

Briefkasten

Überblick

Diese Automatisierung nutzt einen Aqara MCGQ11LM Tür-/Fenstersensor (über Zigbee2MQTT) zur Überwachung eines Briefkastens. Sie benachrichtigt bei Öffnen (z.B. Posteingang) auf zwei Handys, im Dashboard und als Popup – nur während der Postzeit (8-17 Uhr). Außerhalb dieser Zeit gibt es eine separate Warnung für ungewöhnliche Öffnungen. Zusätzlich trackt sie die Öffnungszeit und Dauer, minimiert Fehlalarme durch Self-Öffnungen und warnt bei langem Offenbleiben (z.B. blockierte Post). Throttling verhindert Spam-Benachrichtigungen.

Voraussetzungen

- Home Assistant mit Zigbee2MQTT-Integration.
- Aqara-Sensor gepairt; Entity-ID: `binary_sensor.briefkasten_contact` (State: 'on' = geöffnet, 'off' = geschlossen).
- HA Companion App auf zwei Handys für Benachrichtigungen (Services: `notify.mobile_app_dein_handy1` und `notify.mobile_app_dein_handy2`).
- Custom Integration: Browser Mod (über HACS) für Dashboard-Popups.
- Helpers erstellen (Configuration > Helpers):
 - Input Datetime: `input_datetime.briefkasten_letztes_oeffnen` (für Timestamp).
 - Input Boolean: `input_boolean.briefkasten_snooze` (für manuellen Override).
 - Input Boolean: `input_boolean.briefkasten_offen_cooldown` (für Throttling).
- Sensor für Dauer-Tracking (in `configuration.yaml`):

```
sensor:  
- platform: history_stats  
  name: Briefkasten offen Dauer  
  entity_id: binary_sensor.briefkasten_contact  
  state: 'on'  
  type: time  
  start: '{{ now().replace(hour=0).replace(minute=0).replace(second=0) }}'  
  end: '{{ now() }}'
```

Automatisierungen

Alle in `automations.yaml` oder über den visuellen Editor. Ersetze Entity-IDs bei Bedarf.

Haupt-Automatisierung: Benachrichtigung bei Öffnen (während Postzeit)

Diese Automatisierung erkennt das Öffnen des Briefkastens und überprüft, ob es sich um eine kurze Öffnung handelt (z.B. durch den Postboten), die typisch für den Einwurf von Post ist. Sie triggert nur während der definierten Postzeit (8-17 Uhr), um Fehlalarme außerhalb dieser Stunden zu vermeiden.

Zuerst wartet sie auf das Schließen des Briefkastens, berechnet die Öffnungsdauer und sendet Benachrichtigungen nur, wenn die Dauer unter 30 Sekunden liegt (anpassbar). Zusätzlich speichert sie den genauen Öffnungszeitpunkt in einem Helper und benachrichtigt über Handys und Dashboard-Popup. Der Snooze-Helper verhindert Auslösungen bei manueller Deaktivierung. Der Modus 'single' stellt sicher, dass keine doppelten Auslösungen erfolgen.

```
automation:
- id: briefkasten_benachrichtigung
  alias: Briefkasten geöffnet - Smarte Benachrichtigungen
  trigger:
    - platform: state
      entity_id: binary_sensor.briefkasten_contact
      from: 'off'
      to: 'on'
  condition:
    - condition: and
      conditions:
        - condition: time
          after: '08:00:00'
          before: '17:00:00'
        - condition: state
          entity_id: input_boolean.briefkasten_snooze
          state: 'off'
  action:
    - wait_for_trigger:
      - platform: state
        entity_id: binary_sensor.briefkasten_contact
        from: 'on'
        to: 'off'
      timeout:
        minutes: 5
        continue_on_timeout: false
    - variables:
        open_duration: "{{ (now() -
trigger.to_state.last_changed).total_seconds() }}"
    - condition: numeric_state
      entity_id: "{{ open_duration }}"
      below: 30 # Sekunden; anpassen für kurze Öffnungen
    - service: input_datetime.set_datetime
      target:
        entity_id: input_datetime.briefkasten_letztes_oeffnen
      data:
        datetime: "{{ now() }}"
    - service: notify.mobile_app_dein_handy1
      data:
        title: "Post ist da!"
        message: "Briefkasten geöffnet für {{ open_duration }} Sekunden um
{{ now().strftime('%H:%M Uhr am %d.%m.%Y') }}"
    - service: notify.mobile_app_dein_handy2
```

```

    data:
      title: "Post ist da!"
      message: "Briefkasten geöffnet für {{ open_duration }} Sekunden um
{{ now().strftime('%H:%M Uhr am %d.%m.%Y') }}."
    - service: browser_mod.notification
      data:
        title: "Posteingang"
        message: "Kurze Öffnung ({{ open_duration }} Sek) um {{
now().strftime('%H:%M Uhr am %d.%m.%Y') }}."
      mode: single

