5Tech Edge Bridge (ESP32) — Phase-1 MVP

Datasheet · `esp32-edge-bridge-datasheet` · Revision Draft (Preliminary) · 2026-07-08

PRELIMINARY. This datasheet describes a Phase-1 MVP built on an off-the-shelf ESP32 development board. It is a **monitoring and automation prototype**, not a finished or certified product. **Firmware-verified** constants (pins, BLE UUIDs, MQTT topics, timing, thresholds) are taken directly from the shipping firmware and are stated as-is. Values marked ‡ are **proposed**, ESP32-WROOM-32-class engineering placeholders, given with the condition under which they must be confirmed — they are **not measured, not final**, and **no ‡ value is a certification claim**. Values marked `[TBD]` remain pending. Certifications are **n/a — development prototype**. See §12 for change control.

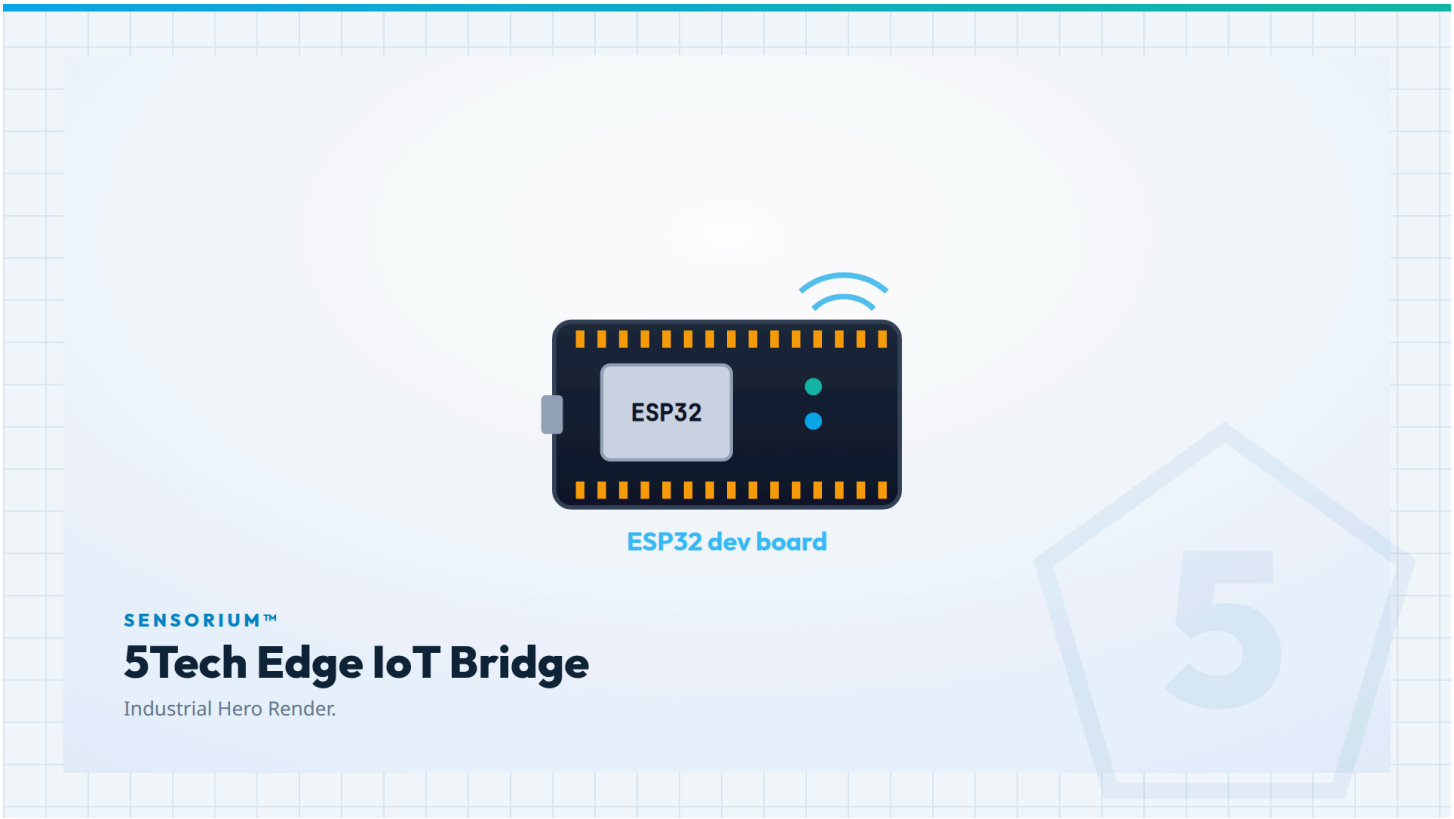

Figure FIG-001 — 5Tech Edge Bridge, the Phase-1 MVP on an ESP32 development board.

