**Solar Module Visualization and Power Estimation Application**

**MATLAB App Designer – Technical Documentation**

**1. Overview**

This application provides an interactive MATLAB App Designer-based interface for simulating and visualizing the structure of a photovoltaic (PV) module. Users can configure the physical characteristics of the module, simulate solar irradiance effects, compute power output, and view the geometry in multiple perspectives. The app supports both educational and design-oriented use cases.

**2. Objectives**

* Enable users to define a solar module with customizable dimensions and materials.
* Provide a realistic 3D visualization of the solar panel including structural layers.
* Allow irradiance-based color mapping of individual cells.
* Support power calculation based on real-time irradiance and cell parameters.
* Offer functionality to save and load complete module configurations.
* Improve interaction with the model via different camera views and component toggles.

**3. Key Features**

**3.1. Geometry Configuration**

The app allows detailed input of physical dimensions and materials, including:

* Module width and height
* Front and rear glass thickness
* Front and rear encapsulant thickness
* Number of cell rows and columns
* Cell spacing
* Frame thickness
* Cell dimensions (selectable from standard dropdown list)

**3.2. 3D Layered Visualization**

The solar panel is rendered as a layered structure using patch() and rectangle() objects. Layers include:

* Rear glass
* Rear encapsulant
* Photovoltaic cells
* Front encapsulant
* Front glass

**3.3. Frame Rendering**

An E-shaped aluminum frame is drawn at multiple vertical layers to simulate structural framing. A checkbox allows users to toggle the visibility of the frame.

**3.4. Camera Views**

Users can select from multiple viewing angles using a dropdown menu:

* Top
* Bottom
* Side
* Front
* Isometric

The app adjusts the camera position and projection method (e.g., orthographic) accordingly to ensure consistent visual representation.

**3.5. Irradiance and Cell Coloring**

Users can enter a manual irradiance value (in W/m²). The color of the solar cells changes dynamically based on the irradiance:

* Low irradiance produces cooler colors (blue)
* High irradiance produces warmer colors (red)  
  The app uses a continuous color gradient and displays a corresponding legend.

**3.6. Power Output Calculation**

The electrical power output is computed using:

P=GHI× Active Area ×η

Where:

* GHI the irradiance in W/m² (user-defined)
* Active Area ​ is the total active cell area
* η is the module efficiency

The computed value is displayed graphically in the plot or via annotation.

**3.87 Save and Load Configuration**

The current module configuration can be saved to a .mat file and reloaded later. Saved parameters include:

* Geometry
* Cell layout
* Material thicknesses
* Irradiance and efficiency
* Frame visibility
* View orientation

**4. User Interface Components**

| **Component** | **Function** |
| --- | --- |
| Numeric Edit Fields | Inputs for dimensions, spacing, thicknesses |
| Dropdown (Cell Size) | Select from standard cell dimensions (in mm) |
| Dropdown (View) | Select camera view mode |
| Checkbox (Frame) | Toggle visibility of the aluminum frame |
| Irradiance Input | Manually specify solar irradiance (W/m²) |
| Efficiency Input | Specify panel efficiency for power calculation |
| Save Button | Save current configuration to file |
| Load Button | Load configuration from file |

**5. How to Use the Application**

1. **Launch the app** in MATLAB.
2. **Configure module parameters** via the numeric input fields and dropdown menus.
3. **Enter irradiance and efficiency** as required.
4. **Change the view** using the View dropdown to inspect the model from different angles.
5. **Enable or disable the frame** using the checkbox.
6. **Observe the color of cells** change based on irradiance level.
7. **Read the power output** displayed in the plot title or a text annotation.
8. Use **Save/Load** to preserve and recall complete module setups.

**6. Dependencies and Requirements**

* MATLAB R2021a or later
* App Designer (built-in)
* Optional: MATLAB Compiler (for packaging as executable)
* Internet access (if using online irradiance APIs)