European Journal of Technique

journal homepage: https://dergipark.org.tr/en/pub/ejt

Vol.15, No.2, 2025

Research Article

# GUI Design with C# Windows Form Application for MPP Estimation

**Fatih Gundogdu**[1*], **Ahmet Gundogdu**[2], **Resat Celikel**[3]

[1] Karamanoglu Mehmetbey University, Computer Engineering Department, Karaman, Turkiye. (fagundogdu23@gmail.com)
[2*]Batman University, Electrical and Electronics Engineering Department, Batman, Turkiye. (agundogdu23@gmail.com).
[3] Firat University, Technology Faculty, Mechatronic Engineering Department, Elazig, Turkiye. (resat.celikel@gmail.com).

## ARTICLE INFO

Received: Oct., 01. 2025
Revised: Oct., 27. 2025
Accepted: Nov., 04. 2025

Keywords:
C#
Windows Form
GUI Design
Photovoltaic
MPPT

Corresponding author: *Ahmet Gundogdu*

ISSN: *2536-5010* | e-ISSN: *2536-5134*

DOI: https://doi.org/10.36222/ejt.1794810

## ABSTRACT

In this study, an interactive graphical user interface was developed to both facilitate the performance analysis of photovoltaic systems and serve educational purposes. The aim was to visualize how the maximum power point of a PV panel changes based on user-defined parameters. The interface was designed using the C# programming language and Windows Forms application. Users can enter key parameters such as ambient temperature and irradiance value via the keyboard. In the background, a mathematical model of the PV panel is executed to calculate the maximum power points for different temperature and irradiance conditions, and the results are displayed on the Graphical User Interface screen. The simulations conducted have shown that the user-friendly interface significantly simplifies the understanding of complex PV panel behaviours. The effects of varying atmospheric conditions on the location of the maximum power point and the overall efficiency of the panel can be clearly observed. The developed C# -based interface is considered an effective, low-cost, and accessible tool for teaching the fundamental principles of photovoltaic systems and enabling quick performance analysis.

## 1. INTRODUCTION

With the increasing global population, the growing demand for energy and the negative environmental impacts of fossil fuels have been driving greater interest in renewable energy sources (RES) [1]. Among these sources, solar energy holds great significance due to its high potential. Photovoltaic (PV) technologies, developed to generate electrical energy from solar power, are among the most promising investment areas for the future. The efficiency of PV panels is largely influenced by environmental factors such as the solar irradiance incident on the panels and their operating temperature. Under these variable atmospheric conditions, it is essential to continuously monitor the operating point and maintain it at the maximum power point (MPP) in order to extract the maximum power from the panel [2]. maximum power point tracking (MPPT) algorithms, developed to track these points, play a critical role in PV systems. Some studies in the literature related to MPPT are summarized below.

MPP estimation was performed by comparing different regression techniques on a large dataset. The results showed that simple models could be sufficient for embedded applications [3]. By utilizing the relationship between the duty cycle of the DC-DC converter and the maximum power point, fast estimation was achieved without the need for complex calculations. The algorithm's robustness was enhanced against atmospheric changes such as sudden shading [4]. Additionally, irradiance and temperature were estimated using only the data around the maximum power point, and the model was updated accordingly. With this updated model, estimation errors under varying environmental conditions were reduced [5]. PV model was created using equivalent circuit parameters extracted from catalog data. This model enables the estimation of the MPP under varying atmospheric conditions and serves as a basis for simulations prior to field installations [6]. Field measurements were conducted to optimize the parameters of the PV system. This approach provides more reliable estimations in practice, as it reflects real environmental effects rather than relying solely on catalog values [7]. A low-cost method utilizing decision trees and their derivatives (such as Random Forest) was developed for MPP prediction and parameter estimation [8]. In a PV-TEG hybrid system, deep learning models were employed to estimate the MPP [9]. ANN-based MPPT algorithm was developed for standalone photovoltaic PMSM drive system without dc-dc boost converter [21].

Advanced and high-cost software packages such as Matlab/Simulink are commonly used to test MPPT algorithms and model PV panel behavior. However, access to such software is not always easy, especially for undergraduate students and researchers aiming for rapid prototyping. This study aims to alleviate or eliminate this accessibility barrier.

