International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9
***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

A Novel Method for Self-Driving Solar-Powered Drones

Amam Hossain Bagdadee[1,2,] *Md. Rashed Rahman[1], Istiak Ali Khan[1], Khandakar Khalid Hossain[1]

[1]Department of Electrical and Electronic Engineering, Presidency University, Dhaka, Bangladesh
[2]College of Energy and Electrical Engineering, Hohai University, Nanjing, China
amamb@pu.edu.bd, a.bagdadee@hhu.edu.cn

Abstract— This project presented the transformative potential of integrating solar panels into drones. This innovative approach addresses the long-standing issue of limited battery life, enabling drones to operate continuously, adapt to changing mission demands, and contribute to sustainability efforts in the field of unmanned aerial vehicles. This development represents a significant step forward in the evolution of drone technology, promising a more versatile and self-sustaining future for drones across various sectors. Factors like high-speed flight, aggressive maneuvers, heavy payloads, and adverse weather can dramatically reduce battery life. Surveillance drones, for instance, are confined to covering limited areas before returning for battery changes or recharging. A groundbreaking solution lies in the incorporation of solar panels directly into the drones, allowing them to self-charge when required. This innovation ensures uninterrupted drone operation, regardless of the prospects of energy demands, thus marking a significant step forward in drone technology. With this integration of solar power, drones are poised to become not only versatile but also autonomous, promising a transformative development in the world of unmanned aerial vehicles.

Keywords- Solar panels, Drone, Sustainability, battery life, Flight duration

I. INTRODUCTION

Drones and solar panels are two of the world's most promising developing technologies; their combined use may increase their value even further. Global Market Insights, Inc. survey predicts that the commercial drone market will reach $17 billion by 2024 [1,2]. By 2023, the installed solar capacity worldwide is expected to reach one terawatt, or one trillion watts, according to the most recent predictions from Wood Mackenzie Power and Renewables [3]. Both technologies, drones and solar panels, are progressing rapidly and becoming increasingly affordable. However, a significant challenge faced by drones is their limited endurance [4]. This issue can be effectively addressed by leveraging the power of the sun to provide nearly continuous energy. Nowadays drones primarily rely on batteries for their electricity, necessitating regular returns to the ground for recharging, typically after just a few hours of operation. Given this scenario, the logical step forward is to harness solar energy as a sustainable and efficient power source for drones. Nowadays, drones are frequently seen and employed for a variety of purposes. Military surveillance, spraying pesticides, and taking photos [5,6]. The issue with surveillance and monitoring is that many users need extended surveillance. Although they have a significant disadvantage, drones offer a good surveillance monitoring view. This is the battery life of the drone. The main concern a drone pilot has when conducting surveillance is that the drone's battery would die, and it might crash into a tree, a building, or another difficult-to-reach object, rendering it unable to be retrieved and charged [7]. This also applies to military surveillance; drone pilots' abilities are constrained by the potential of a drone's battery dying and becoming inoperable. We're working on a drone that addresses these problems by using solar energy to continuously recharge it, extending its range, enabling it to land anywhere, and automatically recharging its battery so it can take off later. Longer flight periods and drones that charge their batteries automatically in inaccessible places so they can take off from the same position after charging as a result [8]. This project aims to contribute to the advancement of UAV technology and provide a valuable tool for researchers, environmentalists, emergency responders, and various other stakeholders. Our innovative UAV offers the promise of revolutionizing unmanned systems, enabling them to transcend traditional boundaries and operate efficiently in both terrestrial and aquatic domains. Drones are frequently utilized in medicine and surveillance. The most common uses of drones in medicine are the provision of crisis evaluations in the absence of other sources of reach and the distribution of medicines, aid packages, blood, vaccinations, and other types of clinical items. Drones have numerous benefits in terms of surveillance, such as crime prosecution, traffic monitoring, and border monitoring, among others [9]. Nonetheless, even the most advanced drones now have flight periods ranging from 20 to 25 minutes. Drones' low battery life causes issues during surveillance and renders their use impractical in all domains. Because of the widespread use of drones in surveillance and medical, it is critical to find a means to enhance the flight time of these drones without interfering with the flights. Onboard batteries with limited capacities often power drones. These drones are expected to perform critical and longer missions. As a result, there is a need for an autonomous mechanism that can extend drone flying length and allow tasks to be performed without interruption. It helps to lengthen the operating time of drones.

UAVs have various applications, including gathering information in hostile areas and providing support in devastated regions where human assistance may be limited.

These UAVs must be portable and reliable for repeated use. Military-grade UAVs designed for such uses are quite expensive and tailored to specific requirements. Therefore, the aim is to create cost-effective alternatives. The envisioned UAV design includes an HD camera with target recognition, a weight under 55 lbs, and a GPS-guided autopilot for precise flight control.

II. LITERATURE REVIEW

A. Drone design

The subsequent phase in the design process involved the selection of an appropriate microcontroller for the project's execution. The initial step was to ascertain the essential hardware interfaces, such as flight controllers' modules, motor actuators, and sensors. The decision regarding the microcontroller to be used hinges on whether digital or analog functions are required [10]. The Pixhawk flight board microcontroller is the most adept choice when serial-to-parallel conversion or vice versa is necessary. The next design step was planning how to position the motors and electronic speed controllers [11]. This was crucial because different motors require special circuits and specific battery eliminator circuits.

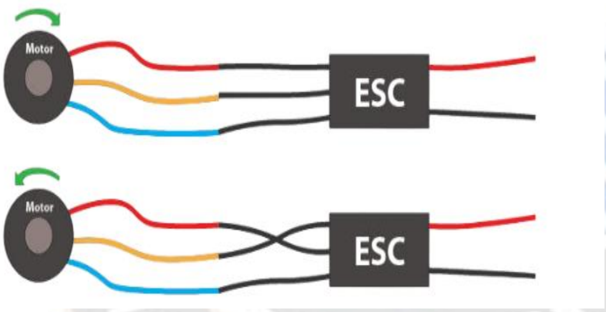

Figure 1. Motor and ESC connection

The following design step involved choosing the right propeller type, size, and dimensions, considering how these aspects impact the design's aerodynamics and how easy it is to control.

Figure 2. Propellers, 2 Pairs of Red and Blue Colors, Each with Clockwise and Counterclockwise

A quadcopter is a compact vehicle equipped with four propellers at the ends of a cross-shaped frame. This configuration employs fixed-pitch rotors to manage the vehicle's movement. The four rotors operate independently, enabling precise control of the Quadcopter's pitch, roll, and yaw attitudes [12].

