**Basic Protocol Layer**

**Commands:**

Movement:

These commands control the position and movement of the drone:

1. SET BASE (x, y, z): sets the base (charging station). **// must be implemented**
2. TAKEOFF: Initiates the drone's take-off. **// implemented in simulation control.cpp – must be taken** from there.
3. LAND: Lands the drone. **// must be implemented (we need to implement it also in DroenControl.cpp)**
4. MOVE TO (x, y, z): Moves the drone to a specified point by coordinates. **// must be implemented**
5. HOVER: Keeps the drone hovering in place. **// empty command (for now)**
6. ROTATE (yaw/pitch/roll): Rotates the drone along its axes. **// empty command (for now)**
7. SET VELOCITY (vx, vy, vz): Sets the drone's speed in a specific direction. **// must be implemented**
8. SET ACCELARATION (ax, ay, az): sets the drone/s acceleration in specific direction. **// must be implemented**
9. RETURN TO BASE: Returns the drone to the starting point (or a designated base point). **// must be implemented (we need to implement it also in DroenControl.cpp)**

State Management Commands:

These commands are used to control and monitor the drone's state:

1. STOP: Immediately halts all drone movements. **// must be implemented**
2. CALIBRATE: Calibrates sensors (e.g., gyroscope, compass). **// empty command (for now)**
3. SET MODE (e.g., manual, autopilot): Switches between manual and autopilot modes. **// empty command (for now)**
4. Power On: Activates the motors (preparing for takeoff). **// implemented in simulation control.cpp – must be taken from there.**
5. Power Off: Deactivates the motors. **// implemented in simulation control.cpp – must be taken from there.**

Sensor Data and Monitoring:

Modern drones are equipped with various sensors. These commands enable data collection and processing:

1. STATUS: Requests the drone's current state (battery charge, coordinates, speed, altitude). **// must be implemented**
2. GET POSITION: Requests the current coordinates. **// must be implemented**
3. GET ALTITUDE: Requests the current altitude. **// must be implemented**
4. GET BATTERY STATUS: Checks the battery level. **// must be implemented**
5. GET SENSOR DATA: Retrieves data from various sensors (e.g., camera, LiDAR, ultrasonic sensors, barometer). **// empty command (for now)**

Inter-Drone Communication:

These commands are useful for swarm coordination:

1. BROADCAST (message, protocol): Sends a message to all drones. **// must be implemented**
2. SEND TO (target\_id, message, protocol): Sends a command or message to a specific drone. **// must be implemented**
3. REQUEST NEIGHBOR LIST: Retrieves a list of nearby drones (based on network or physical topology). **// must be implemented**
4. LEAD (target\_id): Transfers leadership to another drone. **// empty command (for now)**

Energy Management Commands:

These commands focus on energy efficiency and safety:

1. POWER SAVE MODE: Switches to a power-saving mode (e.g., reduces sensor update frequency). **// empty command (for now).**

Emergency Commands:

These commands ensure safe operation in critical situations:

1. EMERGENCY LAND: Initiates immediate landing at the nearest safe location. **// empty command (for now).**
2. RETURN TO HOME: Returns the drone to its starting point. **// empty command (for now).**