In this study, a graphical user interface (GUI) was developed using C#, a widely used and accessible programming language. This interface allows users to enter the basic parameters of a PV panel via the keyboard and calculate the panel’s maximum power point. It also provides the ability to analyze how these points change depending on variations in irradiance and temperature. The remainder of the paper is organized as follows: Section 2 explains the mathematical model of the PV cell. Section 3 is dedicated to the design and coding of the developed interface. Section 4 presents the numerical results and interface visuals obtained through the implemented applications. Finally, Section 5 summarizes the study’s conclusions and discusses potential improvements.

## 2. MATHEMATICAL MODEL of PV CELL

PV cells convert the radiation in sunlight directly to electric energy. It was developed by Carlson and Wronski in 1976 [10]. To analyze the performance of PV systems, optimize their design, and predict their behavior under varying atmospheric conditions, a mathematical model of PV cells is required. The most common and practical model is the single-diode equivalent circuit model. This model, based on the physical operating principle of the PV cell, describes its electrical characteristics with high accuracy. The single-diode equivalent circuit model is presented in Fig.1 [2].

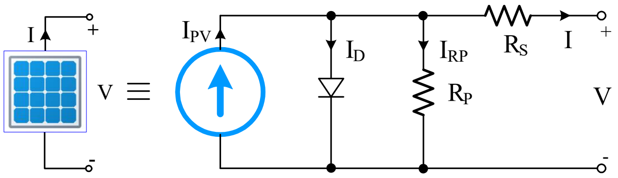


**Figure 1.** Single diode model of a PV cell

The model basically consists of the following components: $I_{PV}$ is referred to as the light-dependent current source. It represents the photovoltaic current generated by the photovoltaic effect. It varies approximately linearly, being directly proportional to the irradiance (G) and inversely proportional to the temperature (T). *D* is used to model the behavior of the semiconductor PN junction. The diode current is expressed by the Shockley diode equation. $R_S$, known as the series resistance, models the losses caused by the ohmic resistances of the semiconductor material, metal contacts, and terminal connections. It has a significant impact on power loss, especially at high panel output currents. $R_P$ is the shunt resistance used to represent the leakage current caused by crystal defects or impurities in the PN junction. It usually has a very high value. Based on these components, the relationship between the output current (*I*) and the terminal voltage (*V*) of the PV cell is expressed by Eq.(1).

$$I = I_{pv} - I_0\left[exp\left(\frac{q(V+IR_S)}{nkT}\right)-1\right]-\left[\frac{V+IR_S}{R_P}\right] \qquad (1)$$

The parameters in the equation are explained below.

*I* : Cell output current (A)
*V* : Cell output voltage (V)
*Ipv* : Photovoltaic current (A)
*Io* : Diode saturation current (A)
*q* : Electron charge ($1.602 \times 10^{-19}$ C)
*k* : Boltzmann constant ($1.38065 \times 10^{-23}$ J/K)
*T* : Absolute temperature (K)
*n* : Diode ideality factor

The practical use of the model depends on the accurate determination of five key parameters ($I_{PV}$, $I_0$, n, $R_S$, $R_P$). These parameters are calculated using data provided by the manufacturer such as the open-circuit voltage ($V_{OC}$), short-circuit current ($I_{SC}$), maximum power point and temperature coefficients measured under standard test conditions (STC)- through iterative methods such as Newton–Raphson and Levenberg–Marquardt. There is no simple algebraic solution for these calculations. In this study, the calculation of these parameters is not addressed. Throughout this study, the focus has been on a generalized equation that simply calculates the maximum power points of a PV system and on the design of a user interface that performs these calculations in the background using this equation. This equation is called the 'Power Geometry-based Linear Estimation Model (PG-LEM)' because it is derived by calculating the slope of the linear line formed by the MP points of the PV system.

The PG-LEM model presented in Eq.(2) was derived using both numerical data obtained from simulations performed in the Matlab/Simulink environment and the power geometry model of the PV system. The derivation details of the model are considered a separate research topic and will not be provided here. However, the accuracy of the model will be demonstrated by comparing it with the graphs obtained from the actual PV models in Matlab/Simulink.

$$P_{MP} = \frac{G}{G_{max}} \cdot \left[P_{max} - \frac{T}{T_{max}} \cdot (P_{max} - P_{min})\right] \qquad (2)$$

Eq.(2) can be used to predict the maximum power points of all small- and large-scale PV systems. As an example application in Fig.2, a PV system producing 1.24 kW under STC was designed in the Matlab/Simulink environment. PV modul parameters are given in Table 1.