1. At a glance

The **5Tech Edge Bridge** is the Phase-1 MVP of the 5Tech stack: firmware on a stock ESP32 development board that bridges sensors, machine signals, and test events to automation systems over Wi-Fi, Bluetooth LE, and MQTT. Phase 1 validates the complete sensor-to-automation pipeline so that Phase 2 can convert a *proven* design into a custom PCB.

Product	5Tech Edge Bridge (ESP32) — Phase-1 MVP
Family / role	5Tech · **Bridge / Prototype** (Phase-1 MVP)
Function	Bridge sensors / machine signals / test events to automation over Wi-Fi, Bluetooth LE, and MQTT; validate the sensor-to-automation pipeline
Hardware platform	Off-the-shelf **ESP32-WROOM-32** development board (`board = esp32dev`)
Connectivity	Wi-Fi 2.4 GHz (station), Bluetooth LE (GATT server), MQTT over TCP
Power	USB 5 V (development bench)
Firmware	v0.1.0; two variants (ESP-IDF C / Arduino C++) sharing one wire contract
Certifications	n/a — development prototype

Key features

- Full sensor-to-automation pipeline on a stock ESP32 dev board — no custom hardware required.
- Three transports: Wi-Fi + MQTT, Bluetooth LE (works with no network), and a serial console.
- Structured JSON: telemetry every 5 s, a 30 s status heartbeat, immediate event publish, retained status + Last Will.
- Nine-state device model with a latching WARNING / DANGER hazard model (explicit clear only).
- Three interchangeable input sources: real GPIO, onboard button, and a simulated sensor.
- Two firmware variants speaking an identical, parity-checked wire protocol.

Typical applications

- Validating the MQTT + Bluetooth LE pipeline end to end before committing to a PCB.
- Client / investor demonstrations of the sensor-to-automation path.

- Bench proving of automation integration (n8n / Node-RED / custom).
- The reference design and firmware base that Phase-2 production hardware is built from.

