

Technical Specifications and System Documentation

Kapil Future Tech Drone Configurator

1. System Overview & Requirements

The **KFT Configurator** is a desktop application developed by [Kapil Future Tech Pvt. Ltd.](#) that allows operators to connect to, configure, tune, and maintain MAVLinkcompatible flight controllers from a Windows platform. [\[1\]](#)

Host System Requirements

- **Operating System:** Windows 10 / 11 (x64 Architecture).
- **Processor:** 1.6 GHz minimum (Intel i5 / Ryzen 5 or higher recommended).
- **Memory:** 4 GB RAM minimum (8 GB recommended).
- **Graphics:** Windows 10 integrated graphics (DirectX 11 compatible GPU recommended).
- **Distribution Platform:** Downloadable via the Microsoft Store. [\[1\]](#)

2. Access and Connectivity Specs

Account Protocols

- **Authentication:** Secure user login backed by cloud-based configuration log storage.
- **Registration Rules:** New accounts must be initiated via "Create One" and require explicit administrator approval before service access is granted. [\[1\]](#)

Communication Interface Channels

The system supports three main link interface layers to interact with aircraft flight hardware: **Connection**

Mode	Parameter Type	Configurable Options / Value Ranges
Serial	Port / Baud Rate	Dropdown hardware port selection. Standard speed benchmark is 115200 or 9600 bits per second.
TCP	Network Node IP	Local Host default: 127.0.0.1. Target Mission Planner & listener default ports: 5762 or 5763.
Bluetooth		Automated system device environment scan with active Wireless Pairing selection and verification handshake.

3. Core Software & Hardware Calibration Specs Firmware

Management

- **Internal Channels:** S3 cloud repository pairing ("Upload from Cloud").
- **External ArduPilot Frameworks:** Dynamic toggle sourcing for "Web: Latest" or "Web: Beta" files.
- **Custom Binary Loading:** Compatibility validation structural formatting limited to .apj files or associated native flight controller compilations.

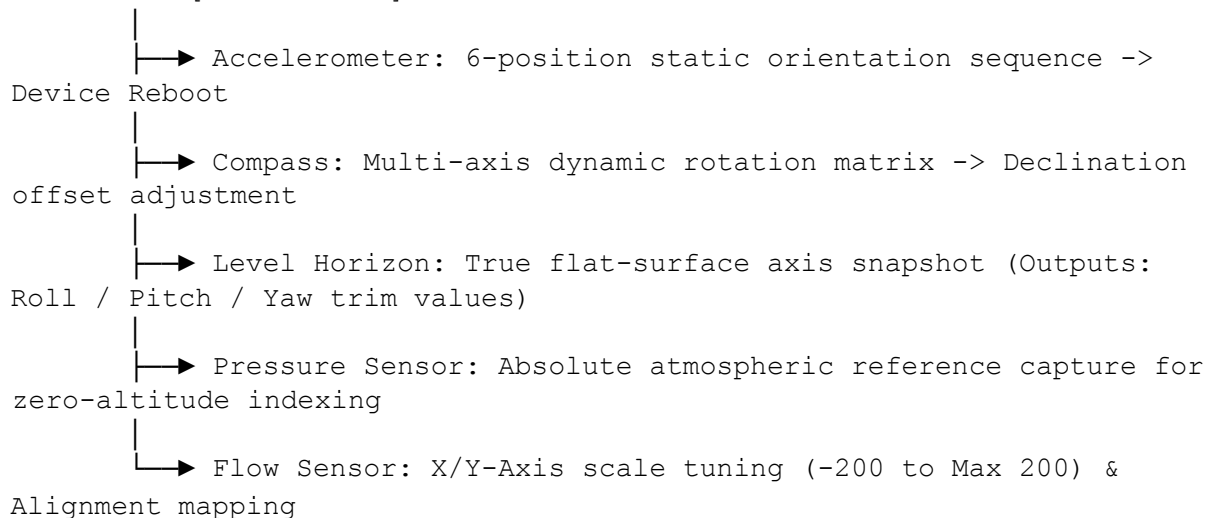
Airframe Kinematics & Matrices

Structure identification requires selection across two specific categories synced with ArduPilot architecture standards:

- **Frame Class:** Options include Quad, Hexa, Octa, OctaQuad, Y6, Tri, BiCopter, HeliQuad, and Single/Dual systems.
- **Frame Type:** Geometrical layout configuration orientation paths (e.g., Plus (+), X, V, H, I, Tail, A-Tail).

