

Penerapan *Model-Based Design* untuk Kendali Elektronika Daya: Studi Kasus STM32 dan Simulink

^{1*} Asep Andang, ² Ervan Paryono, ³ Rizal Anshori

¹ Jurusan Teknik Elektro, Universitas Siliwangi, Tasikmalaya

² School of Electrical Engineering and Informatics, Institute Technology of Bandung, Bandung

³ Jurusan Teknik Elektro, Universitas Siliwangi, Tasikmalaya

¹ andhangs@unsil.ac.id, ² 23224010@mahasiswa.itb.ac.id, ³ ansorrizal9@gmail.com

Article Info

Keyword:

Model-Based Design
Inverter
Model in Loop
Software in Loop
Hardware in Loop

Copyright © 2025 -SNTE
All rights reserved

ABSTRACT

Penelitian ini membahas penerapan *Model-Based Design* (MBD) untuk sistem kendali Elektronika Daya, dengan fokus pada pengendalian inverter berbasis *Sinusoidal Pulse Width Modulation* (SPWM) menggunakan MATLAB Simulink dan mikrokontroler STM32. Metodologi yang digunakan mengikuti kerangka V-Model, meliputi tahapan perancangan konsep, pemodelan, verifikasi *Software-in-Loop* (SIL), dan validasi *Hardware-in-Loop* (HIL). Studi kasus yang digunakan adalah inverter H-bridge satu fasa dengan kendali SPWM. Hasil penelitian menunjukkan keberhasilan implementasi MBD dalam merancang dan menguji sistem kendali inverter, dengan kesesuaian antara hasil simulasi dan pengujian hardware. Pendekatan ini terbukti efektif dalam mempercepat proses pengembangan, meningkatkan keandalan sistem, dan memfasilitasi iterasi desain yang cepat. Hasil pengujian menunjukkan kesesuaian antara simulasi dan implementasi hardware dengan selisih error RMSE sebesar 7,96 dan perbedaan THD sekitar 5,77% (simulasi 59,66% vs pengukuran 65,43%). Hal ini mengindikasikan model cukup representatif terhadap kondisi nyata. Studi ini diharapkan dapat menjadi referensi aplikatif bagi pengembangan sistem kendali daya yang efisien dan mudah direplikasi, baik dalam lingkungan pendidikan maupun industri.

Corresponding Author:

Asep Andang

Jurusan Teknik Elektro, Universitas Siliwangi, Tasikmalaya

Jl. Siliwangi No. 24

Email: andhangs@unsil.ac.id

I. PENDAHULUAN

Elektronika daya merupakan bidang yang sangat vital dalam sistem kelistrikan modern, termasuk pada sistem energi terbarukan[1], penggerak motor listrik[2], dan konversi daya dalam peralatan industri maupun rumah tangga [3]. Salah satu komponen penting dalam sistem ini adalah inverter, yang berfungsi mengubah energi listrik arus searah (DC) menjadi arus bolak-balik (AC) dengan bentuk gelombang sinusoidal. Untuk menghasilkan gelombang sinusoidal yang berkualitas, diperlukan sistem pengendali yang handal dan akurat, salah satunya melalui teknik *Sinusoidal Pulse Width Modulation* (SPWM)[4].

Pada umumnya, kendali untuk inverter biasanya menggunakan mikrokontroler seperti Arduino[5], STM32[6], [7], atau FPGA[8]. Perancangan pemrograman biasanya dilakukan secara manual menggunakan bahasa pemrograman tingkat rendah seperti C atau C++, yang membutuhkan banyak waktu, ketelitian tinggi,

serta proses debugging yang kompleks[9], [10], [11]. Hal ini menjadi tantangan tersendiri terutama bagi kalangan akademisi atau praktisi yang ingin mempercepat proses pengembangan sistem.