---

2. System context

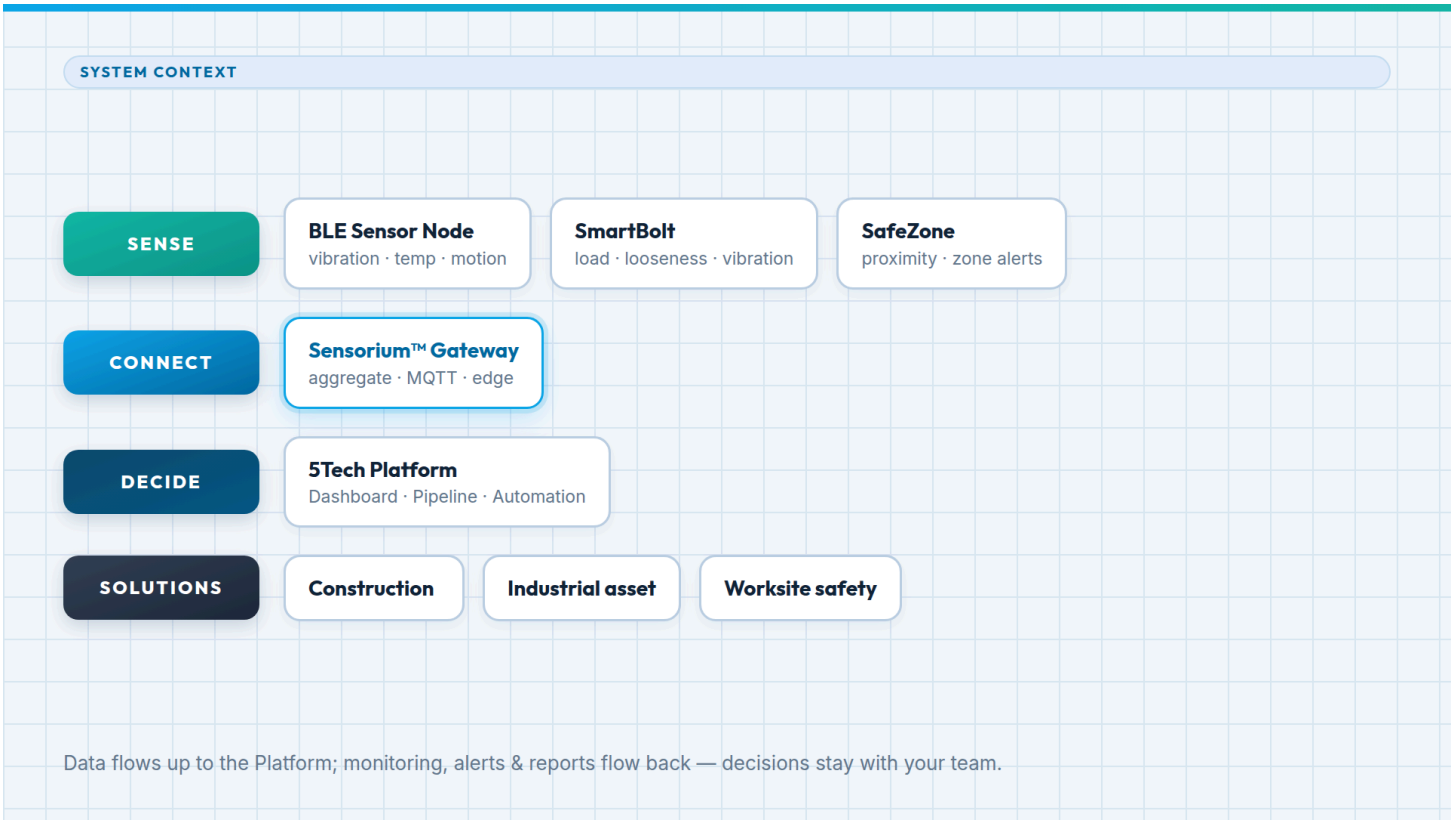

Figure FIG-002 — Sense → Connect → Decide. The Edge Bridge is a single-device edge prototype: it acquires one device's signal and publishes that device's own state upward.

The Edge Bridge is a **BLE peripheral (a GATT server) that publishes its own single device's state**. It acquires a signal, resolves it to a state, exposes that state over Bluetooth LE, and publishes it over MQTT. Architecturally it belongs on the **Sense** side, as an edge acquisition-and-publish node — it is **not** a central aggregator of other nodes.

- **Sense** — a real GPIO sensor, the onboard button, a simulated sensor, or a remote command (BLE / MQTT / serial) indicates a condition **on this one device**.
- **Connect (transport)** — the ESP32 resolves its state, builds a JSON payload, and publishes it over Wi-Fi to an MQTT broker; a Bluetooth LE **GATT server** exposes this device's own characteristics (Det1, Det2, Con1, Cmd, Resp) for local control that survives a dead network.
- **Decide** — an automation flow (n8n / Node-RED / custom) subscribes, filters, and routes the message to dashboards, alerts, and logs.

Product boundary vs. the Sensorium™ Gateway. This prototype is a BLE **peripheral** serving a single device's own state; it does **not** aggregate other BLE nodes. The central-aggregation *Connect* role — a BLE **central** that collects many Sensorium™ nodes and forwards them upstream — belongs to the **Sensorium™ Gateway**, a separate product. What the Edge Bridge validates is the **edge acquisition and MQTT publishing patterns** (device identity, the state model, JSON payloads, retained status + Last Will, and command handling) — not the Gateway's aggregation logic.