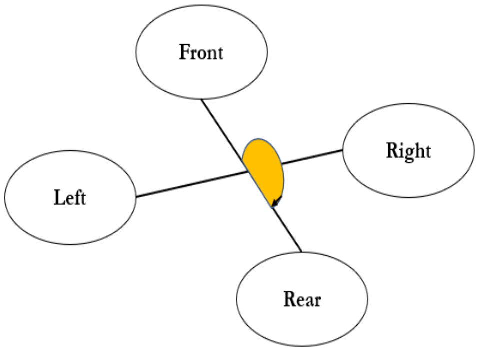

Figure 1 Pitch Direction of Quadcopter

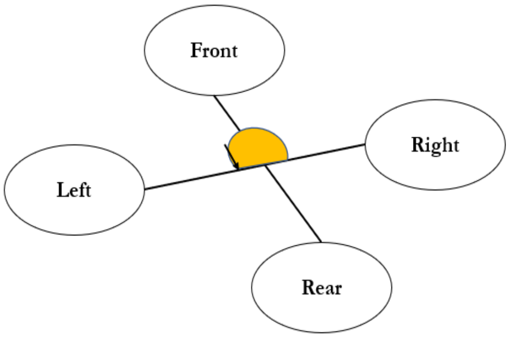

Figure 2 Roll Direction of Quadcopter

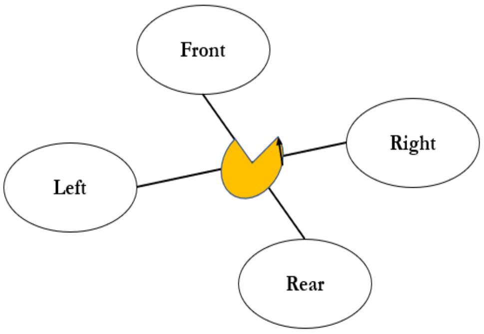

Figure 3 Yaw direction of Quadcopter

A quadcopter relies on four input forces, primarily generated by the propellers connected to its rotors. The Quadcopter's motion is regulated by adjusting the thrust produced by these propellers [13]. This thrust can be managed by altering the speed of each rotor. The take-off involves the Quadcopter lifting off from the ground to reach a hovering position, while landing is the reverse, bringing it back to the ground. These motions are governed by simultaneously increasing (for take-off) or decreasing (for landing) the speed of all four rotors, which, in turn, controls the vertical movement [14]. The

---

figures provide visual representations of the Quadcopter's take-off and landing actions.

Figure 4 Take-off Motion

Figure 5 Landing Motion

The Quadcopter's forward and backward movement is managed by adjusting the speed of the rear (for forward) or front (for backward) rotor. When the speed of the rear (front) rotor simultaneously influences the pitch angle of the Quadcopter[15]. The illustrations in the figure depict the Quadcopter's forward and backward actions.

Figure 6 Forward Motion

Figure 7 Backward Motion

To achieve lateral motion that can adjust the yaw angle of the Quadcopter. This is done by increasing (decreasing) the speed of the counterclockwise rotors while decreasing (increasing) the speed of the clockwise rotors. The below figures illustrate the Quadcopter's right and left movements.

Figure 8 Right Motion

Figure 9 Left Motion

A quadcopter maintains a stable hovering or static position by having two pairs of rotors, with one pair rotating clockwise and the other counterclockwise at the same speed. This setup generates equal and opposite reaction torques, ensuring a net torque of zero, which is essential for the Quadcopter to remain stationary in the air [16]. The schematic diagram of the Quadcopter's movement, as depicted in the figure, serves as the foundation for deriving its mathematical model.

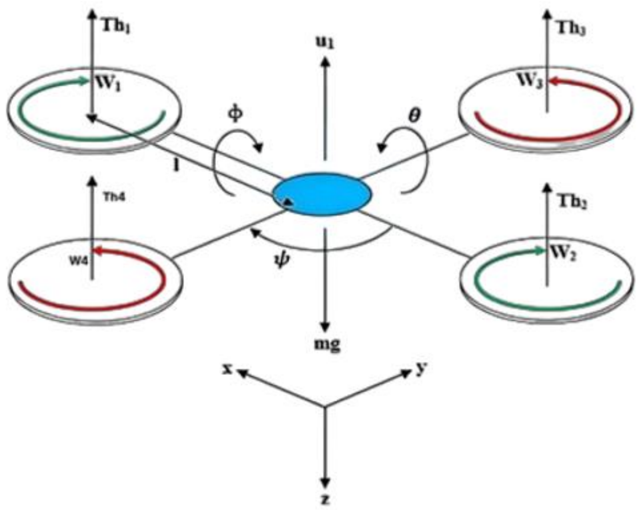

Figure 10 Schematic of Quadcopter

Finally, during the construction phase, a housing made from 6-inch plywood was built to serve as the frame for the Quadcopter [17]. The design and dimensions of this frame are detailed in the Appendix. The flight control board was securely mounted on a soft, shock-absorbing material, which was intended to safeguard the gyro, accelerometer, and Inertia Measurement Unit (IMU) [18].

Figure 11 HJ Quadcopter frame (HJ frame 2016).

- *Solar Panel*

The 5v 200mA Mini Solar Panel for DIY Projects is ready to use without requiring a frame or special modifications [19]. Implement these Polycrystalline solar cells because they are laser cut to the proper size and encapsulated in special sun and weather-resistant materials, which give them unique characteristics [20,21]. The 5v 200mA Mini Solar Panel has Polycrystalline solar cells, which are encased and protected by a durable outer poly frame. This 5v 200mA Mini Solar Panel for DIY Projects is lightly weighted, very strong, and weather-resistant substrates or injection molded trays custom-designed for the target product. These Small Epoxy Solar Panels are simple to install or add to your existing product, and their construction requires no frame or special modifications. Polycrystalline solar cells have 2 to 3 times the power of amorphous thin-film solar panels [22]. Very small space is required for installation and to connect a 5v 200mA Mini Solar Panel; just solder or crimp to the copper tape.

Figure 15 Solar panel

Here,

- **E**: Energy stored in the drone's system (in Joules).
- **P**: Power input from the renewable energy source (in Watts).
- **t**: Time (in seconds).
- **Endurance**: The time a drone can remain in flight before requiring recharging or refueling (in seconds).
- **Efficiency**: The efficiency of the renewable energy source in converting input energy to power output.
- **Payload**: The weight the drone can carry (in kilograms).
- **Range**: The maximum distance a drone can travel before needing recharging or refueling (in meters).

Model for Solar-Powered Drone:

Energy Input from Solar Panel:

$$PSolar = Solar\ Panel\ Efficiency \times Solar\ Irradiance \quad (1)$$

Energy Storage: $$ESolar(t) = \int PSolardt \quad (2)$$

Endurance for Solar-Powered Drone:

$$Endurance_{\text{Solar}} = \frac{ESolar}{PConsumed} \quad (3)$$

Payload-to-Weight Ratio: $$PayloadSolar = \frac{Payload}{DroneWeight} \quad (4)$$

Range for Solar-Powered Drone:

$$RangeSolar=\frac{SolarEnduranceSolar\times Drone\,Speed}{Payload} \qquad (5)$$

III. METHODOLOGY

This Methodology thoroughly covers the quadcopter design, breaking it down into its component requirements, the construction phase, and testing the project's hardware and software aspects.