Namun, perancangan manual menggunakan bahasa pemrograman tingkat rendah (misalnya C/C++) memerlukan waktu pengembangan yang panjang, rawan kesalahan sintaks maupun logika, serta sulit dalam proses debugging. Hal ini menjadi *research gap* yang mendorong penelitian ini. Pendekatan Model-Based Design (MBD) tidak hanya mempersingkat waktu pengembangan, tetapi juga memberikan peluang validasi kuantitatif antara hasil simulasi dan implementasi perangkat keras. Aspek ini masih jarang dilaporkan secara detail dalam literatur, khususnya pada kendali inverter berbasis STM32, sehingga memberikan nilai kebaruan penelitian ini

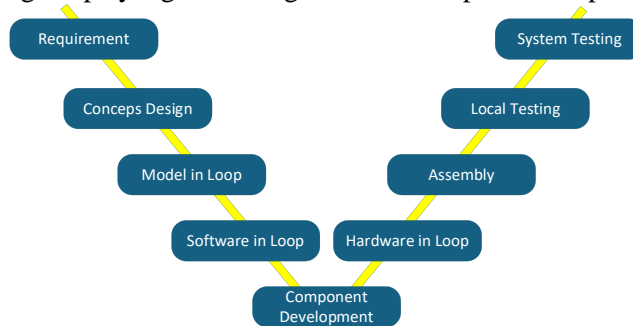
Sebagai solusi, MATLAB Simulink menawarkan pendekatan *Model-Based Design* (MBD) yang memungkinkan pengguna merancang sistem kendali secara visual dalam bentuk blok diagram, kemudian secara otomatis menghasilkan kode tertanam (*embedded code*) untuk diimplementasikan langsung ke perangkat keras, seperti mikrokontroler STM32. Simulink, melalui integrasi dengan *Embedded Coder* dan dukungan STM32-MAT/Target, memfasilitasi proses desain, simulasi, verifikasi, dan *deployment* dalam satu lingkungan terpadu[12].

Pendekatan ini tidak hanya mempercepat proses pengembangan, tetapi juga meningkatkan keandalan sistem karena desain dapat langsung diuji pada tahap simulasi sebelum implementasi. Dengan MBD, pengguna dapat meminimalkan kesalahan sintaks dan logika pemrograman serta mempercepat iterasi desain.

Makalah ini membahas penerapan *Model-Based Design* untuk sistem kendali elektronika daya dengan studi kasus pengendalian inverter berbasis SPWM menggunakan Simulink dan mikrokontroler STM32. Fokus utama meliputi proses pembuatan model, integrasi perangkat lunak dan perangkat keras, deployment kode secara otomatis, serta validasi performa sistem melalui pengamatan bentuk gelombang keluaran inverter. Studi ini diharapkan dapat menjadi referensi aplikatif bagi pengembangan sistem kendali daya yang efisien dan mudah direplikasi, baik dalam lingkungan pendidikan maupun industri.

II. METODE PENELITIAN

Metode *Model-Based Design* membutuhkan proses berjenjang untuk memastikan bahwa model yang akan diimplementasikan sesuai dengan apa yang dirancang. Proses ini dapat dilihat pada gambar 1.



Gambar 1. Vshape Model-Based Design

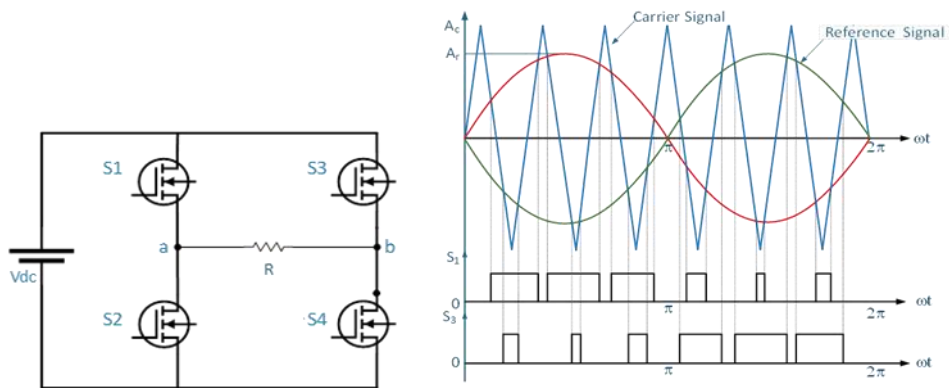
Pendekatan metodologi yang digunakan dalam penelitian ini mengikuti kerangka *Model-Based Design* (MBD) dengan mengacu pada struktur V-Model, seperti ditunjukkan pada Gambar 1. V-Model menggambarkan alur tahapan pengembangan sistem secara sistematis, mulai dari perumusan kebutuhan hingga pengujian sistem secara menyeluruh, dengan pengujian dilakukan secara paralel terhadap setiap tahapan pengembangan. MBD memungkinkan pemodelan sistem kendali secara visual menggunakan blok diagram di Simulink, yang kemudian dapat disimulasikan, diuji, dan diimplementasikan langsung ke perangkat keras, dalam hal ini mikrokontroler STM32.