3. Functional overview

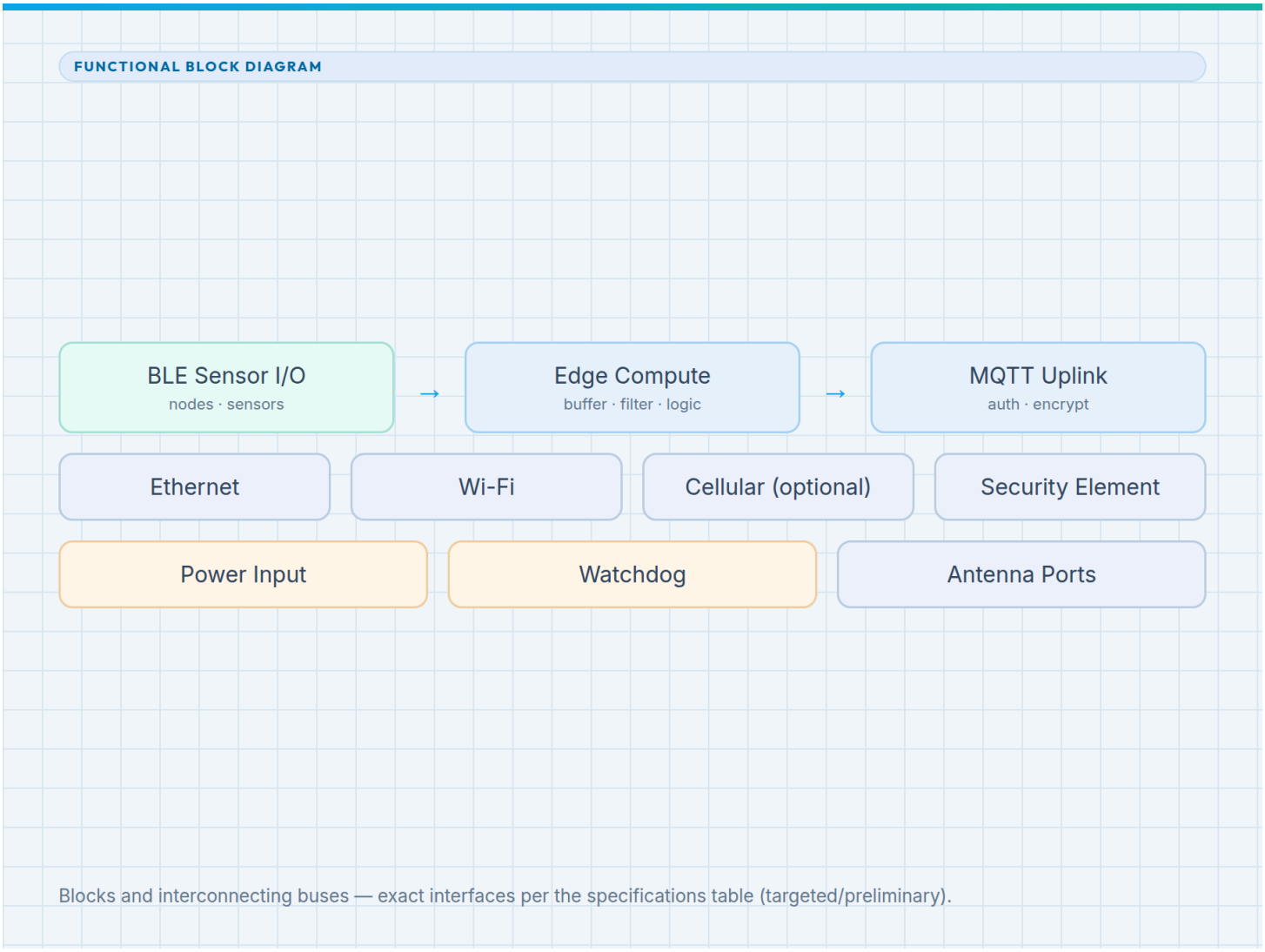

Figure FIG-003 — Functional blocks: input sources, the shared decision core, three transports, and local I/O.

The firmware is organised into four functional groups:

1. **Input sources** — Det1 (GPIO18) and Det2 (GPIO19) digital inputs, the onboard button (GPIO0), and a simulated sensor (a bounded random walk), plus remote commands over any transport. Any source can feed the pipeline; none is required to run the demo.
2. **Shared decision core** — an event detector (`NORMAL < WARNING < DANGER`, latching) and a nine-state device model resolved in strict priority. This core is compiled **verbatim into both firmware variants**, so their logic cannot drift.
3. **Communication** — a Wi-Fi station + MQTT client, a Bluetooth LE GATT server, and a serial console. Inbound commands from all three are queued and executed on the main loop.
4. **Local I/O** — a status LED with a per-state blink pattern (GPIO2), an optional buzzer (GPIO25, disabled by default), and the Con1 contact output (GPIO23).