```

Warnung bei Öffnen außerhalb Postzeit

Diese separate Automatisierung dient als Sicherheitsfeature und triggert speziell bei Öffnungen des Briefkastens außerhalb der üblichen Postzeit (vor 8 Uhr oder nach 17 Uhr). Sie ist nützlich, um ungewöhnliche Aktivitäten zu erkennen, z.B. durch Dritte oder vergessene Öffnungen. Die Bedingung verwendet eine OR-Kombination für die Zeiträume, um den gesamten Tag abzudecken. Benachrichtigungen werden sofort gesendet, ohne Dauer-Überprüfung, da solche Ereignisse potenziell kritisch sind. Der Snooze-Helper erlaubt eine temporäre Deaktivierung, und der Modus 'single' verhindert Mehrfach-Benachrichtigungen bei wiederholtem Öffnen.

```

automation:
- id: briefkasten_ausserhalb_postzeit
  alias: Briefkasten außerhalb Postzeit geöffnet
  trigger:
    - platform: state
      entity_id: binary_sensor.briefkasten_contact
      from: 'off'
      to: 'on'
  condition:
    - condition: or
      conditions:
        - condition: time
          after: '17:00:00'
          before: '23:59:59'
        - condition: time
          after: '00:00:00'
          before: '08:00:00'
    - condition: state
      entity_id: input_boolean.briefkasten_snooze
      state: 'off'
  action:
    - service: notify.mobile_app_dein_handy1
      data:
        title: "Warnung: Briefkasten geöffnet!"
        message: "Briefkasten außerhalb Postzeit geöffnet um {{
now().strftime('%H:%M Uhr am %d.%m.%Y') }}."
    - service: notify.mobile_app_dein_handy2
      data:
        title: "Warnung: Briefkasten geöffnet!"

```

```
message: "Briefkasten außerhalb Postzeit geöffnet um {{
now().strftime('%H:%M Uhr am %d.%m.%Y') }}."
- service: browser_mod.notification
  data:
    title: "Warnung: Außerhalb Postzeit"
    message: "Briefkasten geöffnet um {{ now().strftime('%H:%M Uhr am
%d.%m.%Y') }} – Überprüfen?"
    duration: 60
mode: single
```