Tahapan implementasi MBD mengikuti kerangka V-Model. Pada tahap Model-in-Loop (MIL) dilakukan pemodelan inverter satu fasa berbasis SPWM di Simulink. Selanjutnya, tahap Software-in-Loop (SIL) digunakan untuk menguji algoritma SPWM melalui konfigurasi pin PWM pada STM32 menggunakan CubeMX. Kemudian, tahap Hardware-in-Loop (HIL) memvalidasi sinyal kendali secara real-time menggunakan perangkat inverter dan beban aktual. Dengan tahapan berlapis ini, penelitian memastikan kesesuaian fungsi sejak level simulasi hingga level hardware, sekaligus memfasilitasi iterasi desain yang lebih cepat.

Pada sisi kiri V, proses dimulai dari identifikasi kebutuhan sistem (*requirement*), perancangan konsep (*concept design*), dan pemodelan algoritma kendali menggunakan Simulink (*modelling*). Selanjutnya, dilakukan verifikasi awal melalui tahap *Software-in-Loop* (SIL), di mana model diuji secara simulatif tanpa melibatkan perangkat keras. Setelah model tervalidasi secara fungsional, dilakukan pengembangan komponen sistem secara konkrit (*component development*) sebagai basis untuk implementasi fisik.

Sisi kanan V mencakup proses implementasi dan validasi berbasis perangkat keras. Model yang telah dikembangkan di-deploy ke STM32 dan diuji melalui tahap *Hardware-in-Loop* (HIL), memungkinkan evaluasi performa kendali secara *real-time*. Setelah perangkat keras dirakit (*assembly*), dilakukan pengujian lokal (*local testing*) dan diakhiri dengan *system testing* untuk memastikan bahwa sistem berfungsi sesuai dengan spesifikasi awal. Pendekatan ini menjamin konsistensi antara desain simulatif dan implementasi aktual, serta memungkinkan iterasi cepat selama proses pengembangan sistem kendali elektronika daya.

A. Model in Loop



Gambar 2. Model Inverter satu fasa dengan kendali SPWM

Gambar 2 menunjukkan konfigurasi sistem kendali Elektronika Daya yang digunakan dalam penelitian ini, yaitu rangkaian inverter H-bridge satu fasa yang dikendalikan dengan metode SPWM. Inverter ini terdiri dari empat saklar daya (S1–S4) dinyalakan melalui sinyal SPWM, menghasilkan dua sinyal kendali komplementer (S1 dan S2 untuk sisi kiri, S3 dan S4 untuk sisi kanan) yang menentukan arah aliran arus pada beban resistif (R). Tegangan masukan DC (V_{dc}) akan diubah menjadi tegangan bolak-balik melalui pengaturan urutan pensaklaran tersebut, sesuai logika SPWM.