A. *Development of Drone*

Firstly, need to assemble a frame from a lightweight material that will allow it to be lifted off the ground. A quadcopter is a device that integrates mechanical, electronic and aviation concepts strongly (late 19th century). The Quadcopter has four motors, and changing the Quadcopter's speed and rotation direction causes it to go in the desired direction (Take-off motion, Landing motion, Forward motion, backward motion, Left motion, Right Motion.) The delivered signal from the 6-channel transmitter changes how the motors rotate. The net aerodynamic thrust is produced by turning rotors one and three in the clockwise direction and rotors two and four in the opposite direction. The signal from the microcontroller is sent to the ESCs, which then regulate the motor speed. The final step is the controller, solar power, and prototype manufacturing interface. It will be possible to use GPS and write software for modules.

B. ***Hardware Design***

The hardware design phase primarily involves identifying the system's physical components, elucidating their interconnections, and delineating their integration within the overall system architecture. Additionally, it involves delineating the specifications for the actual hardware and the construction of circuits. The tasks encompassed in the hardware design and development process include:

- Deliberating on design considerations.
- Crafting a detailed block diagram for visualization.
- Discerning the most suitable flight board controller for the task.
- Carefully pinpointing the requisite motors and solar panels and formulating their design approach.
- Meticulously drafting the all-encompassing circuit diagram.

The initial step involved in the process was to conduct a design assessment, wherein an examination of the different elements was carried out, and the selection of the components to be utilized was determined.

Table 1 Parts and components

Parts	**Quantity**	**Specification**	**Weight**
Frame	1	HJ Model	300g
Motors [19g Each]	4	Model Number [A221213T-1000KV]	76g
Propellers	4	10x45	15g
ESC [13g Each]	4	30A SimonK Model	52g
Battery	1	LIPO	200g
Flight Controller	1	Pixhawk 2.4.8	15.8g
Connectors	20	Deans Ultra Plugs Gold Bullet	30g
Solar panel	4	Capable of Charging 5 V	120g
Charging Module	4	1A Current Adjustable	80
Total	**39**	**Specified**	**888.8g**

The system's diagram is given below in Figure 126

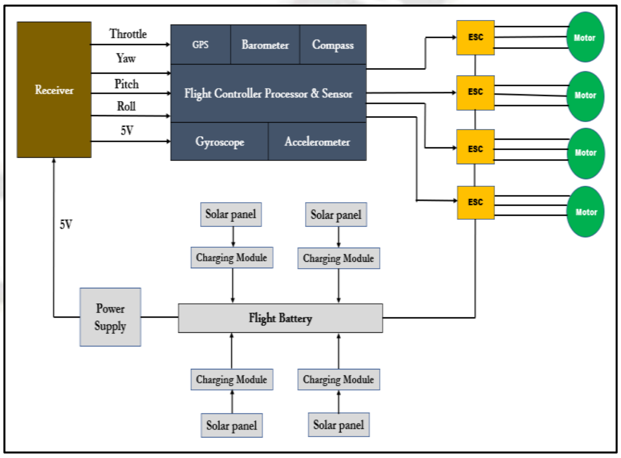

Figure 126 System's block diagram

C. ***Circuit Programming***

For the Quadcopter to operate effectively, it must possess the capability to receive commands and carry out tasks as instructed from a remote base station, even if it's located miles away from the drone's physical location. As a result, circuit programming is a critical component and the core of the prototype's functionality. The flight controller board incorporates four customized buttons and a liquid crystal display screen to facilitate the programming process.

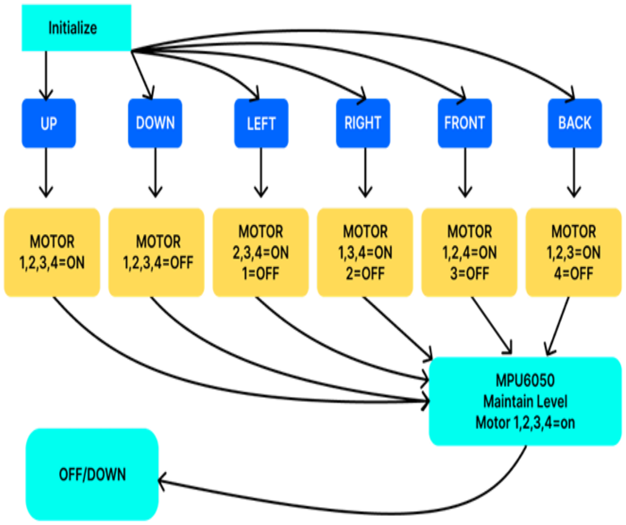

Figure 17 Programming Algorithm

Circuit programming description for a quadcopter's flight controller is a complex task, and the actual programming would involve software development. However, provide an overview of the main components and functions that might be included in the circuit programming of a quadcopter's flight controller:

Components of the Flight Controller Programming:

- ***Microcontroller**:* The flight controller typically uses a microcontroller (e.g., Arduino, Pixhawk) as its core processing unit. This microcontroller receives input from various sensors and the remote base station, processes the data, and sends control signals to the motors and other components.
- ***Sensors**:* Quadcopters are equipped with various sensors to measure parameters such as orientation, acceleration, GPS position, and altitude. These sensors provide crucial data for the flight controller to maintain stability and execute flight commands.
- ***Remote Communication**:* The flight controller communicates wirelessly with a remote base station or transmitter. It receives control commands (throttle, roll, pitch, and yaw) and telemetry data from the base station, allowing a remote pilot to control the Quadcopter.
- ***Control Algorithms**:* Flight control algorithms are implemented in the programming to stabilize the Quadcopter. These algorithms use sensor data to adjust motor speeds and angles to maintain desired flight characteristics.
- ***LCD Display**:* The liquid crystal display (LCD) screen provides visual feedback and information to the pilot, including battery status, telemetry data, and system status. It can also be used to set parameters and view diagnostic information.

