ESPHome

Overview

Integrate ESPHome-based devices into Control4. ESPHome is an open-source system that transforms common microcontrollers, like ESP8266 and ESP32, into smart home devices through simple YAML configuration. ESPHome devices can be set up, monitored, and controlled using a web browser, Home Assistant, or other compatible platforms. This driver enables seamless monitoring and control of ESPHome devices directly from your Control4 system.

Index

System requirements

- Control4 OS 3.3+

Features

- Local network communication requiring no cloud services
- Real-time updates from all supported entities exposed by the device
- Variable Programming Support

Compatibility

Verified Devices

This driver will generically work with any ESPHome device, but we have tested extensively with the following devices:

- ratgdo

If you try this driver on a product listed above, and it works, let us know!

Supported ESPHome Entities

Entity Type	Supported
Alarm Control Panel	❌
API Noise	❌
Binary Sensor	✅
Bluetooth Proxy	❌
Button	✅
Climate	❌
Cover	✅
Datetime	❌
Date	❌
Time	❌
Camera	❌
Event	❌
Fan	❌
Light	✅
Lock	✅
Media Player	❌
Number	✅
Select	❌
Sensor	✅
Siren	❌
Switch	✅
Text	✅
Text Sensor	✅
Update	❌
Valve	❌
Voice Assistant	❌

Installer Setup

> ⚠️ Only a ***single*** driver instance is required per ESPHome device. Multiple instance of this driver connected to the same device will have unexpected behavior. However, you can have multiple instances of this driver connected to ***different*** ESPHome devices.

DriverCentral Cloud Setup

> If you already have the DriverCentral Cloud driver installed in your project you can continue to Driver Installation.

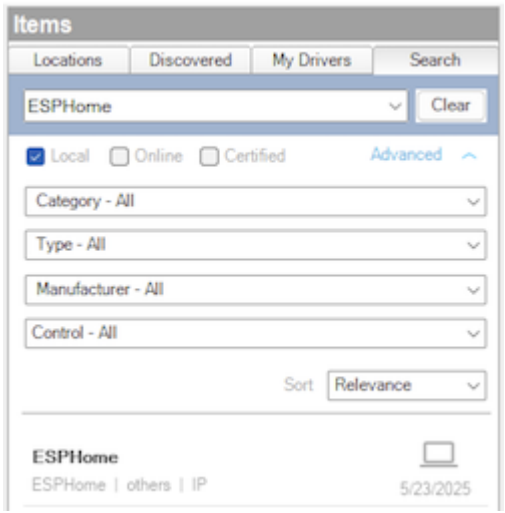
This driver relies on the DriverCentral Cloud driver to manage licensing and automatic updates. If you are new to using DriverCentral you can refer to their Cloud Driver documentation for setting it up.

Driver Installation

Driver installation and setup are similar to most other ip-based drivers. Below is an outline of the basic steps for your convenience.

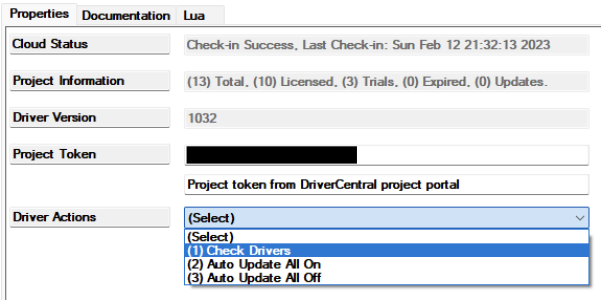
1. Download the latest `control4-esphome.zip` from DriverCentral.
2. Extract and install the `esphome.c4z`, `esphome_light.c4z`, and `esphome_lock.c4z` drivers.
3. Use the "Search" tab to find the "ESPHome" driver and add it to your project.

   > ⚠️ A ***single*** driver instance is required per ESPHome device.



4. Select the newly added driver in the "System Design" tab. You will notice that the `Cloud Status` reflects the license state. If you have purchased a license it will show `License Activated`, otherwise `Trial Running` and remaining trial duration.

5. You can refresh license status by selecting the "DriverCentral Cloud" driver in the "System Design" tab and perform the "Check Drivers" action.



6. Configure the Device Settings with the connection information.

7. After a few moments the `Driver Status` will display `Connected`. If the driver fails to connect, set the `Log Mode` property to `Print` and re-set the `IP Adress` field to reconnect. Then check the lua output window for more information.

8. Once connected, the driver will automatically create variables and connections for each supported entity type.

9. To control lights and/or locks, use the "Search" tab to find the "ESPHome Light" and/or "ESPHome Lock" driver. Add one driver instance for each exposed light or lock entity in your project. In the "Connections" tab, select the "ESPHome" driver and bind the light or lock entities to the newly added drivers.

Driver Setup

Driver Properties

Cloud Settings

Cloud Status

Displays the DriverCentral cloud license status.

Automatic Updates

Turns on/off the DriverCentral cloud automatic updates.

Driver Settings

Driver Status (read-only)

Displays the current status of the driver.

Driver Version (read-only)

Displays the current version of the driver.

Log Level [Fatal | Error | Warning | *Info* | Debug | Trace | Ultra]

Sets the logging level. Default is `Info`.

Log Mode [*Off* | Print | Log | Print and Log]

Sets the logging mode. Default is `Off`.

Device Settings

IP Address

Sets the device IP address (e.g. `192.168.1.30`). Domain names are allowed as long as they can be resolved to an accessible IP address by the controller. HTTPS is not supported.

> ⚠️ If you are using an IP address, you should ensure it will not change by assigning a static IP or creating a DHCP reservation.

Port

Sets the device port. The default port for ESPHome devices is `6053`.

Password

Sets the device password. If the device does not have a password, leave this field blank.

Device Info

Name (read-only)

Displays the name of the connected ESPHome device.

Model (read-only)

Displays the model of the connected ESPHome device.

Manufacturer (read-only)

Displays the manufacturer of the connected ESPHome device.

MAC Address (read-only)

Displays the MAC address of the connected ESPHome device.

Firmware Version (read-only)

Displays the firmware version of the connected ESPHome device.

Driver Actions

Reset Connections and Variables

> ⚠️ This will reset all connection bindings and delete any programming associated with the variables.

Reset the driver connections and variables. This is useful if you change the connected ESPHome device or there are stale connections or variables.

Developer Information

Support

If you have any questions or issues integrating this driver with Control4 or ESPHome, you can contact us at driver-support@finitelabs.com or call/text us at +1 (949) 371-5805.

Changelog

v20250606 - 2025-06-06

Added

- Initial Release