Sensor Calibration Thresholds & Variables

To ensure stability, five core sensor ecosystems must be calibrated via the user interface: [SENSORS TAB]



4. Hardware Actuator & Power Mapping

RC Transmitter Mapping

- **Calibration Bounds:** Dynamic channel reading maps active endpoints, midpoints, and flight stick traveling ranges.
- **Flight Mode Channels:** Supports up to 6 custom slots assigned on a selected control channel (typically Channel 5).

Power Subsystem Constraints

- **Sensor Selection:** Configurable choices include 3DR Power Module, Attopilot (45A/90A/180A), CUAHV PM, CubeOrange, Holybro (PM02/PM06/PM07), Mauch (HS/PL series), or "Other".
- **Monitoring Profiles:** Setup options include Analog Voltage Only or combined Analog Voltage and Current tracking.
- **System Value Limits:**
 - *Default Minimum Arming Voltage threshold capability:* Configured from 0.0000V upward.
 - *Standard Voltage Multiplier coefficient metric:* Defaults around 10.10.
 - *Standard AMPS Per Volt calculation index:* Default parameter mapping sits at 17.000.
 - *Amps Offset parameter scale:* Normalized baseline defaults to 0.0.

Motor & ESC Actuation Bounds

- **Interdependency:** Electronic Speed Controller (ESC) endpoint calibration **must** be executed and finalized before running software-based motor tests.
 - **Mapping Variables:** Actuator test sliders dynamically scale to match hardware configuration classes (4 sliders for quadcopters; 6 sliders for hexacopters).
-

5. Security Protocols & Operational Subsystems

Fail-Safes & Arming Interlocks

- **Pre-Flight Arming Checklist Blocks:** Mandatory flag registers include Barometer, Compass, GPS Lock, INS/IMU, Parameters, RC Receiver, Board Voltage, Battery Level, Safety Switch, and Logging Status.
- **GeoFence Boundaries:** Circle, Polygon, or Max Altitude fences can be applied. Breach triggers prompt automated Return-to-Launch (RTL) or forced landing maneuvers.
- **Battery Failsafe Boundaries:** Low Battery Action (None, Land, RTL, Smart RTL) defaults to a 10.5V trigger, while Critical Action recommends an immediate Land command.

Hardware Payload Modules

Camera & Gimbal Interface

- **Stabilization Axis Profiles:** 2-axis options provide pitch and roll control; 3axis systems add yaw tracking.
- **Gimbal Angular Movement:** Range limits map from a minimum of -90° up to a maximum of 0° vertical tilt window.
- **Servo PWM Constraints:** Valid control signals must stay strictly within **500 to 2200 μ s** limits.
- **Camera Trigger Control Rules:** Supports Servo or Relay configurations across fixed physical distance windows (0 to 1000m) or duration parameters (0 to 50 deciseconds).

Agricultural Spraying System Configurations Designed for [precision farming applications](#): [1]

- **RC Dynamic Switches:** Configurable logic trigger parameters range from 0 to a maximum ceiling index value of 102.
- **BRD PWM Count Limit Matrix:** Controls available Flight Management Unit (FMU) outputs (Values: 0 to 8 total PWM lines). Unassigned channels fallback to standard General Purpose Input/Output (GPIO) functionality.
- **Aux Output Sub-channels:** `SERVO9` (Aux 1) and `SERVO10` (Aux 2) parameters:
 - *Minimum Output Bounds:* Lower limit threshold starts at 500 μ s (Neutral baseline sits at 1500 μ s).
 - *Maximum Output Bounds:* Upper limit threshold tops at 2000 μ s (Lower baseline safety bounds stop at 800 μ s).

Parameter Lifecycle Options

- **Import Processing:** Advanced calibration variables are pulled directly from local `.param` **configuration files** via system lookup paths.
 - **Export Profiles:** System records, tuning coefficients, and modifications can be exported into five structured file formats: `.param`, `CSV`, `.cfg`, `.json`, and `.yaml`.
-