Here's an algorithmic representation of the components involved in flight controller programming for a quadcopter:

```
# Initialize the flight controller
def initialize_flight_controller():
    # A. Microcontroller Initialization
    microcontroller = initialize_microcontroller()  # e.g., Arduino or Pixhawk
    # B. Sensor Initialization
    sensors = initialize_sensors()  # Initialize orientation, acceleration, GPS, and altitude sensors

    # C. Remote Communication Setup
    communication = initialize_remote_communication()  # Set up communication with the remote base station
    # D. Control Algorithm Initialization
    control_algorithms = initialize_control_algorithms()  # Implement flight control algorithm
    # E. LCD Display Initialization
    lcd_display = initialize_lcd_display()  # Initialize the LCD screen
    # Return the initialized components
    return microcontroller, sensors, communication, control_algorithms, lcd_display
# Main flight control loop
def flight_control_loop(microcontroller, sensors, communication, control_algorithms, lcd_display):
    while True:
        # Read sensor data
        sensor_data = read_sensor_data(sensors)
        # Receive control commands and telemetry data from the remote base station
        control_commands, telemetry_data = receive_remote_commands(communication)
        # Execute flight control algorithms to stabilize the quadcopter
        motor_speeds, motor_angles = execute_control_algorithms(control_algorithms, sensor_data, control_commands)
```

```
        # Send control signals to the motors and other
components send_control_signals(microcontroller,
motor_speeds, motor_angles)
        # Display relevant information on the LCD screen
        update_lcd_display(lcd_display, telemetry_data)
# Sample functions for initialization and data handling
def initialize_microcontroller():
    # Initialization logic for the microcontroller
    return Microcontroller()  # Replace with actual
microcontroller initialization
def initialize_sensors():
    # Initialization logic for sensors
    return Sensors()  # Replace with actual sensor initialization
def initialize_remote_communication():
    # Initialization logic for remote communication
    return Communication()  # Replace with actual
communication setup
def initialize_control_algorithms():
    # Initialization logic for flight control algorithms
    return ControlAlgorithms()  # Replace with actual algorithm
implementation
def initialize_lcd_display():
    # Initialization logic for the LCD display
    return LCDDisplay()  # Replace with actual LCD
initialization
def read_sensor_data(sensors):
    # Read data from orientation, acceleration, GPS, and
altitude sensors
    return sensors.read_data()

def receive_remote_commands(communication):
    # Receive control commands and telemetry data from the
remote base station
    return communication.receive_commands(),
communication.receive_telemetry()

def execute_control_algorithms(control_algorithms,
sensor_data, control_commands):
    # Execute flight control algorithms to adjust motor speeds
and angles
    return control_algorithms.execute(sensor_data,
control_commands)

def send_control_signals(microcontroller, motor_speeds,
motor_angles):
    # Send control signals to motors and other components
    microcontroller.send_signals(motor_speeds, motor_angles)

def update_lcd_display(lcd_display, telemetry_data):
    # Update the LCD display with relevant information
    lcd_display.update(telemetry_data)

# Entry point for flight controller program
if __name__ == "__main__":
    microcontroller, sensors, communication,
control_algorithms, lcd_display = initialize_flight_controller()
    flight_control_loop(microcontroller, sensors,
communication, control_algorithms, lcd_display)
```

This algorithm outlines the initialization of flight controller components, the main flight control loop, and sample functions for each component's initialization and data handling. It illustrates how the microcontroller, sensors, communication, control algorithms, and LCD display work together to stabilize and control the quadcopter during flight, while also providing feedback to the remote pilot. The specific implementation details and functions should be customized based on the hardware and software platform used in the flight controller.

Programming Functions:

- ***Initialization:*** When the Quadcopter is powered on, the flight controller initializes sensors, communication modules, and other components.
- ***Data Acquisition:*** The flight controller continuously reads data from onboard sensors, including gyroscopes, accelerometers, barometers, GPS, and magnetometers. This data is used to calculate the Quadcopter's orientation, position, and altitude.
- ***Remote Control Reception:*** The microcontroller listens for control commands from the remote base station. It interprets commands for throttle (altitude control), roll, pitch, and yaw adjustments.
- ***Control Algorithms:*** The flight controller runs control algorithms to stabilize the Quadcopter. PID (Proportional-Integral-Derivative) controllers are commonly used to adjust motor speeds and angles to maintain desired flight characteristics.
- ***Telemetry Transmission:*** Telemetry data, including battery voltage, altitude, speed, and GPS coordinates, is periodically transmitted to the remote base station. This data can be displayed on the LCD screen and monitored by the pilot.
- ***Safety Protocols:*** The programming should include safety features, such as low-battery warnings, automatic return-to-home in case of signal loss, and fail-safes to prevent dangerous maneuvers.
- ***User Interface:*** If the Quadcopter incorporates buttons or switches on the flight controller, the

programming should allow the pilot to configure settings modes and initiate specific actions.

- ***Calibration**:* The flight controller should include calibration routines for sensors and other components to ensure accurate measurements and optimal performance.
- ***Advanced Features***: Depending on the Quadcopter's capabilities, programming can include advanced features such as GPS waypoint navigation, follow-me modes, and obstacle avoidance.
- ***Solar Power Management**:* If solar power integration is part of the design, the programming should include the ability to monitor and manage solar panel output and its impact on battery charging and flight time.

It's important to note that the specific programming code for a flight controller can be quite extensive and may involve multiple software modules. Additionally, open-source flight control software, such as ArduPilot, may be used as a foundation, which can be customized to fit the specific requirements of the Quadcopter. The actual programming process would require expertise in embedded systems and control theory.

Here's an algorithmic representation of the key programming functions for a flight controller in a Quadcopter:

```
# A. Initialization
def initialize_flight_controller():
    initialize_sensors()
    initialize_remote_communication()
    initialize_control_algorithms()
    initialize_lcd_display()
    initialize_safety_protocols()
    initialize_user_interface()
    initialize_calibration()
    initialize_solar_power_management()
    initialize_advanced_features()

# B. Data Acquisition
def acquire_sensor_data():
    read_gyro_data()
    read_accelerometer_data()
    read_barometer_data()
    read_gps_data()
    read_magnetometer_data()
    calculate_orientation()
    calculate_position()
    calculate_altitude()

# C. Remote Control Reception
def receive_control_commands():
    throttle_command = receive_throttle_command()
    roll_command = receive_roll_command()
    pitch_command = receive_pitch_command()
    yaw_command = receive_yaw_command()

# D. Control Algorithms
def run_pid_controllers():
    adjust_motor_speeds()
    adjust_motor_angles()

# E. Telemetry Transmission
def transmit_telemetry_data():
    transmit_battery_voltage()
    transmit_altitude()
    transmit_speed()
    transmit_gps_coordinates()

# F. Safety Protocols
def monitor_battery_voltage():
    if low_battery_voltage():
        initiate_low_battery_protocol()

def signal_loss_handler():
    if signal_lost():
        initiate_return_to_home_protocol()

# G. User Interface
def configure_settings():
    if user_input():
        initiate_setting_change()

# H. Calibration
def calibrate_sensors():
    calibrate_gyro()
    calibrate_accelerometer()
    calibrate_magnetometer()

# I. Advanced Features
def waypoint_navigation():
    if gps_navigation_mode():
        navigate_to_waypoint()

def follow_me_mode():
    if follow_me_activated():
        follow_pilot()

def obstacle_avoidance():
    if obstacle_detected():
        avoid_obstacle()

# J. Solar Power Management
def monitor_solar_panel_output():
    check_solar_panel_voltage()
```

```
    adjust_battery_charging()

# Main Flight Control Loop
def flight_control_loop():
    while True:
        acquire_sensor_data()
        receive_control_commands()
        run_pid_controllers()
        transmit_telemetry_data()
        monitor_battery_voltage()
        signal_loss_handler()
        configure_settings()
        calibrate_sensors()
        waypoint_navigation()
        follow_me_mode()
        obstacle_avoidance()
        monitor_solar_panel_output()

# Entry point for flight controller program
if __name__ == "__main__":
    initialize_flight_controller()
    flight_control_loop()
```

This algorithm outlines the main programming functions for a Quadcopter's flight controller. It includes initialization, data acquisition, remote control reception, control algorithms, telemetry transmission, safety protocols, user interface, calibration, advanced features, and solar power management. These functions work together to ensure stable flight, safety, and efficient use of available resources. Specific implementation details should be customized based on the hardware and software platform used in the flight controller.