Data path: *input → event detector → device state → JSON payload → MQTT / BLE / serial → automation*. Commands travel the reverse path and are enqueued, never run on a radio callback task, so a QoS 1 reply cannot deadlock against the task that must acknowledge it.

---

4. Ordering information

The Edge Bridge is a **Phase-1 MVP**, not a commercial product; there are no order codes. It is assembled from standard parts:

ITEM	SPECIFICATION	NOTES
ESP32 dev board	Any **ESP32-WROOM-32** devkit (`board = esp32dev`)	The only mandatory hardware
USB cable	Data-capable (not charge-only)	Power, flashing, and serial console
Button *(optional)*	GPIO0	The onboard BOOT button works
Sensors *(optional)*	GPIO18 (Det1), GPIO19 (Det2), active-low	Any dry-contact / digital source
Buzzer *(optional)*	GPIO25	Disabled by default (`ENABLE_BUZZER = 0`)

Phase-2 production hardware (custom PCB + enclosure) order codes are `[TBD]` and are not committed until Phase 1 freezes the sensor, connector, power, and enclosure requirements.

---

5. Mechanical specifications

There is **no custom enclosure or mechanical design in Phase 1** — the MVP is a bare development board. A custom PCB and enclosure are Phase-2 work.

PARAMETER	VALUE	CONDITION / NOTE
Form factor	ESP32-WROOM-32 development board (bare PCB)	No 5Tech enclosure in Phase 1
Dimensions (W × H × D)	`[TBD]`	Board-dependent; varies by the specific devkit used
Weight	`[TBD]`	Board-dependent
Mounting	None / breadboard / bench	No mounting provision in Phase 1
Enclosure	None	Custom enclosure is Phase 2

6. Electrical specifications

Preliminary. Firmware-verified behaviour (logic levels, the active-low sensor assumption, the Con1 pin) is taken directly from the shipping firmware and carries no mark. Values marked **‡** are proposed ESP32-WROOM-32-class figures, stated with their condition and **not yet measured**; they are confirmed by measurement during Phase 1. `[TBD]` marks a value still pending.

PARAMETER	MIN	TYP	MAX	UNIT	CONDITION
Supply (USB VBUS)	4.75	5.0	5.25	V DC	Bench power over USB (USB 5 V ±5 %)
Module logic rail		3.3		V DC	On-board regulator
Supply current	—	120‡	500‡	mA	**Unmeasured.** @ 5 V; typ‡ = Wi-Fi associated, idle loop; peak‡ = momentary Wi-Fi TX burst. ESP32-WROOM-32-class estimate, to be confirmed by measurement
GPIO logic level		3.3		V	Inputs are **not** 5 V tolerant
Det1 / Det2 input high, V_IH	2.48‡	—	—	V	Logic-high threshold ≈ 0.75 × 3.3 V (ESP32-WROOM-32-class); the firmware treats a Det input as **active-low** with internal pull-ups
Det1 / Det2 input low, V_IL	—	—	0.83‡	V	Logic-low threshold ≈ 0.25 × 3.3 V (ESP32-WROOM-32-class)
Det1 / Det2 sense polarity	—	—	—	—	**Active-low** (`DET_INPUT_ACTIVE_LOW = 1`), internal pull-ups enabled (firmware-verified); a floating or open input reads **de-asserted**
Contact output (Con1) drive	—	20‡	40‡	mA	GPIO23, 3.3 V logic-level output; typ‡ recommended continuous, max‡ absolute per-pin (ESP32-WROOM-32-class). See the Con1 NOTICE below

NOTICE. GPIO pins are 3.3 V logic and are not 5 V tolerant. Do not connect a sensor or signal that exceeds 3.3 V to a GPIO input without appropriate level conditioning.

NOTICE — Con1 is a logic-level pin, not a relay driver. Con1 (GPIO23) is a bare 3.3 V GPIO output. To switch a relay, contactor, or any inductive or mains load it needs an **external driver** (a transistor / MOSFET stage or an opto-isolated relay module), a **flyback / free- wheeling diode** across any inductive load, and appropriate **electrical isolation**. Do not switch such a load directly from the pin.