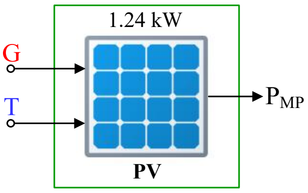


**Figure 2.** 1.24 kW PV Power System under STC

**TABLE I**
PARAMETERS OF PV MODULE

| Symbol | Value | |
|---|---|---|
| $P_{mp}$ | 155 | W |
| $V_{MP}$ | 66.52 | V |
| $I_{MP}$ | 2.33 | A |
| $V_{OC}$ | 82.85 | V |
| $I_{SC}$ | 2.52 | A |
| $T_C$ @ $V_{OC}$ | -0.356 | %/$^0$C |
| $T_C$ @ $I_{SC}$ | 0,07 | %/$^0$C |
| $R_S$ | 2.686 | Ω |
| $R_P$ | 1085 | Ω |
| $N_{Cell}$ | 133 | |

The operating range was defined as G=[0-1000] W/m$^2$ and T=[0-100] $^o$C. In the simulations, the maximum power was measured as $P_{max} = 1367\ W$ for G=1000 W/m$^2$ and T=0 $^o$C. Similarly, the minimum power was measured as $P_{min} = 834\ W$ for G=1000 W/m$^2$ and T=100 $^o$C. In all comparative simulations conducted for different G and T values within the operating range, it was observed that the mathematical model in Eq.(2) produced approximately accurate results. In this respect, Eq.(2) can be readily applied within any predefined operating ranges $[T_{min} < T < T_{max}]$ and $[G_{min} < G < G_{max}]$. However, if this model is to be used for PV systems with different power ratings, the values of $P_{min}$ and $P_{max}$ must be redefined for the corresponding operating conditions, and $P_{mp}$ points should be recalculated accordingly. The 1.24 kW PV power system consists of eight series-connected panels of the "TSMC Solar TS-155C2" model. The $P_{min}$ and $P_{max}$ values of the PV power system correspond to the peak points of the blue *P-V* curves shown in Fig.3. These graphs were obtained from simulations carried out in the Matlab/Simulink environment. Iterative solution methods such as Newton–Raphson and Levenberg–Marquardt were used in generating the *P-V* curves.

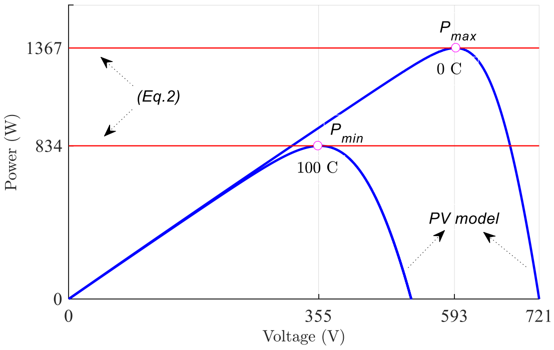


**Figure 3.** P-V curves of PV system and its MP points (TSMC Solar TS-155C2)

The $P_{min}$ and $P_{max}$ values of the power system correspond to the peak points of the blue *P-V* curves shown in Fig.3. MPPT algorithms generally perform intensive calculations using these *P-V* curves. However, Eq.(2) does not require these curves; thanks to its simple structure, it can calculate the maximum power point more quickly. For the same irradiance and temperature values, the maximum power points calculated using Eq.(2) are shown in red in Fig.3 together with the *P-V* curves. It can be observed that the discrete points calculated by Eq.(2) are tangent to the peak points of the *P-V* curves. This demonstrates the accuracy of the calculations performed by Eq.(2).

## 3. DEVELOPMENT OF GUI

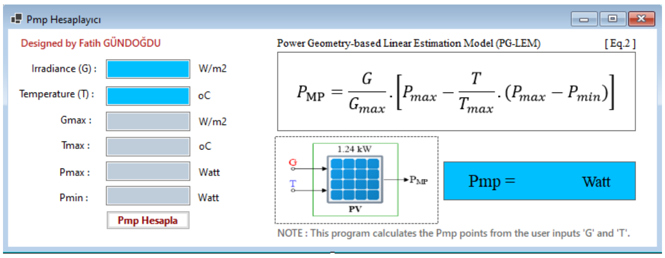
The C# programming language provides a comprehensive infrastructure for interface design within the .NET ecosystem. Interface design with C# is considered an interdisciplinary research area, addressed in conjunction with concepts such as human–computer interaction and software engineering. Some studies in the literature that have utilized C# are summarized below.