D. Mathematical Model:

This mathematical model for a solar-based self-charging monitoring drone involves several variables and factors that need to be considered. Here's a simplified model that takes into account the key parameters:

Let:

- B(t) is the battery level of the drone at time t (measured in Watt-hours, Wh).
- P_ solar (t) is the power generated by the solar panels at time t (measured in Watts, W).
- P_ consumption (t) is the power consumption of the drone at time t (measured in Watts, W).
- η_ solar is the efficiency of the solar panels, typically less than 1.
- B_ max is the maximum battery capacity of the drone (measured in Watt-hours, Wh).
- B_ initial is the initial battery level of the drone (measured in Watt-hours, Wh).
- η_ battery is the efficiency of the battery charging and discharging, typically less than 1.
- E_ loss is the energy loss in the charging and discharging processes.

The following differential equation can describe the battery level of the drone:

$$dB(t)/dt = \eta\_solar \ * \ P\_solar(t) - P\_consumption(t) \qquad (6)$$

The power generated by the solar panels depends on factors such as solar panel efficiency (η_ solar), solar radiation (I), and the area of the solar panels (A_ solar). It can be represented as:

$$P\_solar(t) = \eta\_solar \ * A\_solar \ * I(t) \qquad (7)$$

The power consumption of the drone (P_ consumption) depends on the flight mode, mission, and operational factors.

The energy losses in the battery charging and discharging processes can be represented as:

$$E\_loss = 1 - \eta\_battery \qquad (8)$$

The battery capacity should be constrained within its maximum capacity:

$$0 \le B(t) \le B\_max \qquad (9)$$

The initial battery level can be set as:

$$B(0) = B\_initial \qquad (10)$$

The goal is to maintain a positive battery level throughout the monitoring mission:

$$B(t) \ge 0 \ for \ all \ t \qquad (11)$$

The mission can be defined by specifying the solar radiation (I(t)), power consumption (P_ consumption(t)), and other relevant parameters over time.

This mathematical model can be used in simulations and numerical methods to predict the drone's behavior assess its ability to maintain a positive energy balance and optimize its operation. Keep in mind that this is a simplified model, and in practice, additional factors and complexities, such as variations in solar radiation, environmental conditions, and mission-specific requirements, would need to be considered for a more accurate representation of a solar-based self-charging monitoring drone.

IV. REAL IMPLEMENTATION

To initiate the assembly process, start by affixing a female Dean's ultra-plug to the battery. Create a corresponding pigtail by using the male Dean ultra-plug and short segments of 12-gauge wire for both the power and ground connections. Alternatively, it can purchase a pre-made pigtail with a male plug for convenience. Ensure extreme caution to prevent any wiring mix-up; double-check the connection by joining the male and female plugs. Next, solder the pigtail connection to the relevant terminals on the hub's power distribution mains. Specifically, solder the red wire to the positive '+' terminal and the black wire to the negative terminal to complete the setup.

***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

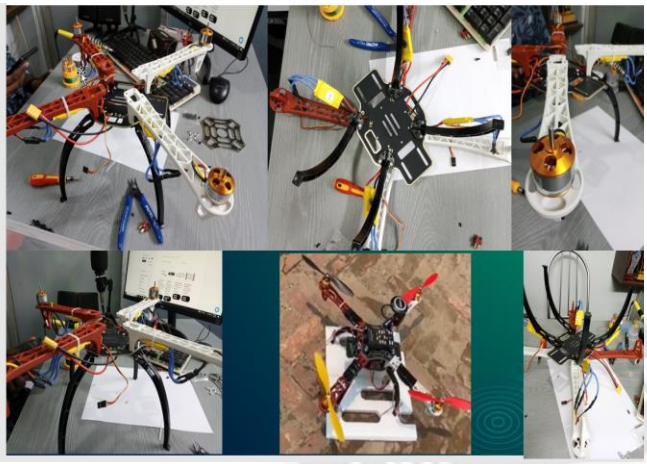

Figure 17 Motor Connections and Frame Assembly

Next, begin the process by preparing the Electronic Speed Controllers (ESCs). This involves soldering three female bullet connectors to each of the three black wires, which will be used to connect the ESCs to the motors. After completing this step, attach the arms to the hub, making sure that the white arms are positioned at the front and the red arms are at the rear. Secure the motors to the arms using the screws provided in the flame wheel kit. Subsequently, fasten an ESC to each arm using cable ties and connect them to their respective motors using the bullet connectors. Trim the positive and ground wires that come out from the ESCs to the appropriate length and solder them to the corresponding terminals on the Power Distribution Board (PDB). As a point of reference for future steps, establish that the side of the Quadcopter with the white arms is considered the front, while the side with the red arms is designated as the rear. This orientation will remain consistent throughout the assembly process and beyond. The red and white arms will serve as visual indicators to determine the Quadcopter's direction from a distance.

At this point, the assembly is nearly finished. The remaining tasks involve installing the receiver and the flight controller and then connecting all the components. The receiver should be linked to the left side, while the motors should be connected to the right side of the Pixhawk controller board. Affix the receiver to one of the side panels using double-sided tape. For securing the flight controller, it can utilize the foam padding that it originally came with. Apply hot glue to attach the controller board to the padding and then affix the padding to the hub's mounting plate. This approach not only protects the flight controller from potential damage during rough landings but also reduces vibrations transmitted from the frame to the flight controller, thereby enhancing its performance.

After ensuring that all components are properly installed, perform testing on both the receiver and the Pixhawk flight controller to verify the proper functioning of all systems. It's important to keep in mind that the Pixhawk controller receives its power through the ESC connected to motor 1. Therefore, if the connection to motor 1 and the battery linked to the Power Distribution Board (PDB) is not established, the flight controller will not receive power. Once satisfied with the testing of the Pixhawk, it can proceed to attach the top plate to the hub.