NOTICE — No input protection (non-industrial). Power is a **bare USB 5 V** input with **no reverse-polarity, surge, or ESD protection** on the assembly, and the GPIO / Det inputs carry no added surge or ESD protection either. This is a bench development board, not an industrial-grade input stage; provide external protection if the deployment environment demands it.

7. Interfaces & protocols

INTERFACE	SPECIFICATION	CONNECTOR / PIN	NOTES
Wi-Fi	2.4 GHz 802.11 b/g/n, station mode	PCB / module antenna	Non-blocking connect with exponential backoff
Bluetooth LE	GATT server; service `4fafc201-1fb5-459e-8fcc-c5c9c331914b`	Shared 2.4 GHz antenna	Advertised as `5Tech-IoT-Bridge-<mac6>`; open GATT, no pairing
MQTT	TCP, plaintext; broker host + port **configurable** (template default `1883`); root `5tech/iotbridge`	over Wi-Fi	telemetry / event / status / command / debug topics; QoS per topic
Serial console	USB UART, 115 200 baud	USB	Same command grammar as BLE and MQTT
Det1	Digital input, active-low, pull-up	GPIO18	Asserted → WARNING; exposed as a BLE characteristic
Det2	Digital input, active-low, pull-up	GPIO19	Asserted → DANGER; exposed as a BLE characteristic
Con1	Contact output, remotely settable	GPIO23	Settable over BLE / MQTT / serial
Status LED	Per-state blink pattern	GPIO2	Onboard LED on most ESP32 devkits
Button	Input, active-low, pull-up	GPIO0	Onboard BOOT button; short = warning/clear, long ≥1.5 s = danger
Buzzer *(optional)*	Output	GPIO25	`ENABLE_BUZZER = 0` by default

BLE GATT characteristics (all under the service UUID above):

CHARACTERISTIC	UUID	PROPERTIES
Det1	`beb5483e-36e1-4688-b7f5-ea07361b26a8`	Read, Notify
Det2	`beb5483f-36e1-4688-b7f5-ea07361b26a9`	Read, Notify
Con1	`beb54840-36e1-4688-b7f5-ea07361b26aa`	Read, Write
Cmd	`beb54841-36e1-4688-b7f5-ea07361b26ab`	Write
Resp	`beb54842-36e1-4688-b7f5-ea07361b26ac`	Read, Notify

MQTT topics (root `5tech/iotbridge`, `<id>` = `5tech-bridge-<mac6>`):

TOPIC	DIRECTION	QOS (ESP-IDF)	RETAIN
`.../<id>/telemetry`	published, every 5 s	0	no
`.../<id>/event`	published, immediately on change	1	no
`.../<id>/status`	published on change **and every 30 s** (heartbeat); retained; also the Last Will	1	yes
`.../<id>/command`	subscribed	1 requested	—
`.../<id>/debug`	published	0	no
`.../all/command`	subscribed (broadcast)	1 requested	—

Publish cadence (firmware-verified). Telemetry publishes every **5 s** (`TELEMETRY_INTERVAL_MS = 5000`); status republishes on every state change **and on a 30 s heartbeat** (`STATUS_INTERVAL_MS = 30000`), so a late subscriber always sees a live retained status; events publish immediately on change. The MQTT keep-alive is **30 s** (`MQTT_KEEPALIVE_S = 30`).

> The Arduino / PlatformIO variant publishes every topic at **QoS 0** (its MQTT client has no QoS parameter); its subscriptions and Last Will are still registered at QoS 1. All topics, payload keys, and key order are otherwise byte-identical between the two variants and are parity-checked in the build.

---

8. Environmental & ratings

This is a **bare development board with no environmental sealing or rating**. Operate it on a bench in a dry, indoor, room-temperature environment.

