat digunakan untuk mendownload hasil plot smithchart yang didapat dengan format png.

$$f = 200 \text{ MHz}$$



Gambar 10. Contoh Impedansi Input

Berikut merupakan penyelesaiannya dengan cara perhitungan manual :

$$\lambda = \frac{3 \times 10^8}{200 \times 10^6} = 1.5 \text{ meter}$$

$$\beta = \frac{2\pi}{1.5} = 4.18879 \approx 4.189$$

$$\beta l = 4.189 \times 2 = 8.378$$

$$\beta l \text{ (derajat)} = 8.378 \times \frac{180^\circ}{\pi} \approx 480.024^\circ$$

$$Z_{IN} = (50) \left(\frac{(75+50j) + j(50)\tan(480.024^\circ)}{Z_0 + j(75+50j)\tan(480.024^\circ)} \right)$$

$$Z_{IN} \approx (21.1 + 6.68j) \Omega$$

$$Z'_{IN} = \frac{(21.1 + 6.68j)}{50} \approx (0.42 + 0.133j) \Omega$$

V. KESIMPULAN

Setelah dilaksanakan project “Penyelesaian Plotting Smithchart untuk Impedansi Input dengan Pemrograman Berbasis GUI Python” didapatkan kesimpulan bahwa program telah berhasil dibuat dan dijalankan dengan baik. Hasil perhitungan impedansi input serta hasil smithchart yang didapat dengan menggunakan program juga sudah sesuai dengan perhitungan matematis. Program ini dapat memudahkan pengguna untuk menghitung impedansi input dan menampilkan smithchart tanpa harus melakukan perhitungan matematis.

DAFTAR PUSTAKA

- [1] R. F. Keefe et al., “Positioning Methods and the Use of Location and Activity Data in Forests,” *Forests*, vol. 10, no. 5, pp. 458–458, May 2019, doi: <https://doi.org/10.3390/f10050458>.
- [2] V. Frendiana, *Penyelesaian Sistem Transmisi Dengan Pemrograman*, Depok: *PNJ Press*, 2018. ISBN: 978-623-7342-98-4
- [3] M. Al-Khalid dan M. Mahdi, “Antennas, 3rd-year, Fifth Lecture,” Al-Mustansiriyah University, Bagdad, Irak, 2018.
- [4] V. Frendiana and M. T. Rahmansyah, “Penyelesaian Perhitungan pada Matching Saluran Transmisi $\lambda/4$ Menggunakan Pemrograman Berbasis Website,” *Spektral*, 2021. DOI:10.32722/spektral.v2i2.4125
- [5] Plotly, “Smith Charts in Python,” plotly | Graphing Libraries, [Online]. Available: <https://plotly.com/python/smith-charts/#reference>. [Accessed 28 January 2024].
- [6] Python, “tkinter — Python interface to Tcl/Tk,” Python.org, [Online]. Available: <https://docs.python.org/3/library/tkinter.html>. [Accessed 28 January 2024].
- [7] P. Smith, “Electronic Applications of the Smith Chart,” *Noble Publishing Corporation*, 2000
- [8] Bill Schweber, “The Smith Chart: A Vital Graphical Tool”, *DigiKey*, 2018.
- [9] D. P. Wati, I. Santoso, and A. A. Zahra, “Simulasi Smith Chart Untuk Penyesuai Impedans Tipe Trafo $1/4 \lambda$ dan Tipe Single Stub - Diponegoro University | Institutional Repository (UNDIP-IR),” *Undip.ac.id*, Jan. 2011, doi: <http://eprints.undip.ac.id/25486/1/ML2F306021.pdf>.
- [10] H. U. O. Tampubolon, “Pembuatan Alat Bantu Ajar Matching Impedance Dengan Menggunakan Smith Chart,” *Repositori*, 2024. <https://repositori.telkomuniversity.ac.id/pustaka/96602/pembuatan-alat-bantu-ajar-matching-impedance-dengan-menggunakan-smith-chart.html>