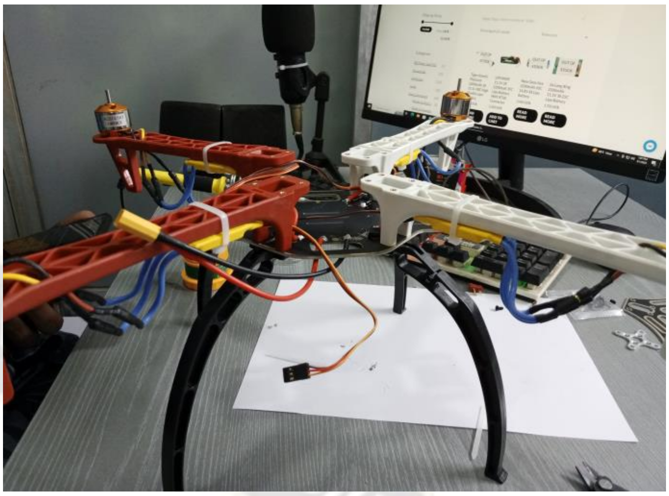

Figure 18 Complete Layout of the Quadcopter

Prior to mounting the propellers, it's essential to verify that each motor is spinning in the correct direction. Each motor is assigned a distinct number, commencing with the front left motor designated as number 1, followed by the front right motor, and so on in a clockwise sequence. As a customary practice, motors 1 and 3 should rotate in a clockwise direction, while motors 2 and 4 should spin counterclockwise.

To confirm the motor directions, we will need to activate the Pixhawk. This is accomplished by manipulating the left stick on the transmitter, moving it downward (reducing throttle) and to the right. Maintaining this position for a few seconds arms the Pixhawk and subsequently raising the throttle stick will engage the motors, prompting them to start spinning. Checking the rotation direction for each motor should be a swift and uncomplicated procedure. Suppose the motor is spinning in the incorrect direction. In that case, it can easily be rectified by disconnecting any two of the wires that link the Electronic Speed Controller (ESC) to the motor and then exchanging their positions. It doesn't make a difference which two wires of switch; the motor will change its rotation direction. This is a distinctive attribute of a stepper motor.

Figure 19 Complete Wiring of the Flight Control Board, Receiver, and the Motors

At this stage, the Quadcopter is prepared for propeller installation that can initiate the tuning of the flight controller. Start by ensuring that the gyroscopes are calibrated and zeroed out on a level surface. Next, proceed with adjusting the proportional (P) and integral (I) gains on the controller to fine-tune the self-leveling mode, which enhances the stability of the Quadcopter during flight. This tuning process is particularly valuable for FPV (First-Person View) flying and aerial photography, as it helps to reduce the overall agility of the vehicle, providing smoother and more controlled flight.

Constructing a solar panel-based drone project involves utilizing four small solar panels to replenish a LiPo battery. To get started, prepare the solar panels by ensuring their cleanliness and then connect them to the charge controller, which regulates the electricity flow from the solar panels to the LiPo battery. Securely affix the solar panels to the drone's frame, maximizing their exposure to sunlight during flight. Position the LiPo battery in its designated compartment on the drone frame and link it to the charge controller. Assemble the drone's frame, which includes components like motors, propellers, and the flight controller. Ensure that all wiring is neatly organized and firmly secured inside the drone. Test the system to confirm that the LiPo battery charges when it's low and that the solar panels generate electricity. Program the flight controller to monitor battery levels and switch to solar charging as needed. Calibration and test flights are essential to verify the proper functioning of the system. Always adhere to safety guidelines, especially when dealing with LiPo batteries and soldering.

Figure 20 Project design

V. RESULT

A. *Calculation*

Total weight of the drone =1000 gram=1killogram
Using the Bench Testing method to get per motor thrust = 396 gram
Total thrust= 396*4=1584 gram
Drone weight < Total thrust
So, the drone will fly.

B. *Battery Charging Time*

Battery =2200 mAh =2.2 Ah *80%=1.76 Ah (Battery can drain 80% charge)
Solar panel voltage= 20 volt
6 solar panel= 1.5*4=0.6 Ampere
Battery voltage =11.11 Volt
Charging time = Ah/A = 1.76/0.6= 3 hour
(Here Ah =Ampere hour, A= ampere)

Table 2 Overall Performance of the Drone

Throttle	Current (Ampere)	Power (Watt)
For 30%	3.25	23
For 50%	6.50	70
For 70%	9.25	91
For 100%	13.00	144
Average	32/4=8 Ampere	328/4=82Watt

$$Battery\ backup\ time = \frac{Battery\ voltage*AH}{Average\ voltage} = \frac{11.1*2.2}{82} = 17{,}70\ minutes \quad (12)$$

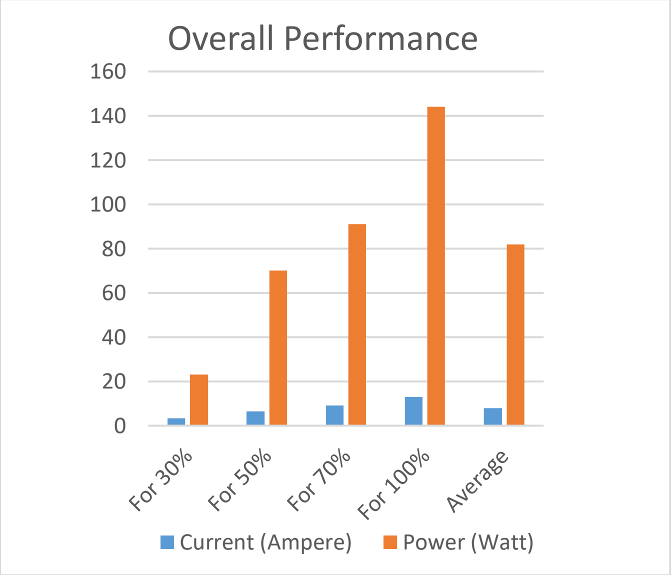

Figure 21 shows the drone's flight time, battery size.

Table 2 contains the data that appears to represent the drone's electrical characteristics at different throttle settings, likely for different flight modes or operational conditions. Here's a description of the table:

1) ***Throttle and Power Relationship****: The table presents data that illustrates the relationship between throttle* *percentage and both current (measured in Amperes) and power (measured in Watts). As the throttle percentage increases, both the current and power consumption of the system also increase.*

2) ***Direct Proportionality****: The data shows a direct proportionality between the throttle setting and the current drawn by the system. In other words, as the throttle setting is increased, the current consumed by the system increases linearly. This indicates that the throttle setting directly controls the amount of current used.*

3) ***Power Consumption****: Similarly, the table demonstrates that power consumption is directly proportional to the throttle setting. As the throttle setting is increased, the power consumption of the system also increases proportionally. This suggests that the power drawn by the system is dependent on the throttle setting.*

4) ***Variation with Throttle Percentage****: The table provides specific data points for different throttle percentages, including 30%, 50%, 70%, and 100%. The current and power values increase as the throttle percentage is raised, with the highest values recorded at 100% throttle.*

5) ***Average Current and Power****: The table also calculates the average current and power values. This information is useful for understanding the overall power consumption of the system during different operating conditions. In this case, the average current is calculated as 8 Amperes, and the average power is calculated as 82 Watts.*

6) ***Control Over Consumption****: The data highlights the importance of throttle control in managing the current and power consumption of a system. By adjusting the throttle setting, users can control the energy consumption of the device.*

7) ***Applications****: These findings are valuable for understanding the power requirements of a device at different throttle settings. This information is crucial in applications where power management and efficiency are essential, such as electric vehicles, drones, and other electric systems.*

In summary, the table provides insights into the relationship between throttle settings, current consumption, and power consumption. It underscores the direct proportionality between throttle percentages and both current and power, which can be useful for optimizing the performance and energy efficiency of systems where throttle control is a critical factor.