PARAMETER	VALUE	NOTE
Ingress protection	None	Bare PCB; no enclosure in Phase 1
Operating temperature	0 °C to +40 °C‡ (assembly)	Bench / indoor ‡ estimate for the **assembled** board; deliberately **narrower** than the ESP32-WROOM-32 module's own −40 °C to +85 °C rating, which the devkit's regulator, USB, and passives do not inherit. Confirm by test
Storage temperature	−40 °C to +85 °C‡	ESP32-WROOM-32-class; board-dependent
Humidity	`[TBD]`	Bench use, non-condensing
Vibration / shock	n/a	Prototype; not rated

9. Compliance & standards

n/a — development prototype. This device holds **no product certifications**, and none are claimed. The ESP32-WROOM-32 module itself carries its own **modular FCC / ISED (IC) / CE-RED** radio approval, but that module-level approval does **not** transfer to the assembled Phase-1 prototype, which is **not** a certified product and must not be represented as one. Certification (radio/telecom, EMC, safety, environmental) is Phase-2 work, once there is a production board to certify.

AREA	STATUS
Radio / telecom	n/a — development prototype (no product-level approval). The ESP32-WROOM-32 module carries its own **modular FCC / ISED (IC) / CE-RED** radio approval; that module approval does **not** transfer certification to the assembled prototype
EMC	n/a — development prototype
Safety	n/a — development prototype; **not a safety-rated device**
Environmental (RoHS/REACH)	n/a — development prototype

> **This is not a safety device.** It carries no safety certification of any kind and must not be relied on to protect people or equipment.

---

10. Accessories & related products

ITEM	ORDER CODE	PURPOSE
USB data cable	`[TBD]`	Power, flashing, serial console
Digital sensor / dry contact	`[TBD]`	Optional Det1 / Det2 input
Buzzer	`[TBD]`	Optional local annunciation (GPIO25)

Related documents

- **Manual:** `manuals/esp32-edge-bridge-manual/` — setup, flashing, configuration, demo flow.
- **Product overview:** `products/esp32-edge-bridge/` — positioning and highlights.
- **Firmware & full engineering docs:** `modules/5tech-edge-iot-bridge/` — architecture, MQTT API, Bluetooth API, Phase-1 scope, and the Phase-2 PCB roadmap.
- **Related hardware:** the **Sensorium™ Gateway** — a separate product that performs the BLE central-aggregation *Connect* role (collecting many Sensorium™ nodes upstream). The Edge Bridge does **not** implement or target that role; it validates edge acquisition and MQTT publishing patterns on a single device (see §2).