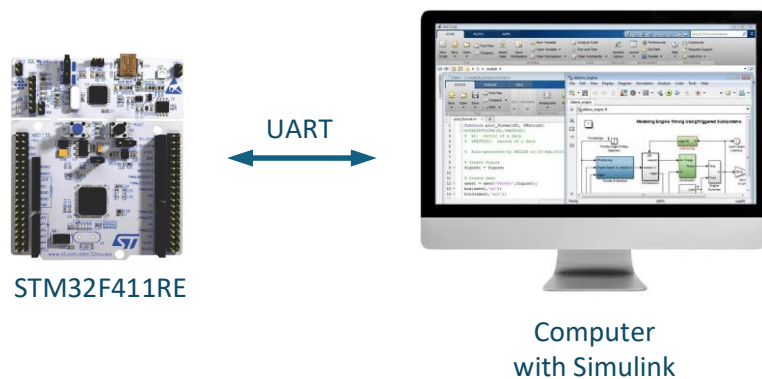
Tegangan inverter yang terbentuk adalah tegangan antara titik a dan titik b dengan besaran:

$$V_o = V_a - V_b \tag{1}$$

Dengan nilai V_o berupa:

$$V_o = \begin{cases} +V_{dc}, & \text{jika } S1 = 1 \text{ dan } S4 = 1 \\ -V_{dc}, & \text{jika } S2 = 1 \text{ dan } S3 = 1 \\ 0, & \text{lainnya} \end{cases} \tag{2}$$

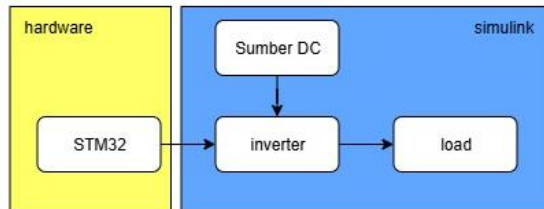
B. Software in Loop



Gambar 3. Konfigurasi SIL

Sistem *Software in loop* pada gambar 3 merujuk pada referensi[13], [14]. Pendekatan SIL telah terbukti efektif dalam mempercepat proses pengujian dan validasi algoritma kendali sebelum

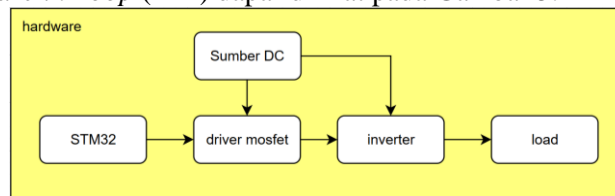
implementasi ke perangkat keras sebenarnya. Vignati et al. (2021) menggunakan SIL untuk menguji algoritma *Vehicle Control Unit* (VCU) pada sistem kereta tanpa pengemudi dengan model kendaraan yang dibuat dalam lingkungan Simulink dan terhubung melalui protokol CAN untuk mereplikasi kondisi nyata dengan resolusi waktu tinggi[13]. Sementara itu, Hisar et al. (2021) memanfaatkan simulasi SIL sebagai bagian dari proses prototyping cepat (*rapid prototyping*) untuk konverter *buck interleaved* berbasis MATLAB/Simulink, di mana kode hasil desain diuji secara lokal dalam lingkungan simulasi sebelum di-deploy ke prosesor melalui pendekatan *Processor-in-Loop* (PIL)[14]. Diagram blok dari SIL ini dapat dilihat pada gambar 4.



Gambar 4. Diagram blok SIL

C. *Hardware in Loop (HIL)*

Setelah diuji dan divalidasi bahwa hasil keluaran dari STM32 dapat bekerja pada inverter dan beban secara simulasi, maka langkah selanjutnya adalah menguji hardware inverter dan beban secara langsung. Blok diagram dari *Hardware in Loop* (HIL) dapat dilihat pada Gambar 5.