A Windows Forms interface based on C# was developed and integrated with a Raspberry Pi to control a smart waste bin. The system detects the waste level, provides visual notifications to the user, and offers remote access over the network [11]. A visual training software for artificial neural networks was introduced, developed on the C#.NET platform. Thanks to the user-friendly interface, algorithm parameters can be easily entered and the training process can be visualized [12]. Using a Kinect sensor and a C# WPF interface, an assistant system was developed that provides real-time monitoring and visual feedback [13]. For the inverse kinematic analysis of a 5-degree-of-freedom gantry-type robot, a C#-based graphical user interface was designed. The interface visually presents the calculations and integrates with the Mach3 software [14]. A cross-platform interface using a C# backend and an Avalonia UI frontend was developed to monitor and control RF telemetry data of unmanned ground vehicles [15].

C#-based graphical user interface software was developed to facilitate bolt placement in engineering designs. Thanks to user-friendly screens, design outputs can be obtained with minimal input [16]. A web-based software suite was created using ASP.NET MVC and C# for the post-processing stages of image processing algorithms, providing a mobile-compatible interface with Bootstrap support [17]. In this study, a C# WinForms–based intelligent teaching system was presented to teach the concepts of algorithms and flowcharts. Users can create flow diagrams via drag-and-drop and experience the execution processes [18]. A system for analyzing facial expressions was developed, and a user interface was implemented in Visual C#. Using .NET 4.5, the interface enables fast data processing and visualization of results [19]. A C# Windows Forms–based interface was developed to monitor real-time data wirelessly received from mobile robots used in agriculture [20]. An interface has been created in the MATLAB Graphical User Interface (GUI) environment using the Newton-Raphson method to solve the current equation of the solar cell [22]. MATLAB/GUI-based training set for solar energy conversion systems was designed [23].

Thanks to the modern approaches it introduces, C# provides developers with the ability to create scalable, maintainable, and user-friendly interfaces. One of these is the Windows Forms application. Its areas of use include customer data entry forms, registration screens, survey applications, library systems, calendars, agendas, task planners, and similar tools. The Windows Forms application is designed for rapid application development and uses an event-driven programming model.

In this section, a practical user interface was designed to calculate the maximum power points of a PV system using irradiance and temperature values entered via the keyboard. The interface was developed with Microsoft Visual Studio 2022 using the C# Windows Forms application. Windows Forms provides rapid prototyping capabilities through its drag-and-drop design convenience. Taking advantage of this feature, the form layout of the interface developed in the C# environment was created as shown in Fig.4.



Designed by Fatih GÜNDOĞDU

Power Geometry-based Linear Estimation Model (PG-LEM) [ Eq.2 ]

$$P_{MP} = \frac{G}{G_{max}} \cdot \left[ P_{max} - \frac{T}{T_{max}} \cdot (P_{max} - P_{min}) \right]$$

NOTE : This program calculates the Pmp points from the user inputs 'G' and 'T'.

**Figure 4.** Interface view (form field) during the design phase.

The code lines related to the preliminary design of the form area are provided below. Eq.(2), which calculates the MPP, is implemented on line 21 of the program. A button was used to trigger the calculation. When clicked, it executes the program lines to calculate the maximum power point and displays the result on the interface screen. For this purpose, a label component was used as the result indicator. To enhance the visual appeal of the interface, a PictureBox was included. Through the PictureBox, the user can visually interact with Eq.(2), which runs in the background to calculate the MPP.

The values entered from the keyboard are initially integers. To enable decimal calculations, the input parameters were defined as float. The input parameters were determined as $G, G_{max}, T, T_{max}, P_{min}, P_{max}$ . Input fields for these parameters were designed using TextBox components. The data entered through the TextBoxes are received as strings and then converted to 32-bit integers using Convert.ToInt32.

```
1    namespace deneme3
2    {
3    public partial class Form1 : Form
4    {
5    public Form1()
6    {
7    InitializeComponent();
8    }
9    private void Form1_Load(object sender, EventArgs e)
10   {
11   }
12   private void button1_Click(object sender, EventArgs e)
13   {
14   float G,T,Gmax,Tmax,Pmin,Pmax,Pmp,sonuc=0;
15   G=Convert.ToInt32(txtG.Text);
16   T=Convert.ToInt32(txtT.Text);
17   Gmax=Convert.ToInt32(txtGmax.Text);
18   Tmax=Convert.ToInt32(txtTmax.Text);
19   Pmin=Convert.ToInt32(txtPmin.Text);
20   Pmax=Convert.ToInt32(txtPmax.Text);
21   Pmp=(G/Gmax)*(Pmax-(T/Tmax)*(Pmax-Pmin));
22   lblSonuc.Text="Pmp=  " + Pmp;
23   }
24   private void pictureBox3_Click(object sender, EventArgs
e)
25   {
26   }
27   private void label15_Click(object sender, EventArgs e)
28   {
29   }
30   private void label17_Click(object sender, EventArgs e)
31   {
32   }
33   }
34   }
```