The figure 21 provides the following information:

- At 30% throttle, the drone draws a current of 3.25 Amperes, consuming 23 Watts of power.
- At 50% throttle, the current increases to 6.50 Amperes, and the power consumption rises to 70 Watts.
- With the throttle set at 70%, the drone draws 9.25 Amperes and consumes 91 Watts.
- At 100% throttle, the highest setting, the drone draws 13.00 Amperes and consumes 144 Watts.

Additionally, this graph provides an "Average" row, which appears to summarize the data. It calculates the average current (8 Amperes) and the average power (82 Watts) based on the values at the different throttle settings.

This information is essential for understanding the drone's energy consumption under different operating conditions. It can be valuable for optimizing the drone's flight time, battery size, and overall performance, as it indicates how power usage scales with throttle settings.

VI. COMPARATIVE ANALYSIS

A comparative analysis can be conducted based on the information provided in this conversation, particularly in the context of solar-powered drones and power consumption. Here's a brief comparative analysis:

Solar-Powered Drones vs. Battery-Powered Drones:

A. ***Battery Life:***

- Battery-Powered Drones: Limited flight duration, typically ranging from 10 to 20 minutes.
- Solar-Powered Drones: Extended flight duration, capable of continuous operation with energy harvested from solar panels.

B. ***Environmental Impact:***

- Battery-Powered Drones: Higher environmental impact due to reliance on traditional batteries.
- Solar-Powered Drones: Reduced environmental impact, lower carbon footprint, and alignment with sustainability goals.

C. ***Operational Versatility:***

- Battery-Powered Drones: Limited adaptability to mission demands and environmental conditions.
- Solar-Powered Drones: Enhanced versatility, ability to adapt to changing conditions and extended mission requirements.

D. ***Continuous Surveillance:***

- Battery-Powered Drones: Require frequent battery changes or recharging, leading to interruptions in surveillance missions.
- Solar-Powered Drones: Capable of continuous surveillance, improving effectiveness in monitoring tasks.

E. ***Energy Management:***

- Battery-Powered Drones: Limited control over energy consumption, dependent on battery capacity.

- Solar-Powered Drones: Energy management systems required for balancing energy harvesting and consumption.

F. ***Cost-Benefit Analysis:***

- Battery-Powered Drones: Lower initial cost but may incur higher operational and maintenance costs due to battery replacement and frequent recharging.
- Solar-Powered Drones: Higher initial investment in solar panel integration but potential long-term cost savings in terms of reduced operational downtime and recharging.

G. ***Geographic Variations:***

- Battery-Powered Drones: Performance consistent regardless of geographic location.
- Solar-Powered Drones: Effectiveness influenced by geographic location, sunlight availability, and seasonal changes.

H. ***Safety and Reliability:***

- Battery-Powered Drones: Standard safety measures but limited in addressing battery-related emergencies.
- Solar-Powered Drones: Require safety features and fail-safes for solar panel integration and battery management.

I. ***Battery Life and Degradation****:*

- Battery-Powered Drones: Battery life and degradation impact operational efficiency.
- Solar-Powered Drones: Battery life still plays a role but is supplemented by solar energy harvesting.

J. ***Mission-Specific Applications:***

- Both types of drones have mission-specific applications, but solar-powered drones offer advantages in missions requiring continuous monitoring.

The comparative analysis highlights the transformative potential of solar-powered drones, particularly in addressing the limitations of battery-powered drones. While there are initial investment costs and considerations related to geographic variations, solar-powered drones offer extended flight duration, lower environmental impact, and increased versatility, making them a valuable asset in various sectors requiring continuous monitoring and sustainable energy solutions.

VII. FINDINGS AND DISCUSSIONS

This project highlights a significant finding in the field of drone technology, which is the incorporation of solar panels directly into drones to address the persistent challenge of limited battery life. The key finding can be summarized as follows:

A. ***Incorporating Solar Panels into Drones Enhances Operational Versatility***: The integration of solar panels represents a groundbreaking solution to overcome the constraint of limited drone battery life. By harnessing solar energy for in-flight charging, drones can achieve uninterrupted operation, regardless of the mission's energy demands. This innovation marks a significant step forward in drone technology.

B. ***Addressing Battery Limitations****:* The limited battery life of drones has been a persistent challenge, restricting their capabilities and operational range. Factors such as drone size, weight, flight mode, and environmental conditions have a significant impact on flight time. The finding acknowledges that this challenge has been a critical hurdle in realizing the full potential of drones.

C. ***Varied Flight Durations****:* The abstract points out that consumer drones typically offer flight durations ranging from 10 to 20 minutes per charge. However, it's important to note that this timeframe can vary considerably due to the nature of the mission and external conditions, such as weather. High-speed flight, aggressive maneuvers, heavy payloads, and adverse weather can dramatically reduce the battery life, further emphasizing the need for a more sustainable solution.

D. ***Application across Sectors***: Drones have found applications across various sectors, including agriculture, forestry, infrastructure inspection, surveillance, and more. Their ability to provide valuable data and services in these sectors is limited by their constrained flight time. The incorporation of solar panels directly into drones offers the potential to unlock new possibilities and extend the scope of applications.

E. ***Continuous Surveillance****:* Surveillance drones, in particular, are mentioned as a use case where limited battery life has posed challenges. These drones are often tasked with covering specific areas, and they must return for battery changes or recharging, leading to interruptions in surveillance. The use of solar panels enables these drones to operate continuously, enhancing their effectiveness in surveillance tasks.

F. ***Environmental Considerations****:* The integration of solar power is not only about improving drone performance but also aligns with environmental considerations. Solar-powered drones have a reduced environmental impact compared to traditional battery-powered counterparts, as they produce fewer greenhouse gas emissions and reduce their carbon footprint.

This finding underscores the potential for solar-powered drones to revolutionize various sectors by offering extended flight durations and increased adaptability, making them a valuable asset for applications that require continuous monitoring and remote operations. The integration of solar power not only addresses the limitations of battery life but also aligns with sustainability and environmental considerations, reducing the carbon footprint associated with drone operations.