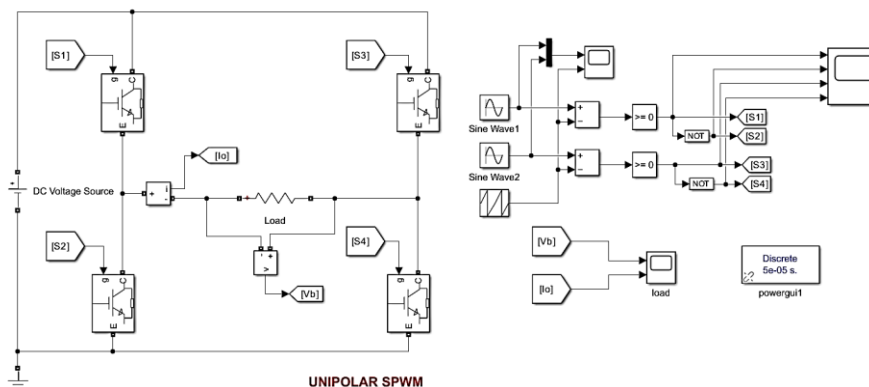


Gambar 5. Blok diagram HIL

III. HASIL DAN DISKUSI

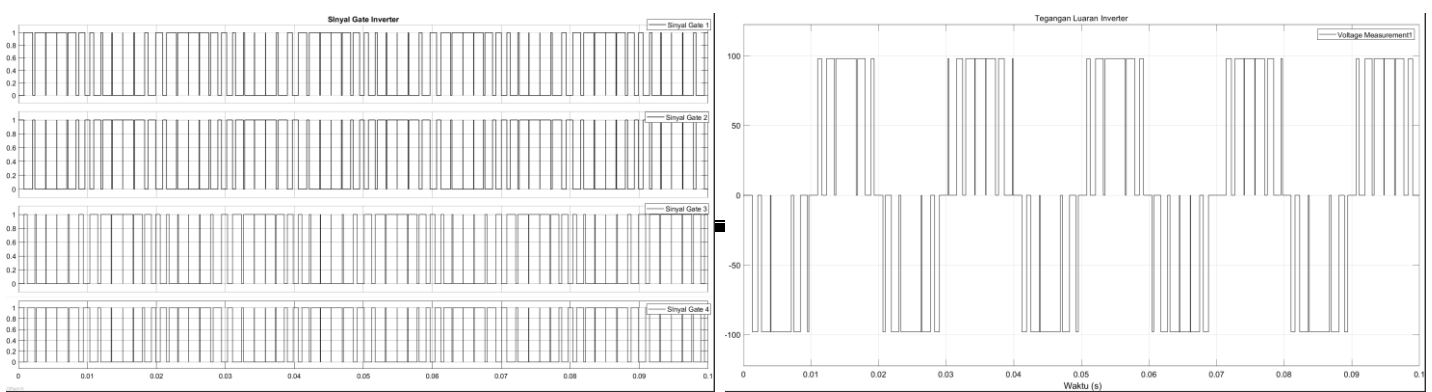
A. *Simulasi Model in Loop*

Model inverter SPMW satu fasa yang diimplementasikan pada simulink menghasilkan rangkaian seperti pada gambar 6.



Gambar 6. Rangkaian inverter SPWM satu fasa yang dimodelkan

Pengujian dalam *Model in Loop* menghasilkan sinyal-sinyal yang ditampilkan dalam gambar 7 yaitu berupa sinyal gate 1 yang terhubung ke MOSFET 1, Sinyal Gate 2 yang terhubung ke MOSFET 2, Sinyal



Gate 3 yang terhubung ke MOSFET 3, Sinyal Gate 4 yang terhubung ke MOSFET 4 dan tegangan luaran dari inverter