## 4. RESULTS AND DISCUSSIONS

The developed interface software aims to make the theoretical structure of complex MPPT algorithms more perceptible and accessible. For this purpose, temperature and irradiance values were entered through the parameter input fields on the designed interface, and the $P_{mp}$ points of the PV system under varying atmospheric conditions were calculated. Interface screenshots related to the calculations are provided below. The accuracy of the calculations was verified by comparing them with the *P–V* curves obtained from a PV power system modeled in Matlab/Simulink using TSMC Solar TS-155C2 panels. The maximum power points obtained from the *P–V* curves are presented in Table 2 for comparison. The MP points obtained from the both P-V curves and PG-LEM model under different temperature and irradiance values, along with the model estimation errors (in watts and percentages), are presented in Table 2 for comparison. The percentage errors represent the difference between the results from the PV system and the estimations of the PG-LEM model. The small percentage errors confirm that the model performs estimations with high.

**TABLE II**

$P_{mp}$ POINTS OF PG-LEM MODEL AND 1.24 KW PV SYSTEM

| T ($^0$C) | G (W/m$^2$) | P$_{MP}$ (W) (PV System) | P$_{MP}$ (W) (PG-LEM) | Error (W) | Error (%) |
|---|---|---|---|---|---|
| 0 | 500 | 692.6 | 683.5 | -9.1 | -1.31 |
| | 1000 | 1367 | 1367 | 0 | 0 |
| 25 | 500 | 627.7 | 616.9 | -10.8 | -1.72 |
| | 1000 | 1240 | 1234 | -6 | -0.48 |
| 100 | 500 | 419.4 | 417 | -2.3 | -0.54 |
| | 1000 | 834.1 | 834 | 0.1 | 0.011 |

Application-1:

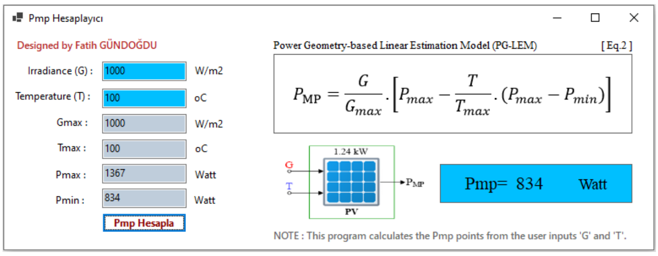
At a constant temperature of T = 100 ºC, the $P_{mp}$ values were calculated as 834 W and 417 W for G=1000 W/m$^2$ and G=500 W/m$^2$, respectively. The corresponding interface screenshots are shown in Fig.5-a and Fig.5-b. These values are identical to the $P_{mp}$ points of the PV system listed in Table 2.

Application-2:

At a constant temperature of T = 25 ºC, the $P_{mp}$ values were calculated as 1233.75 W and 616.87 W for G=1000 W/m$^2$ and G=500 W/m$^2$, respectively. The corresponding interface screenshots are shown in Fig.6-a and Fig.6-b. These values are identical to the $P_{mp}$ points of the PV system listed in Table 2.

Application-3:

At a constant temperature of T = 0 ºC, the $P_{mp}$ values were calculated as 1367 W and 683.5 W for G=1000 W/m$^2$ and G=500 W/m$^2$, respectively. The corresponding interface screenshots are shown in Fig.7-a and Fig.7-b. These values are identical to the $P_{mp}$ points of the PV system listed in Table 2.



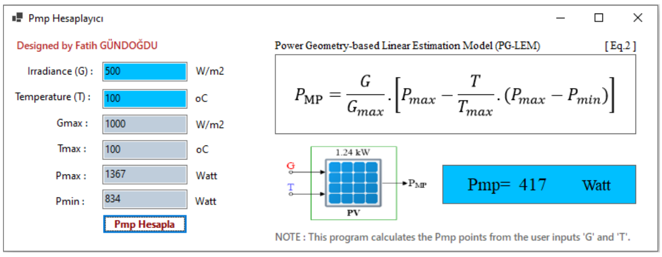
Irradiance (G) : 1000 W/m2
Temperature (T) : 100 oC
Gmax : 1000 W/m2
Tmax : 100 oC
Pmax : 1367 Watt
Pmin : 834 Watt

$$P_{MP} = \frac{G}{G_{max}} . \left[P_{max} - \frac{T}{T_{max}} . (P_{max} - P_{min})\right]$$

Pmp= 834 Watt

NOTE : This program calculates the Pmp points from the user inputs 'G' and 'T'.

