AI-Kit Edge Controller

Telemetry Integration & Cloud Communication Specification

Version 1.1

Altius Surya Tech Pvt Ltd

Document Control

Item	Details
Document Title	AI-Kit Edge Controller Telemetry Integration & Cloud Communication Specification
Document Version	1.1
Company	Altius Surya Tech Pvt Ltd
Project	AI-Kit Edge Controller Platform
Status	Approved for Firmware Development
Related Document	AI-Kit Register Mapping - INVT v1.0
Date	01/07/26

Revision History

Version	Date	Description	Status
1.0	22/06/26	Initial telemetry integration and cloud communication specification.	Approved
1.1	01/07/26	Added document control, revision history, VFD configuration and abstraction, firmware architecture, local configuration requirements, updated telemetry dictionary, and reference to separate INVT register mapping document. Existing Version 1.0 content preserved unless consistency required minor clarification.	Approved for Firmware Development

1. Introduction

The AI-Kit Edge Controller is an intelligent field data acquisition and analytics platform developed by Altius Surya Tech Pvt Ltd for renewable energy, water pumping, and industrial monitoring applications.

This document defines the firmware requirements, telemetry structure, communication protocol, and cloud integration requirements for the first production batch of AI-Kit controller PCBs.

The AI-Kit Cloud backend has been deployed, tested, and validated on the production Google Cloud environment and is ready to receive telemetry from field devices.

2. Project Scope

Included in Current Release

The firmware shall support:

- Acquisition of telemetry from sensors and controllers.
- Data collection and local storage.
- Secure cloud communication.
- Automatic retry during communication failures.
- Telemetry upload to AI-Kit Cloud.
- Future compatibility with AI analytics and dashboarding.
- VFD telemetry abstraction through configurable VFD driver selection.

Excluded from Current Release

The following functions are NOT part of this release:

- Remote VFD write operations.
- Remote start/stop commands.
- Remote parameter modification.
- OTA firmware upgrades.
- Cloud-to-device command execution.

These functions may be enabled in future product releases after validation and safety testing.

3. Recommended Hardware Architecture

Recommended architecture:

- ESP32 Controller
- RS485 Modbus Interface
- GSM / 4G Communication Module
- Local Flash Storage
- RTC Timekeeping
- Watchdog Protection

Alternative controller platforms may be proposed subject to approval.

4. Device Identification

The first production batch consists of five devices.

Device	Device ID
PCB-1	AST-PCB-001
PCB-2	AST-PCB-002
PCB-3	AST-PCB-003
PCB-4	AST-PCB-004
PCB-5	AST-PCB-005

Each device shall use:

- Unique Device ID
- Unique API Key

API keys will be supplied separately and shall not be shared between devices.

4A. Device Credential Usage

Each AI-Kit Edge Controller shall be assigned:

- A unique Device ID
- A unique API Key

The Device ID and API Key together identify and authenticate the device with the AI-Kit Cloud Platform.

Purpose

The API Key acts as a device-specific authentication credential and must be included in every telemetry upload request.

The cloud platform validates:

1. Device ID
2. API Key

before accepting telemetry data.

Request Header Example

```
Content-Type: application/json
X-API-Key: <Assigned Device API Key>
```

Firmware Storage Requirements

The following values shall be stored in non-volatile memory:

- Device ID

- API Key
- Firmware Version

These values should be configurable without requiring firmware source code modifications.

Preferred storage methods:

- ESP32 NVS
- EEPROM
- Flash Configuration Area

Security Requirements

- Each device shall use only its assigned API Key.
- API Keys shall not be shared between devices.
- API Keys shall not be printed in diagnostic logs.
- API Keys shall not be exposed in user interfaces.
- API Keys shall not be committed to source code repositories.

Future Serviceability

The firmware architecture shall allow Device ID, API Key, and Firmware Version to be updated independently of the main application firmware.

This requirement will simplify future credential rotation, field replacement, and maintenance activities.

4B. VFD Configuration & Abstraction

The firmware shall implement a VFD abstraction layer so that the same AI-Kit cloud telemetry payload can be generated from different VFD brands or controller platforms without changing the cloud API contract.

Supported VFD Type for Version 1.1

Configuration Key	Value	Purpose
VFD_TYPE	INVT	Enables the INVT Modbus register driver and scaling rules.
VFD_TYPE	FUJI	Reserved for Fuji VFD driver support where applicable.
VFD_TYPE	GENERIC_MODBUS	Reserved fallback for standard Modbus telemetry reads.

Driver Abstraction Requirement

- The firmware shall isolate brand-specific register addresses, scaling, and status decoding inside a VFD driver module.
- The cloud telemetry payload shall remain stable even if the underlying VFD register map changes.

- The selected driver shall expose normalized values such as frequency_hz, rpm, power_pct, power_kw, run_status, and fault_code.
- Write registers shall remain disabled in this release unless separately approved after validation and safety testing.

Register Mapping Reference

The INVT register addresses, scaling rules, status decoding, and reserved write registers are defined in the separate document: AI-Kit Register Mapping - INVT v1.0. Firmware teams shall use that document as the implementation reference for VFD_TYPE = INVT.

Firmware Architecture

Layer	Responsibility
Hardware Interface Layer	RS485 transceiver, UART configuration, GPIO control, watchdog, and power-related protections.
Modbus Driver Layer	Read holding registers, handle retries/timeouts, validate response lengths, and report communication errors.
VFD Abstraction Layer	Apply VFD_TYPE-specific register mapping, scaling, and status/fault decoding.
Telemetry Builder	Construct normalized JSON payload using AI-Kit telemetry field names.
Local Buffer	Queue telemetry records during outages using non-volatile circular storage.
Cloud Client	Upload telemetry over HTTPS with Device ID and X-API-Key authentication.
Configuration Manager	Load Device ID, API Key, Firmware Version, VFD_TYPE, RS485 parameters, and timing parameters from non-volatile memory.

