

Stamp-AddOn Cam3660 ESPHome 統合

このチュートリアルでは、Stamp-AddOn Cam3660 を Stamp-S3Bat に接続し、ESPHome を介して Home Assistant に統合する方法を説明します。



1. デバイス概要

項目	情報
製品名	Stamp-AddOn Cam3660
SKU	A182
対応コントローラー	Stamp-S3Bat
イメージセンサー	OV3660、3MP
イメージインターフェース	SCCB、8 ビット DVP パラレルインターフェース
ESPHome バージョン	2026.3.0 以降

2. 設定手順

ESP32 フレームワーク

完全なサンプルでは Arduino フレームワークを使用し、カメラに必要なコンパイルオプションを設定します。

```
esphome:
  name: stamp-addon-cam3660-example
  friendly_name: Stamp Addon Cam3660
  environment_variables:
    IDF_COMPONENT_STORAGE_URL: https://components-file.espressif.cn

esp32:
  board: esp32-s3-devkitc-1
  variant: esp32s3
  flash_size: 8MB
  framework:
    type: arduino
  sdkconfig_options:
    CONFIG_CAMERA_TASK_STACK_SIZE: "2048"
    CONFIG_CAMERA_PSRAM_DMA: "n"
```

起動時の初期化

カメラドライバーを初期化する前に、PWDN (GPIO39) を Low、RESET (GPIO41) を High に設定し、500ms 待機する必要があります。完全なサンプルでは、M5Stack の `early_init` 外部コンポーネントを使用してこの起動シーケンスを実行します。

```
external_components:
  - source: github://m5stack/esphome-yaml/components@main
    components: [early_init]

early_init:
  id: camera_power_up_barrier
  pins:
    - pin:
        number: GPIO39
        allow_other_uses: true
        level: false
    - pin:
        number: GPIO41
        allow_other_uses: true
        level: true
  delay: 500ms
```

起動シーケンス

`early_init`、GPIO39/GPIO41 のレベル設定、および 500ms の遅延は、カメラを正常に起動するために必要です。この設定は変更しないでください。

カメラ設定

Cam3660 の SCCB データラインには GPIO48/GPIO47 を使用し、画像データは 8 ビット DVP インターフェースから入力されます。

`allow_other_uses: true` を指定すると、カメラコンポーネントは起動時の初期化で使用した GPIO39 と GPIO41 を再利用できます。

```
i2c:
  - id: camera_i2c
    sda: GPIO48
    scl: GPIO47
    frequency: 100kHz

esp32_camera:
  name: "OV3660 Camera"
  id: cam
  i2c_id: camera_i2c
  data_pins:
    - GPIO21
    - GPIO18
    - GPIO16
    - GPIO15
    - GPIO14
    - GPIO12
    - GPIO38
    - GPIO40
  vsync_pin: GPIO42
  href_pin: GPIO17
  pixel_clock_pin: GPIO13
  external_clock:
    pin:
      number: GPIO46
      ignore_strapping_warning: true
    frequency: 20MHz
  power_down_pin:
    number: GPIO39
    allow_other_uses: true
  reset_pin:
    number: GPIO41
    allow_other_uses: true
  resolution: 320x240
  jpeg_quality: 12
  max_framerate: 10 fps
  idle_framerate: 0.5 fps
  frame_buffer_location: PSRAM
  frame_buffer_count: 1
  horizontal_mirror: true
  vertical_flip: true
  brightness: 1
  saturation: -2
```

このサンプルでは、デフォルトで QVGA (320 x 240) 解像度、最大フレームレート 10 fps、アイドル時フレームレート 0.5 fps を使用します。解像度やフレームレートを調整する場合は、設定を 1 項目ずつ変更し、デバイスが安定して動作することを確認してください。