**Figure. 5-a.** Application screen for T=100 ºC ve G=1000 W/m$^2$



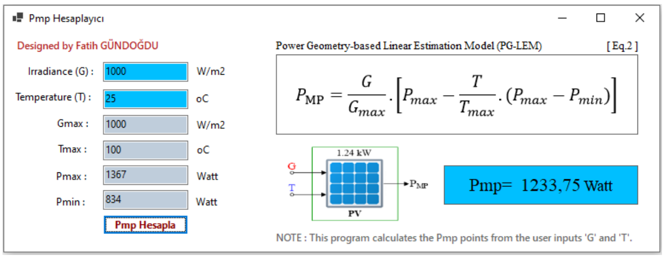
Irradiance (G) : 500 W/m2
Temperature (T) : 100 oC
Gmax : 1000 W/m2
Tmax : 100 oC
Pmax : 1367 Watt
Pmin : 834 Watt

$$P_{MP} = \frac{G}{G_{max}} . \left[P_{max} - \frac{T}{T_{max}} . (P_{max} - P_{min})\right]$$

Pmp= 417 Watt

NOTE : This program calculates the Pmp points from the user inputs 'G' and 'T'.

**Figure. 5-b.** Application screen for T=100 ºC ve G=500 W/m$^2$


Pmp Hesaplayıcı

Designed by Fatih GÜNDOĞDU

Irradiance (G) : 1000 W/m2
Temperature (T) : 25 oC
Gmax : 1000 W/m2
Tmax : 100 oC
Pmax : 1367 Watt
Pmin : 834 Watt

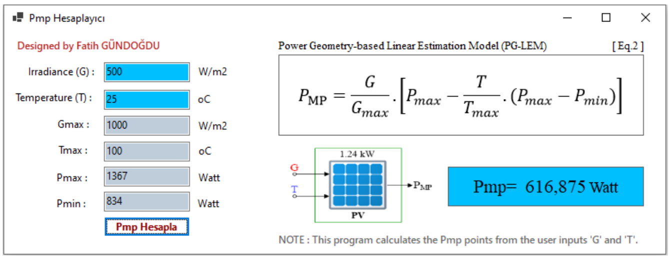
Power Geometry-based Linear Estimation Model (PG-LEM) [ Eq.2 ]

$$P_{MP} = \frac{G}{G_{max}} \cdot \left[ P_{max} - \frac{T}{T_{max}} \cdot (P_{max} - P_{min}) \right]$$

Pmp= 1233,75 Watt

NOTE : This program calculates the Pmp points from the user inputs 'G' and 'T'.

**Figure 6-a.** Application screen for T=25 $^{o}$C ve G=1000 W/m$^2$


Pmp Hesaplayıcı

Designed by Fatih GÜNDOĞDU

Irradiance (G) : 500 W/m2
Temperature (T) : 25 oC
Gmax : 1000 W/m2
Tmax : 100 oC
Pmax : 1367 Watt
Pmin : 834 Watt

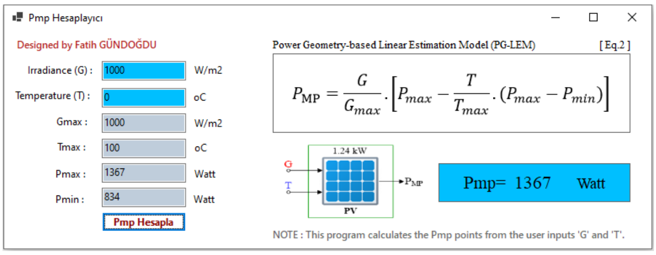
Power Geometry-based Linear Estimation Model (PG-LEM) [ Eq.2 ]

$$P_{MP} = \frac{G}{G_{max}} \cdot \left[ P_{max} - \frac{T}{T_{max}} \cdot (P_{max} - P_{min}) \right]$$

Pmp= 616,875 Watt

NOTE : This program calculates the Pmp points from the user inputs 'G' and 'T'.