---

11. Signal & alert behaviour

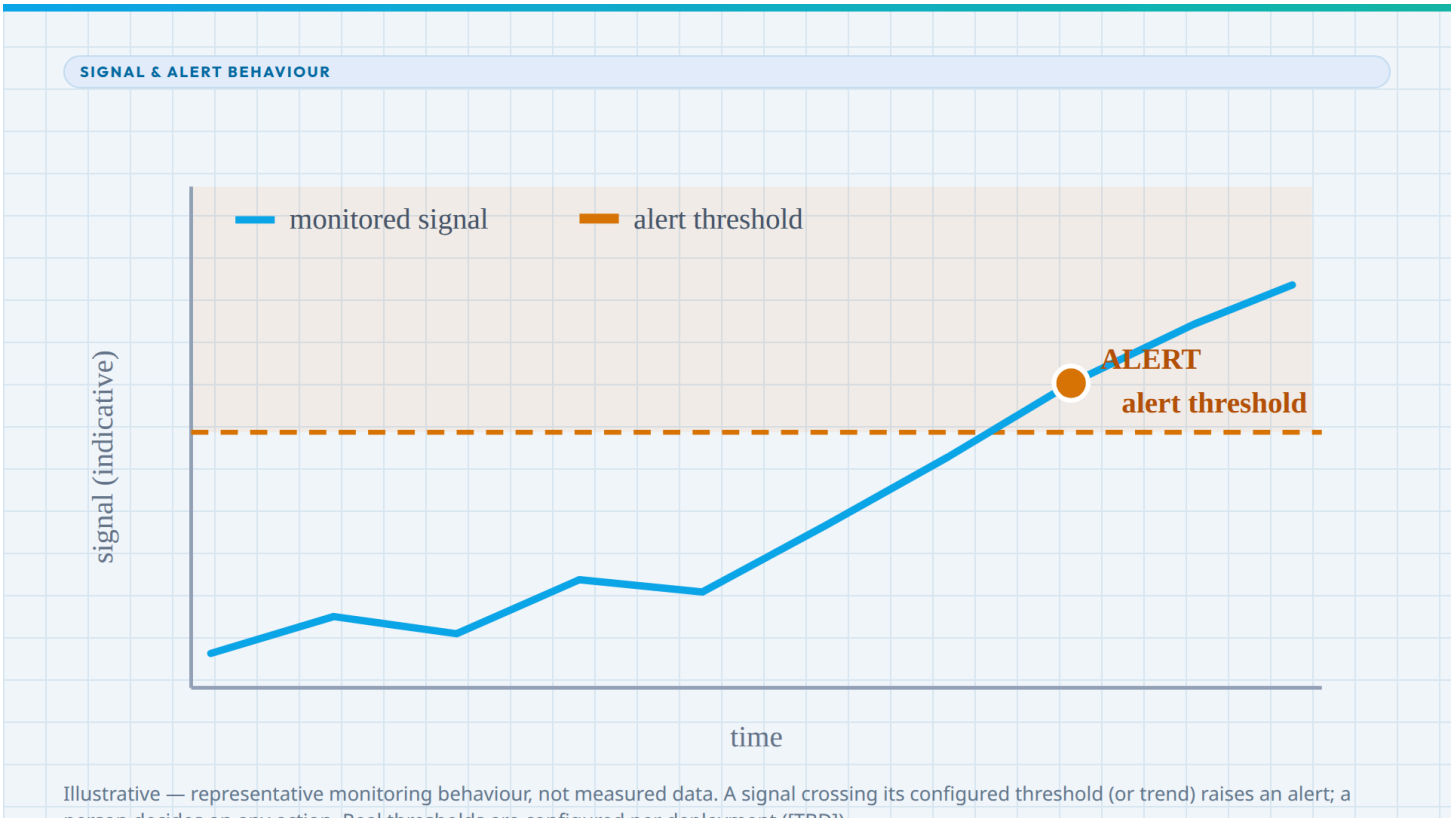

Figure FIG-004 — Illustrative — a monitored signal crossing its configured alert threshold (or an abnormal trend) raises an alert; a person decides on any action. Representative behaviour, not measured data.

The Edge Bridge reports a monitored signal and evaluates a simple rule; when an input crosses its configured threshold (a DANGER state) or shows an abnormal trend, it raises an **alert** and a person decides on any action. It is a prototype validating this path — it takes **no autonomous action** and is **not a safety device**. The chart is illustrative; thresholds are configured per deployment (`[TBD]`).

Example monitored signals & alert triggers

SIGNAL	EXAMPLE ALERT TRIGGER	RESPONSE
Sensor event / input (Det1 / Det2)	A monitored input crosses its configured threshold	Alert + notify (person decides)
DANGER state	The decision core evaluates its rule to a DANGER state	Alert + notify (person decides)
Connectivity / transport	An uplink transport fails or the device stops reporting	Alert + notify (person decides)

Illustrative — representative, not measured; values `[TBD]`.

12. Revision & change control

REVISION	DATE	CHANGE	AUTHOR
Draft (Preliminary)	2026-07-08	Initial authored content, grounded in `modules/5tech-edge-iot-bridge`; unknown values `[TBD]`.	Documentation
Draft (Preliminary)	2026-07-18	Clarified the BLE-peripheral / single-device product boundary vs. the Sensorium™ Gateway; added Det1/Det2 input thresholds, the Con1 drive rating, and the no-input-protection limitation; marked the unmeasured supply current and the assembly temperature range ‡; documented the 30 s status heartbeat and the configurable MQTT port; renamed FIG-001. **No firmware-verified constant was changed.**	Documentation

Full change history: `revision.md`. Sources and cited standards: `references.md`.

Figures

ID	TITLE	STATUS	FILENAME	PURPOSE
FIG-001	Edge Bridge — Prototype Hero Render	Missing	`renders/hero_edge-bridge.png`	Cover / hero image
FIG-002	System Context	Missing	`diagrams/system_context.png`	Where the product sits in the Sense → Connect → Decide chain
FIG-003	Functional Block Diagram	Missing	`diagrams/block_diagram.png`	Internal functional architecture
FIG-004	Signal & Alert Behaviour (illustrative)	Missing	`diagrams/signal_threshold_chart.png`	How a monitored signal crossing a threshold raises an alert (illustrative, not measured data)

Specifications are preliminary and subject to change. See `revision.md` for change control, `references.md` for sources, and `image_prompts.md` for the figure/prompt index.