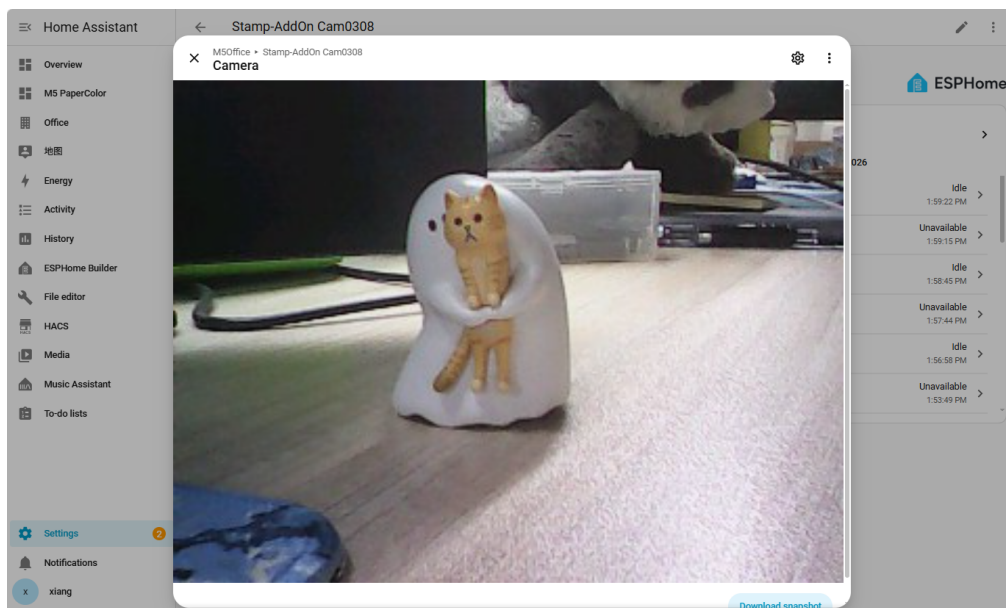
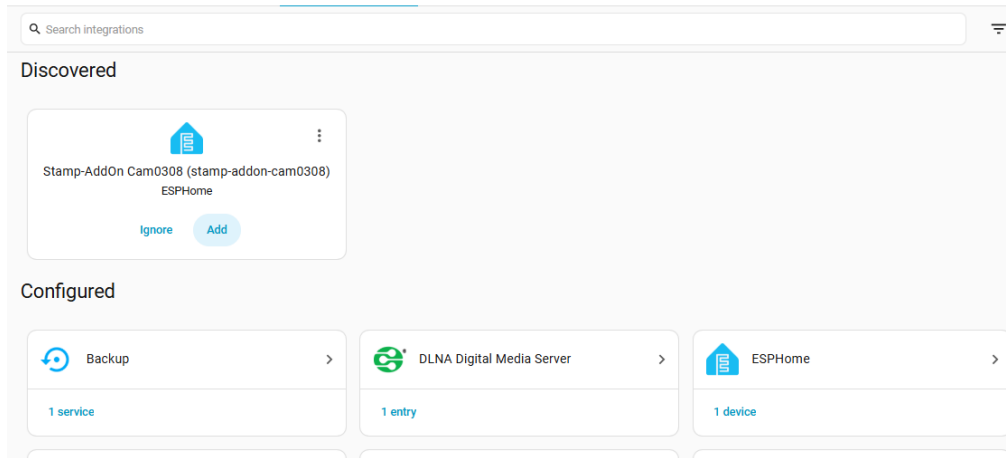
3. サンプル

完全な設定については、[stamp-addon-cam3660.yaml](#) を参照してください。このファイルには、デバイス、ネットワーク、API、OTA、起動時の初期化、およびカメラの設定が含まれています。

設定の修正

現在の完全なサンプルでは、`esphome.name` の末尾に不要なピリオドが含まれています。使用する前に `stamp-addon-cam3660-example.` を `stamp-addon-cam3660-example` に変更してください。変更しない場合、ESPHome でデバイス名が無効であると表示されます。

1. 完全なサンプルを ESPHome にインポートし、上記の手順に従ってデバイス名を修正します。
2. ESPHome の通常の手順に従って設定、コンパイル、ファームウェアの書き込みを行います。
3. デバイスがネットワークに接続されたら、Home Assistant で検出と追加を完了します。
4. デバイスページを開き、**OV3660 Camera** エンティティをダッシュボードに追加すると、カメラ映像を表示できます。



4. 関連リンク

- [Stamp-AddOn Cam3660 製品ページ](#)
- [Stamp-AddOn Cam3660 完全な設定](#)
- [early_init 外部コンポーネント](#)
- [ESP32 Camera 設定ドキュメント](#)
- [OV3660 データシート](#)