Local Configuration

The following configuration values shall be stored locally and shall be editable without source code modification:

Parameter	Example / Default	Notes
DEVICE_ID	AST-PCB-001	Unique per controller PCB.
API_KEY	Supplied separately	Must not be logged or committed to source code.
FIRMWARE_VERSION	1.1.0	Reported in telemetry payload.
VFD_TYPE	INVT	Selects the VFD driver and register map.
MODBUS_SLAVE_ID	1	Field configurable based on VFD setting.
MODBUS_BAUDRATE	19200	Recommended default for INVT integration; field configurable.
MODBUS_PARITY	E	Even parity unless site configuration differs.
MODBUS_STOP_BITS	1	Recommended default.
SAMPLING_INTERVAL_SEC	5	Telemetry sampling interval.
UPLOAD_INTERVAL_SEC	30	Cloud upload interval.
API_ENDPOINT	https://ai.altiussuryatech.in/api/telemetry/ingest	Production cloud endpoint.

5. Data Acquisition Requirements

Recommended sampling interval:

5 seconds

Recommended cloud upload interval:

30 seconds

The firmware shall continue collecting telemetry during communication outages.

Buffered data shall be uploaded automatically when connectivity is restored.

6. Local Data Buffering

Minimum storage requirement:

1000 telemetry records

Preferred implementation:

Circular buffer in non-volatile memory

Telemetry shall survive temporary communication failures and normal power interruptions wherever possible.

7. Cloud Communication

Production Endpoint

https://ai.altiussuryatech.in/api/telemetry/ingest

Method

POST

Protocol

HTTPS

TLS 1.2 or higher

Request Headers

```
Content-Type: application/json
X-API-Key: <Assigned Device API Key>
```

8. Telemetry Payload Format

Example:

```
{
  "device_id": "AST-PCB-001",
  "timestamp": "2026-01-01T12:00:00Z",
  "firmware_version": "1.1.0",
  "vfd_type": "INVT",
  "pv_voltage_v": 580.2,
  "motor_voltage_v": 415.0,
  "motor_current_a": 8.2,
  "power_pct": 51.0,
  "power_kw": 5.1,
  "frequency_hz": 45.0,
  "rpm": 1350,
  "flow_lpm": 370,
  "run_status": 1,
  "fault_code": 0,
  "signal_strength": 64,
  "controller_temp_c": 60.3
}
```

9. Telemetry Field Definitions

Mandatory Fields

Field	Type	Description
device_id	String	Unique AI-Kit Device Identifier
timestamp	ISO8601 UTC	Telemetry Timestamp
run_status	Integer	Pump Running Status

Recommended Fields

Field	Type	Description
firmware_version	String	Firmware Version
vfd_type	String	Configured VFD driver type, for example INVT.
pv_voltage_v	Float	Solar PV Voltage
motor_voltage_v	Float	Motor Output Voltage
motor_current_a	Float	Motor Output Current
power_pct	Float	VFD reported output power as percentage of rated power, after scaling.
power_kw	Float	Motor Power in kW. Where the VFD provides only power percentage, firmware may derive kW using configured rated motor power.
frequency_hz	Float	Operating Frequency
rpm	Float	Motor Speed
flow_lpm	Float	Water Flow Rate
fault_code	Integer/String	Controller Fault Code
signal_strength	Integer	GSM/WiFi Signal Strength
controller_temp_c	Float	Controller Temperature

10. Cloud Responses

Success

```
HTTP 200
{
  "ok": true
}
```

Authentication Failure

```
HTTP 403
{
  "detail": "invalid device or api key"
}
```

Invalid Request

```
HTTP 400
{
  "detail": "missing device_id"
}
```

Missing API Key

```
HTTP 401
{
  "detail": "missing api key"
}
```

11. Reliability Requirements

The firmware shall:

- Retry failed uploads automatically.
- Queue telemetry during communication outages.
- Upload queued data when communication is restored.
- Maintain timestamp accuracy using RTC or NTP.
- Avoid duplicate uploads wherever practical.
- Recover automatically after power interruptions.

12. Security Requirements

The firmware shall:

- Use HTTPS for all communications.
- Store Device ID in non-volatile memory.
- Store API Key securely in non-volatile memory.
- Avoid exposing credentials in diagnostic logs.
- Prevent unauthorized cloud access.

13. Firmware Team Deliverables

The firmware development team shall provide:

3. Firmware architecture description.
4. Communication architecture.
5. Controller platform details.
6. Modbus register mapping.
7. Local storage implementation details.
8. Successful cloud upload test results.
9. Estimated completion timeline.
10. VFD abstraction implementation details, including selected VFD_TYPE.
11. Confirmation that INVT implementation follows AI-Kit Register Mapping - INVT v1.0.

14. Information Required from Firmware Team

Please confirm:

12. Controller/MCU selected.
13. Communication method (WiFi / GSM / Both).
14. Modbus implementation details.
15. Local storage capacity available.
16. Telemetry buffering strategy.
17. Expected firmware completion date.
18. Configured VFD_TYPE for each field device.
19. Rated motor power value used for power_kw derivation where applicable.

15. Contact

K N Vishwanath

Altius Surya Tech Pvt Ltd

Project: AI-Kit Edge Controller Platform

Purpose: Cloud-connected intelligent monitoring and analytics platform for renewable energy and industrial automation applications.

Document Version: 1.1

Status: Approved for Firmware Development