

Voici toutes les suggestions et explications que nous avons abordées pour ton projet de home-lab avec Flipper Zero ou équivalent DIY :

Home-lab IoT avec Flipper Zero / équivalent DIY

Objectif

Créer un système de contrôle et d'automatisation pour ton home-lab, en utilisant un Flipper Zero ou un équivalent basé sur ESP32, orienté IR, boutons physiques, et éventuellement NFC, avec intégration dans Home Assistant via MQTT.

Fonctions clés possibles

1. **Hub IR portable** : piloter TV, ampli, projecteur, etc., et déclencher des scènes Home Assistant.
2. **Bouton physique / HID** : déclencher des scripts ou automatisations sur un PC ou Home Assistant.
3. **Inventaire / observatoire** : catalogue des commandes IR et badges NFC possédés, dashboard de l'inventaire.
4. **Extension portable pour automatisation** : déclencheur pour scènes ou alertes.

Matériel recommandé (sans radio)

- ESP32-S3 devboard (ou ESP32-WROOM) : 8-20€
- LED IR + résistance 100-220Ω : ~2€
- Récepteur IR TSOP382/1838 : ~2€
- Câbles dupont / breadboard : ~5€
- Alimentation USB 5V : ~5€
- Optionnel : Module PN532 NFC pour futur ajout : 8-15€
- Raspberry Pi 4 (ou VM) pour Home Assistant / MQTT broker : variable

Plan 6 semaines

S1 : Préparation, commandes, installation Home Assistant ou vérification MQTT.

S2 : Flash ESP32 avec ESPHome minimal et Wi-Fi.

S3 : Ajouter IR transmit (LED), tester envoi de codes IR.

S4 : Ajouter IR receive + bouton physique (binary_sensor).

S5 : Automatisations dans Home Assistant (scènes, notifications).

S6 : Boîtier, sécurité réseau, sauvegarde, documentation.

Exemple ESPHome (IR transmit + bouton)

```
esphome:  
  name: esp-ir-button  
  platform: ESP32  
  board: esp32-s3-devkitc-1
```

```

wifi:
  ssid: "TON_SSID"
  password: "TON_MOT_DE_PASSE_WIFI"

ap:
  ssid: "EspFallback-XXXX"

logger:

ota:

mqtt:
  broker: 192.168.1.10
  username: mqtt_user
  password: mqtt_pass

remote_transmitter:
  pin: GPIO17
  carrier_duty_percent: 50%

switch:
  - platform: template
    name: "IR Send Play"
    turn_on_action:
      - remote_transmitter.transmit_nec:
          address: 0x00
          command: 0x10EF

binary_sensor:
  - platform: gpio
    pin:
      number: GPIO00
      mode: INPUT_PULLUP
      inverted: True
    name: "Physical Button"
    on_press:
      - switch.turn_on: ir_send_play

```

Apprentissage IR

1. Activer `remote_receiver` pour apprendre les trames IR de tes télécommandes.
2. Copier les codes affichés pour `remote_transmitter`.

Alternatives / variantes

- USB HID sur ESP32-S3 ou Raspberry Pi Pico W (plus complexe).
- Ajouter NFC PN532 plus tard.

Sécurité & bonnes pratiques

- VLAN ou isolement réseau pour IoT.
 - Changer tous les mots de passe par défaut.
 - OTA sécurisé pour les firmware ESPHome.
 - Éviter radio sub-1GHz pour l'instant pour limiter complexité et risques légaux.
-

Résumé

Même sans Flipper Zero réel, tu peux créer un équivalent pratique et utile pour ton home-lab avec ESP32 + IR + bouton, piloté via Home Assistant. Ce setup est rapide à prototyper, extensible (NFC plus tard), et légal/contrôlable entièrement sur ton réseau.