Gambar 7. Gelombang hasil dari model in loop

B. Hasil Software in Loop

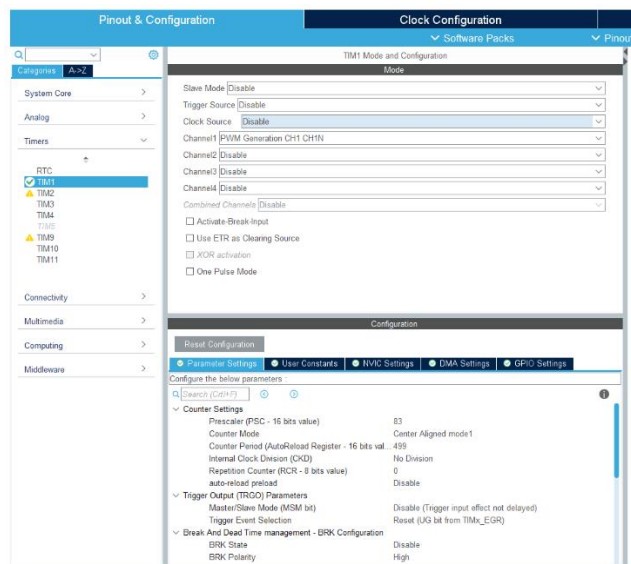
Software in Loop pada STM32 dilakukan dengan: Tahap pertama dilakukan setting konfigurasi pin STM32 menggunakan CubeMX. CubeMX menyeting TIM1 → Channel1 menjadi PWM Generation CH1 CH1N sehingga pin CH1(PA7) dapat digunakan sebagai output PWM serta CH1N (PA8) menjadi complement output dari CH1(PA7). Untuk mendapatkan frekuensi SPWM dilakukan dengan persamaan:

$$f_{spwm} = \frac{f_{clock_system}}{(Prescaler+1)} \times (counter\ period + 1) \quad (3)$$

Berdasarkan persamaan (3) untuk mendapatkan frekuensi SPWM menjadi 500 Hz maka:

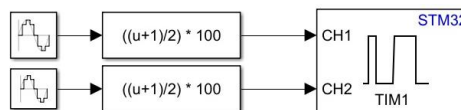
$$f_{spwm} = \frac{84.000.000}{(83 + 1)} \times (499 + 1)$$

Prescaler di setting menjadi 83 dan counter period menjadi 499. Setting CubeMX dapat dilihat pada gambar 8.



Gambar 8. Konfigurasi CubeMX

Tahap kedua dilakukan dengan membuat sinyal yang diinginkan pada pin CH1 menggunakan Model-Based Design (MBD) pada simulink. Sistem MBD agar menghasilkan SPWM dapat dilihat pada gambar 9.



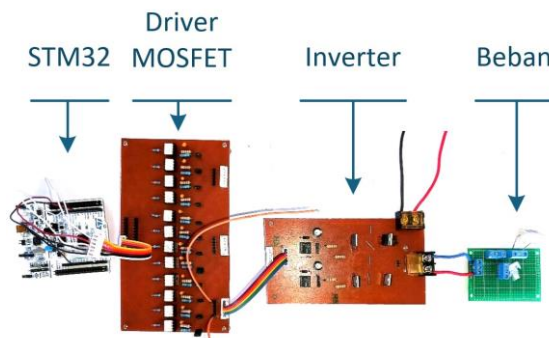
Gambar 9. MBD SPWM untuk STM32

Hasil pembacaan sinyal gelombang pensaklaran dari STM32 dapat dilihat pada gambar 10. Diperlihatkan bahwa sinyal gating untuk 4 saklar sama dengan simulasi. Tahap ketiga, MBD yang telah dibuat dapat di-deploy langsung ke SMT32. STM32 akan membuat sinyal SPWM dengan frekuensi 1000 Hz, maka setiap 1 ms STM32 akan mengirim data pensaklaran menuju simulink (komputer). Data pensaklaran ini akan digunakan untuk mengendalikan inverter.

HIL ini terdiri dari berbagai perangkat hardware. Pada HIL inverter diuji secara langsung dalam bentuk hardware. Oleh karena itu inverter memerlukan perangkat driver. Untuk lebih jelasnya dapat dilihat pada Gambar 11.



Gambar 10. Hasil pengujian *Software in Loop* pada STM32



Gambar 11. Sistem rangkaian HIL

Berdasarkan hasil simulasi, dihasilkan output pensaklaran seperti yang ditunjukkan pada gambar 7. Hasil simulasi pensaklaran ini dibandingkan dengan hasil HIL yang diperlihatkan pada gambar 10. Dapat dilihat bahwa hasil simulasi MIL pada gambar 7 dan hasil pengujian HIL secara langsung menggunakan osiloskop pada gambar 12 menghasilkan output yang sama.