**Figure 6-b.** Application screen for T=25 $^{o}$C ve G=500 W/m$^2$


Pmp Hesaplayıcı

Designed by Fatih GÜNDOĞDU

Irradiance (G) : 1000 W/m2
Temperature (T) : 0 oC
Gmax : 1000 W/m2
Tmax : 100 oC
Pmax : 1367 Watt
Pmin : 834 Watt

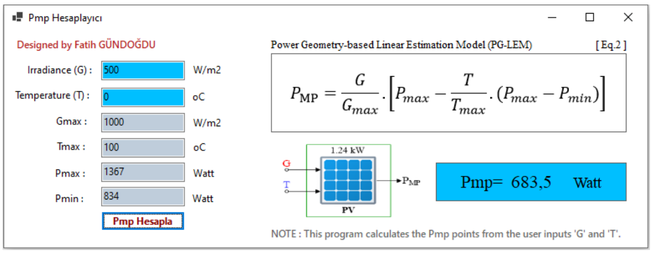
Power Geometry-based Linear Estimation Model (PG-LEM) [ Eq.2 ]

$$P_{MP} = \frac{G}{G_{max}} \cdot \left[ P_{max} - \frac{T}{T_{max}} \cdot (P_{max} - P_{min}) \right]$$

Pmp= 1367 Watt

NOTE : This program calculates the Pmp points from the user inputs 'G' and 'T'.

**Figure 7-a.** Application screen for T=0 $^{o}$C ve G=1000 W/m$^2$


Pmp Hesaplayıcı

Designed by Fatih GÜNDOĞDU

Irradiance (G) : 500 W/m2
Temperature (T) : 0 oC
Gmax : 1000 W/m2
Tmax : 100 oC
Pmax : 1367 Watt
Pmin : 834 Watt

Power Geometry-based Linear Estimation Model (PG-LEM) [ Eq.2 ]

$$P_{MP} = \frac{G}{G_{max}} \cdot \left[ P_{max} - \frac{T}{T_{max}} \cdot (P_{max} - P_{min}) \right]$$

Pmp= 683,5 Watt

NOTE : This program calculates the Pmp points from the user inputs 'G' and 'T'.

**Figure 7-b.** Application screen for T=0 $^{o}$C ve G=500 W/m$^2$

## 5. CONCLUSION

The results showed that the designed interface successfully performs fundamental functions such as real-time data entry, parameter adjustment, algorithm simulation, and visualization of the results. The designed interface plays a critical role in understanding system behavior by allowing the user to dynamically modify environmental parameters such as temperature and irradiance and numerically observe their instantaneous effects on the maximum power point of the PV system. It is effective in terms of both functionality and design. The logical grouping of parameter inputs, the use of descriptive labels, the user-guiding layout, and the ability of the developed software to be used for both educational and research–development purposes are among its significant advantages. In conclusion, this study demonstrates how an effective GUI design, serving as a bridge between theoretical control algorithms and practical applications, can add significant value to engineering education and system prototyping processes. The flexibility of C# and the Windows Forms application, combined with the powerful graphical libraries of the .NET framework, enabled the rapid and efficient development of such an application. Future work may further expand the scope and impact of the application by integrating the GUI with a real-time data acquisition card, adding cloud-based data logging features, or incorporating more advanced AI-based MPPT methods for simulation. In this way, the developed software can go beyond being merely a simulation tool and evolve into an operational platform for monitoring and optimizing real PV systems.

## REFERENCES

[1] R. Ahmad, A. F. Murtaza, H. A. Sher, “Power tracking techniques for efficient operation of photovoltaic array in solar applications-A review”, Renewable and Sustainable Energy Reviews, vol.101, pp.82-102, 2019..

[2] A. Gundogdu, “System identification-based ARV-MPPT technique for PV systems under variable atmospheric conditions,” IEEE Access, 10, 51325–51342, 2022.

[3] M. Farhat, et al., “Optimizing photovoltaic performance: Data-driven maximum power point prediction via advanced regression models,” Results in Control and Optimization, vol.20, 100586, 2025.

[4] V. H. Bui, et al., “Estimating the potential maximum power point based on thecalculation of short-circuit current and open-circuit voltage,” IET Power Electronics, vol.17, No.3, pp.402-421, 2024.

[5] E. Moshksar, T. Ghanbari, “Real-time estimation of solar irradiance and module temperature from maximum power point condition,” IET Power Electronics, vol.12, No.6, pp.807-815, 2018.

[6] K. M. Charu, et al. “An efficient data sheet based parameter estimation technique of solar PV,” Scientific Reports, vol.14, 6461, 2024.

[7] Elkholy, A. (2019). Optimal parameters estimation and modelling of PV modules from outdoor data.