Warnung bei langem Offenbleiben

Diese Automatisierung überwacht, ob der Briefkasten länger als 5 Minuten (anpassbar) geöffnet bleibt, was oft auf blockierte Post oder einen nicht richtig geschlossenen Deckel hinweist. Der Trigger wartet explizit auf die Dauer, bevor er auslöst, um kurze Öffnungen zu ignorieren. Die Benachrichtigung enthält die genaue Dauer seit dem Öffnen, berechnet dynamisch. Throttling wird durch den Cooldown-Helfer umgesetzt, der nach Auslösung aktiviert wird und Benachrichtigungen für eine Stunde blockiert. Der Modus 'restart' sorgt dafür, dass bei anhaltendem Offenbleiben keine Spam-Nachrichten entstehen, sondern der Timer neu gestartet wird.

```
automation:
- id: briefkasten_offen_warnung
  alias: Briefkasten steht offen - Warnung
  trigger:
    - platform: state
      entity_id: binary_sensor.briefkasten_contact
      to: 'on'
      for:
        minutes: 5
  condition:
    - condition: state
      entity_id: input_boolean.briefkasten_offen_cooldown
      state: 'off'
  action:
    - service: notify.mobile_app_dein_handyl
      data:
        title: "Briefkasten steht offen!"
        message: "Der Briefkasten ist seit {{ (now() -
states.binary_sensor.briefkasten_contact.last_changed).total_seconds() | int
// 60 }} Minuten offen. Vielleicht blockiert Post? (Stand: {{
now().strftime('%H:%M Uhr') }})"
    - service: notify.mobile_app_dein_handyl2
      data:
        title: "Briefkasten steht offen!"
        message: "Der Briefkasten ist seit {{ (now() -
states.binary_sensor.briefkasten_contact.last_changed).total_seconds() | int
// 60 }} Minuten offen. Vielleicht blockiert Post? (Stand: {{
now().strftime('%H:%M Uhr') }})"
```

```
- service: browser_mod.notification
  data:
    title: "Warnung: Briefkasten offen"
    message: "Offen seit {{ (now() -
states.binary_sensor.briefkasten_contact.last_changed).total_seconds() | int
// 60 }} Min. – Überprüfen?"
    duration: 60
- service: input_boolean.turn_on
  target:
    entity_id: input_boolean.briefkasten_offen_cooldown
mode: restart
```

Snooze-Reset

Diese Hilfsautomatisierung setzt den Snooze-Helper automatisch zurück, nachdem er 10 Minuten aktiviert war. Sie triggert auf den 'on'-Zustand des Booleans und wartet die Dauer ab, bevor sie ihn deaktiviert. Das ermöglicht eine temporäre Unterdrückung von Benachrichtigungen (z.B. bei eigener Nutzung des Briefkastens), ohne dass man manuell eingreifen muss. Sie betrifft alle Automatisierungen, die den Snooze prüfen, und sorgt für eine nahtlose Wiederaktivierung des Systems.

```
automation:
- id: briefkasten_snooze_reset
  alias: Briefkasten Snooze zurücksetzen
  trigger:
    - platform: state
      entity_id: input_boolean.briefkasten_snooze
      to: 'on'
      for:
        minutes: 10
  action:
    - service: input_boolean.turn_off
      target:
        entity_id: input_boolean.briefkasten_snooze
```

Cooldown-Reset für Offen-Warnung

Ähnlich wie der Snooze-Reset, aber speziell für den Cooldown der Offen-Warnung. Sie triggert nach 1 Stunde auf den 'on'-Zustand des Booleans und setzt ihn zurück. Das implementiert Throttling, um zu verhindern, dass bei dauerhaftem Offenbleiben stündlich Benachrichtigungen gesendet werden. Es hält das System effizient und vermeidet Überflutung mit redundanten Warnungen.

```
automation:
- id: briefkasten_offen_cooldown_reset
  alias: Offen-Warnung Cooldown zurücksetzen
  trigger:
    - platform: state
      entity_id: input_boolean.briefkasten_offen_cooldown
```

```
to: 'on'
for:
  hours: 1
action:
- service: input_boolean.turn_off
  target:
    entity_id: input_boolean.briefkasten_offen_cooldown
```

Dashboard-Integration

- Entities Card für Sensor-Status.
- Markdown Card für Timestamp: Letztes Öffnen:
states_input_datetime.briefkasten_letztes_oeffnen
- Gauge Card für Dauer: sensor.briefkasten_offen_dauer
- Button für Snooze: Toggle input_boolean.briefkasten_snooze
- Conditional Card für Warnung: Briefkasten:
offen_blockiert

Tipps und Anpassungen

- Teste mit Developer Tools > States/Services.
- Passe Dauern (z.B. 30 Sek, 5 Min) an dein Verhalten an.
- Die Postzeit (8-17 Uhr) ist in den Conditions integriert; bei Bedarf erweitern (z.B. Wochentage mit condition: time weekday).
- Erweiterbar mit Kamera oder Vibration-Sensor für bessere Genauigkeit.

Diese Setup minimiert Fehlalarme und trackt alles Elegante.

From:
<https://drklipper.de/> - **Dr. Klipper Wiki**

Permanent link:
https://drklipper.de/doku.php?id=haussteuerung:ha_automatisierungen:briefkasten

Last update: **2025/10/28 07:47**