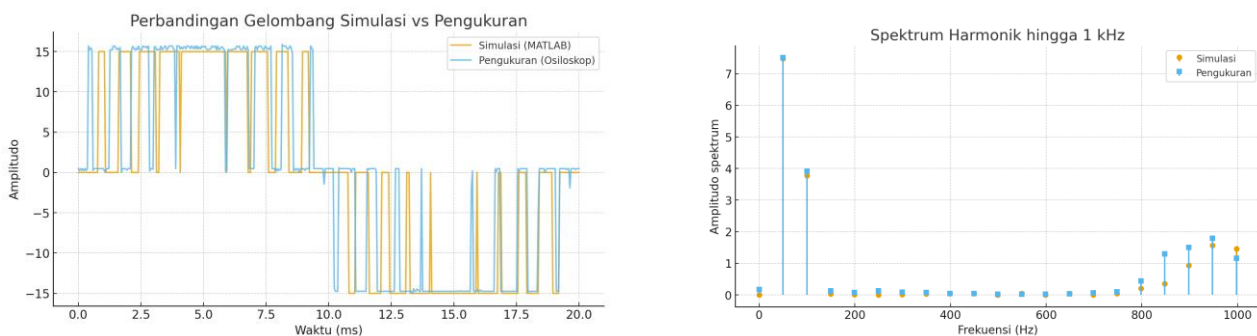
MSO5104 Fri August 08 11:36:26 2025



Gambar 12. Gelombang tegangan keluaran rangkaian HIL

Hasil analisa dari simulasi dan pengukuran dapat dilihat pada gambar 13(a) untuk bentuk gelombang satu periode sedangkan gambar 13(b) merupakan spektrum harmonisa dimana nilai tegangan THD simulasi $V_{sTHD} = 59,66\%$ sedangkan hasil pengukuran tegangan THD pengukuran $V_{pTHD} = 65,43\%$ terdapat selisih $5,77\%$ menunjukkan adanya tambahan harmonisa pada implementasi aktual, terutama di sekitar orde ke-5, 9, dan 19, yang disebabkan oleh rugi-rugi switching, toleransi komponen, dan noise lingkungan. Sedangkan panalisa RMSE dengan membandingkan secara langsung antara sinyal simulasi dan pengukuran

memberikan nilai RMSE = 7,96, yang menunjukkan deviasi rata-rata relatif kecil terhadap amplitudo gelombang.



Gambar 13. (a) Perbandingan gelombang simulasi dan pengukuran (b) perbandingan THD simulasi dan pengukuran

IV. PENUTUP

Pada Artikel ini telah dibahas proses perancangan inverter dengan berbasis pada metode Model Based Design dengan langkah per langkah serta perbandingan hasil simulasi dengan pengujian hardware. Analisis kuantitatif melalui THD dan RMSE membuktikan bahwa pendekatan MBD tidak hanya mempersingkat proses pengembangan, tetapi juga menghasilkan performa inverter yang mendekati kondisi aktual dengan deviasi error di bawah 10%. Pengembangan ke depannya proses MBD ini dapat dikembangkan pada sistem yang lebih kompleks pada topologi multilevel inverter, integrasi dengan filter LCL untuk aplikasi grid-tied, serta evaluasi parameter kualitas daya lain seperti faktor daya dan efisiensi.

UCAPAN TERIMA KASIH

Ucapan terima kasih kepada Jurusan Teknik Elektro dan Fakultas Teknik Universitas Siliwangi yang telah memberikan fasilitas pendanaan untuk mengikuti Seminar nasional Teknik Elektro.