[8] P. V. Mahesh, “Maximum power point tracking using decision-treemachine-learning algorithm for photovoltaic systems,” Clean Energy, vol. 6, No. 5, pp.762–775, 2022.

[9] N. M. Khan, et al., “Analysis of deep learning models for estimation of MPP and extraction of maximum power from hybrid PV-TEG: A step towards cleaner energy production”,Energy Reports, vol.11, pp.4759–4775, 2024.

[10] Carlson, D. E., & Wronski, C. R. “Amorphous silicon solar cells”, Applied Physics Letters, vol. 28, pp.671–673, 1976.

[11] B. Beder, O. Karahan, “Design of Smart Bin based on C# Through Raspberry PI”, Journal of ESTUDAM Information, Vol.4, No.1, pp.1-7, 2023.

[12] K. K. Cevik, E. Dandil, “Development of Visual Educational Software for Artificial Neural Networks on .Net Platform”, Bilisim Teknolojileri Dergisi, vol.5, No.1, pp.19-28, 2012.

[13] H. Guruler, U. Ayvaz, ” Designing an Assistant System Encouraging Ergonomic Computer Usage”, GU J Sci, Part C, vol.5, no.4, pp.99-107, 2017.

[14] N. Gundogan, C. Dogan, “Inverse Kinematic Analysis of a 5 DOF Gantry Type Welding Robot”, Tr. J. Nature Sci. Vol.14, No.2, pp.195-203, 2025.

[15] U. Kircil, C. Tepe, “Cross-Platform Telemetry System Design For Unmanned Ground Vehicles”, AKU J. Sci. Eng. Vol.24, 015203, pp.53-60, 2024.

[16] K. Sari, H. Dilipak, “A Design Software Based on The Placement of Bolts for Bolted Joints”, Journal of Engineering and Sciences, vol:1, no:2, pp.104-114, 2023.

[17] D. Yosunlu, E. Avaroglu, “Web-based Software Suit Development For Post Procesing Algorithms”, European Journal of Science and Technology, 28, pp.493-499, 2021.

[18] U. Kose, A. Tufekci, “Usage of an Intelligent Software System in Teaching Algorithm and Flowchart Concepts”, PEGEM Journal of Education and Instruction, vol.5, no.5, pp.569-586, 2015.

[19] S. Bayrakdar, D. Akgun, I. Yucedag, “An accelerated approach for facial expression analysis on video files”, Pamukkale University Journal of Engineering Sciences, vol.23, no.5, pp.602-613, 2017.

[20] S. Turan, T. Aydin, “Analysis of the agricultural product information by processing the images obtained with the help of a mobile robot”, Sakarya University Journal Of Science, vol. 22, no.2, pp.480-488, 2018.

[21] E. Deniz, "ANN-based MPPT algorithm for standalone photovoltaic PMSM drive system without dc-dc boost converter." IENSC2018 International Engineering and Natural Sciences Conference, Diyarbakir, Turkiye, 14-17 November 2018.

[22] I. Buldu ve M.E. Sahin, "A MATLAB/GUI based photovoltaic system simulator for estimation of PV parameter using Newton-Raphson method", Gazi Journal of Engineering Sciences, vol.7, no.3, pp. 196-212, 2021.

[23] S. Duman, N. Yorukeren, I. H. Altas, "Güneş Enerji Sistemleri için Bir Eğitim Aracı", ISITES2014, Karabuk, Turkiye, June 2014.

## BIOGRAPHIES

**Fatih Gundogdu** a third-year student in the Department of Computer Engineering at Karamanoglu Mehmetbey University. My interests include artificial intelligence, software development, data analytics, and hobby electronics. My goals include gaining experience in algorithm design, programming languages, and engineering solutions by participating in various software and hardware projects. In particular, I plan to pursue a career in innovative engineering applications, with a focus on defense industry technologies and AI-supported systems.

**Ahmet Gundogdu** Elazig in 1974. He received the B.S. and M.S. degrees in electrical teaching from the University of Firat. Elazig, in 2004 and the Ph.D. degree in electrical engineering from Firat University, Elazig, in 2012. He has been working as an Assoc. Prof. Dr. at Electrical Engineering Department of Batman University. Research interests include electrical machines, motor control, renewable energy, MPPT.

**Resat Celikel** was born in 1980. He received B.S, M.S and PhD degrees from Firat University, Elazig, Turkey. He has been working as an Assoc. Prof. Dr. at Firat University. His current interests are BLDC motors, flywheel energy storage, FPGA, renewable energy, MPPT.