VIII. CONCLUSION

The "Design of a Solar-Powered Drone" project has successfully achieved its primary objective of creating an innovative and versatile Unmanned Aerial Vehicle, harnessing the power of renewable energy for extended flight capabilities. The project's conclusion highlights the substantial contributions and advancements made during the design and development process. The effective integration of high-efficiency solar panels, advanced power management systems, and optimized payload capacity has empowered the solar-based drone to operate for extended durations without compromising its functionality. The drone's stable performance, energy efficiency, and reliable data collection validate its potential applications across a spectrum of fields, including environmental monitoring, precision agriculture, disaster response, and surveillance. This project's success in overcoming the limitations of conventional drones, especially in terms of limited flight time, opens up new frontiers for unmanned aerial systems. By making use of renewable energy sources, the solar-based drone offers the promise of sustained and eco-friendly aerial operations, contributing to environmental sustainability and cost-efficiency. The safety and reliability assessments confirm that the solar-based drone is equipped to handle a wide range of operational challenges and emergencies, establishing it as a trustworthy tool for critical missions. Prospects for future advancements in solar panel technology, autonomous features, and enhanced energy storage systems present exciting opportunities for further breakthroughs in this field.

As this project surfaces the way for ongoing research and development in the realm of solar-powered drones, its impact is poised to extend across various industries and applications. The successful completion of this project marks a significant milestone in the advancement of UAV technology, showcasing its potential to contribute to a sustainable and environmentally conscious future.

ACKNOWLEDGMENT

This work is based on the preprint version uploaded to Research Square (https://www.researchsquare.com/), which is a preprint; it has not been peer reviewed or published by any journal.

REFERENCES

[1] Zhang, J., Wang, J., Ni, Y., May, R., & Singer, A. C. (2012). Solar Drone. Retrieved from https://courses.engr.illinois.edu/ece445/getfile.asp?id=2277

[2] Kumar, N. (n.d.). Project report on the solar drone. Retrieved from https://www.academia.edu/42977895/Project_report_on_solar_drone

[3] The Drone Life. (2023, March 23). 7 Valuable Uses For Drones In The Solar Industry. Retrieved from https://thedronelifenj.com/uses-for-drones-in-the-solar-industry/

[4] Merz, M., Pedro, D., Skliros, V., Bergenhem, C., Himanka, M., Houge, T., ... & Hamrén, R. (2022). Autonomous UAS-Based Agriculture Applications: General Overview and Relevant European Case Studies. Drones, 6(4), 128. doi: 10.3390/drones6040128

[5] Sudbury, A. W., & Hutchinson, E. B. (2016). A Cost Analysis of Amazon Prime Air (Drone Delivery). Journal of Economic Education, 16(1), 1-12. doi: 10.1080/15371240903248316

[6] Chu, Y., Ho, C., Lee, Y., & Li, B. (2021). Development of a Solar-Powered Unmanned Aerial Vehicle for Extended Flight Endurance. Drones, 5(2), 44. doi: 10.3390/drones5020044

[7] Safyanu, B., Abdullah, M., & Omar, Z. (2019). Review of Power Device for Solar-Powered Aircraft Applications. J. Aerosp. Technol. Manag., 11, 4119. doi: 10.5028/jatm.v11.1027

[8] Zhang, J., Wang, J., Ni, Y., May, R., & Singer, A. C. (2012). Solar Drone. Retrieved from https://courses.engr.illinois.edu/ece445/getfile.asp?id=2277

[9] Kumar, N. (n.d.). Project report on solar drone. Retrieved from https://www.academia.edu/42977895/Project_report_on_solar_drone

[10] The Drone Girl. (2021, August 25). Solar drones: Everything you need to know. Retrieved from https://thedronegirl.com/2021/08/25/solar-drones-everything-you-need-to-know/

[11] Instructables. (2021, May 14). How to Build a Solar Drone Program. Retrieved from https://www.instructables.com/How-to-Build-a-Solar-Drone-Program/

[12] Arani, M., Abdolmaleki, S., Maleki, M., Momenitabar, M., & Liu, X. (2021). A SimulationOptimization Technique for Service Level Analysis in Conjunction with Reorder

[13] Point Estimation and Lead-Time Consideration: A Case Study in Sea Port. In

[14] H. R. Arabnia, et al. (Eds.), Advances in Parallel & Distributed Processing, and Applications. Transactions on Computational Science and Computational Intelligence (pp.839–858). Cham: Springer. https://doi.org/10.1007/978-3-030-69984-0_61.

[15] Babak, S. J., Hussain, S. A., Karakas, B., & Cetin, S. (2017, July). Control of autonomous ground vehicles: A brief technical review. In , Vol. 224, No. 1. IOP conference series: Materials science and engineering (p. 012029). IOP Publishing.

[16] Bagla, R. K., & Khan, J. (2017). Customers' expectations and satisfaction with online food ordering portals. Prabandhan: Indian. Journal of Management, 10(11), 31–44.

[17] Bashuna, D., Griffin, O., & Iskandarova, S. (2017, June). The role of educational media and Technology in Developing Countries (REMOTE). In EdMedia+ innovate learning(pp. 38-42). Association for the Advancement of Computing in Education (AACE).

[18] Bechtsis, D., Tsolakis, N., Vlachos, D., & Iakovou, E. (2017). Sustainable supply chain management in the digitalization era: The impact of automated guided vehicles. Journal of Cleaner Production, 142, 3970–3984.

[19] Borghetti, F., Caballini, C., Carboni, A., Grossato, G., Maja, R., & Barabino, B. (2022). The use of drones for last-mile delivery: A numerical case study in Milan, Italy. Sustainability, 14(3), 1766. https://doi.org/10.3390/su14031766

[20] Boukoberine, M. N., Zhou, Z., & Benbouzid, M. (2019, October). Power supply architectures for drones-a review. In , Vol. 1. IECON 2019-45th annual conference of the IEEE industrial electronics society (pp. 5826–5831). IEEE.

[21] Boysen, N., Fedtke, S., & Schwerdfeger, S. (2021). Last-mile delivery concepts: a survey from an operational research perspective. OR Spectrum, 43, 1–58. https://doi.org/10.1007/s00291-020-00607-8

[22] Buchal, C. (2020). The challenge of discussing energy facts and climate change. In , Vol. 246. EPJ web of conferences. EDP Sciences.

[23] Chen, H., Jin, Y., & Huo, B. (2020). Understanding logistics and distribution innovations in China. International Journal of Physical Distribution and Logistics Management, 50(3), 313–322.

[24] Chitta, S., & Jain, R. K. (2017). Last Mile Delivery Using Drones. Technology Convergence, innovation & Decision Sciences At: Seoul, South Korea [Online]. Avaliable at:https://www.researchgate.net/publication/321488276 LastMile Delivery Using Drones.

[25] Cohort, P. R. D. (2016). A framework for the implementation of drones in German automotive OEM logistics operations (Doctoral dissertation). University of Portsmouth.

[26] Dekhne, A., Hastings, G., Murnane, J., & Neuhaus, F. (2019). Automation in logistics: Big opportunity, bigger uncertainty. The McKinsey Quarterly, 1–12.

[27] Ndiaye, M., Salhi, S., & Madani, B. (2020). When Green technology meets optimization modeling: The case of routing drones in logistics, agriculture, and healthcare. In Modeling and optimization in Green logistics (pp. 127–145). Springer, Ch.