REFERENSI

- [1] B. K. Bose, "Power Electronics, Smart Grid, and Renewable Energy Systems," *Proceedings of the IEEE*, vol. 105, no. 11, pp. 2011–2018, Nov. 2017, doi: 10.1109/JPROC.2017.2745621.
- [2] F. A. Silva and M. P. Kazmierkowski, "Power Electronics and Motor Drives: Advances and Trends, Second Edition [Book News]," *IEEE Industrial Electronics Magazine*, vol. 15, no. 2, pp. 89–90, Jun. 2021, doi: 10.1109/MIE.2021.3071209.
- [3] J. Falck, C. Felgемacher, A. Rojko, M. Liserre, and P. Zacharias, "Reliability of Power Electronic Systems: An Industry Perspective," *IEEE Industrial Electronics Magazine*, vol. 12, no. 2, pp. 24–35, Jun. 2018, doi: 10.1109/MIE.2018.2825481.
- [4] A. H. Al-Omari and S. Al-Zubaidi, "Investigating low order harmonics of sinusoidal pulse width modulation with voltage closed loop control," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 26, no. 3, p. 1258, Jun. 2022, doi: 10.11591/ijeecs.v26.i3.pp1258-1265.
- [5] R. S. Lubis, A. Haris, and T. Tarmizi, "UPS Design for Increased Flexibility of Use and More Economic with PWM Controlled Inverter Based on ATmega 328 Microcontroller," *TEKNIK*, vol. 43, no. 1, pp. 102–111, May 2022, doi: 10.14710/teknik.v43i1.32736.
- [6] S. YARIKKAYA and K. VARDAR, "Stm32f4 Ve Matlab Kullanılarak Tek Faz Şebeke Bağlantılı Pv İnverterin Hızlı Prototip Geliştirilmesi," *European Journal of Science and Technology*, pp. 213–223, Apr. 2020, doi: 10.31590/ejosat.680586.
- [7] M. R. Rusli *et al.*, "BLDC Motor Drives with A Programmable Simplified C-Block to Generate Accurate Six-Step PWM Based on STM32 Microcontroller," *Elinvo (Electronics, Informatics, and Vocational Education)*, vol. 7, no. 2, pp. 112–118, Jan. 2023, doi: 10.21831/elinvo.v7i2.52992.
- [8] D. Sankar, L. Syamala, B. Chembathu Ayyappan, and M. Kallarackal, "FPGA-Based Cost-Effective and Resource Optimized Solution of Predictive Direct Current Control for Power Converters," *Energies (Basel)*, vol. 14, no. 22, p. 7669, Nov. 2021, doi: 10.3390/en14227669.

-
- [9] W.-J. Li, D.-S. Li, and J.-W. Zhang, “Model-Based Design and Experimental Validation of Control System for a Three-Level Inverter,” *Electronics (Basel)*, vol. 11, no. 13, p. 1979, Jun. 2022, doi: 10.3390/electronics11131979.
- [10] A. Bucaioni, F. Ciccozzi, A. Di Salle, and M. Sjödin, “From low-level programming to full-fledged industrial model-based development: the story of the Rubus Component Model,” *Softw Syst Model*, vol. 22, no. 4, pp. 1085–1097, Aug. 2023, doi: 10.1007/s10270-023-01107-3.
- [11] R. Wang *et al.*, “A Formal Model-Based Design Method for Robotic Systems,” *IEEE Syst J*, vol. 13, no. 1, pp. 1096–1107, Mar. 2019, doi: 10.1109/JSYST.2018.2867285.
- [12] T. Jongsataporn and S. Chivapreecha, “Embedded hardware implementation of noise gate on STM32 processor using MATLAB/Simulink,” in *2024 International Conference on Photonics Solutions (ICPS2024)*, A. Bhatranand, Ed., SPIE, Feb. 2025, p. 18. doi: 10.1117/12.3058631.
- [13] M. Vignati, N. Debattisti, M. L. Bacci, and D. Tarsitano, “A software-in-the-loop simulation of vehicle control unit algorithms for a driverless railway vehicle,” *Applied Sciences (Switzerland)*, vol. 11, no. 15, Aug. 2021, doi: 10.3390/app11156730.
- [14] C. Hisar, I. Sefa, and N. Altin, “Processor-in-the-Loop Simulation of an Interleaved Buck Converter With MATLAB/Simulink,” in *2021 13th International Conference on Electronics, Computers and Artificial Intelligence (ECAI)*, IEEE, Jul. 2021, pp. 1–6. doi: 10.1109/ECAI52376.2021.9515058.