

SmartStall_API



SmartStall Bluetooth Peripheral API

This document describes the public-facing **Bluetooth GATT API** for the SmartStall touchless locking system. It is intended for developers building Bluetooth central clients (e.g., ESP32-based hubs) that connect to SmartStall peripherals.

The SmartStall device is a Bluetooth Low Energy (BLE) peripheral. It advertises a custom GATT service that exposes:

- BLE device name **SmartStall**
- Stall lock **status** (enum values)
- **Reference Switch** (locked/unlocked position sensor)
- **Battery voltage** (in millivolts)

Note: The Bluetooth MAC address of the SmartStall device is not exposed in the GATT profile. Use the 6-byte Device ID (MAC address) for identification.

GATT Service Overview

- **Service UUID:** c56a1b98-6c1e-413a-b138-0e9f320c7e8b
- **Characteristics:**

Characteristic	UUID	Properties	Type	Description
Stall Status	47d80a44-c552-422b-aa3b-d250ed04be37	READ	uint16_t	Indicates current lock state (e.g., INIT, LOCKED, SLEEP).
Reference Switch	2f8a5c10-8d9e-4b7f-9c11-0d2e5b7a4f22	READ	uint8_t	Physical reference switch: 0 = UNLOCKED , 1 = LOCKED .
Battery Voltage (mV)	7d108dc9-4aaf-4a38-93e3-d9f8ff139f11	READ	uint16_t	Battery voltage as an unsigned 16-bit integer in millivolts.

Characteristics

▀ Stall Status

- **UUID:** 47d80a44-c552-422b-aa3b-d250ed04be37
- **Properties:** READ
- **Format:** uint16_t
- **Description:** Indicates the current state of the stall lock.

Enum Values

Value	Name	Description
1	STATE_INIT	Device booted, reporting initialization
2	STATE_LOCKED	Stall is physically locked
3	STATE_UNLOCK	Stall has been unlocked
4	STATE_OPEN	Stall door has been opened
5	STATE_CLOSED	Stall door has been closed
6	STATE_SLEEP	Entering sleep mode
7	LOCK_TIMEOUT	20-minute timeout occurred
8	LOW_BATT	Battery too low to operate

▀ Reference Switch

- **UUID:** 2f8a5c10-8d9e-4b7f-9c11-0d2e5b7a4f22
- **Properties:** READ
- **Format:** uint8_t (boolean-like)
- **Description:** Hardware reference indicating the **actual mechanical position** of the lock, independent of software state.

Values

Value	Meaning
0	UNLOCKED
1	LOCKED

Use this to cross-check **Stall Status** or for safety interlocks.

▀ Battery Voltage

- **UUID:** 7d108dc9-4aaf-4a38-93e3-d9f8ff139f11
- **Properties:** READ
- **Format:** uint16_t

- **Units:** Millivolts (mV)
- **Description:** Most recent measured battery voltage of the device (typically measured at boot).

Example: 4910 → 4.910 V

BLE Advertisement

- **GAP Device Name:** SmartStall
- The SmartStall peripheral advertises with:
 - Connectable mode
 - Its 128-bit service UUID

Interaction Workflow

1. Central (hub) scans and connects to a SmartStall peripheral.
2. Read the desired characteristics as needed:
 - Stall Status
 - Reference Switch
 - Battery Voltage
3. Poll at your preferred cadence to detect changes (since all characteristics are **READ-only**).

Examples

We provide an `examples/` folder to help you build your own **Hub Firmware**:

- `examples/arduino/` – Arduino sketches (e.g., ESP32 using a BLE Central library)
- `examples/circuitpython/` – CircuitPython scripts for supported boards

What the examples do

- Discover all nearby **SmartStall** devices by filtering for the service UUID `c56a1b98-6c1e-413a-b138-0e9f320c7e8b` (*you may also optionally filter by the device name SmartStall*)
- Connect to each device in turn
- **Poll** the three READ-only characteristics and parse values
- Expose simple hooks so you can forward data to your back end

Custom integrations

The examples are structured so end customers can add cloud integrations such as:

- **AWS** (e.g., AWS IoT Core, Lambda, API Gateway)
- **Azure** (e.g., IoT Hub, Functions)
- Other REST/MQTT endpoints

Modify the provided publishing stub in each example to push readings upstream (per your security and networking model).

Example Usage (ESP32, pseudocode)

```
connectToSmartStall();

uint16_t status      = readCharacteristic(STALL_STATUS_UUID);
uint8_t  refSwitch  = readCharacteristic(REFERENCE_SWITCH_UUID);
uint16_t battMv     = readCharacteristic(BATTERY_VOLTAGE_UUID);

// Example interpretation
bool isLocked = (refSwitch == 1);
```

License

This API specification is provided under the MIT License. See LICENSE file for details.

© 2025 SmartStall. All rights reserved.