


# LoRa APRS-iGate für den Raspberry Pi

(c) 2024, Peter Stirnimann, [hb9pae@gmail.ch](mailto:hb9pae@gmail.ch) Software-Version: 1.3.0



Status Wetter Logfile Konfiguration Info

## Status - LoRa APRS-iGate

iGate Call	HB9PAE-1
Connect to APRS-IS	True
iGate LAT	47.54645
iGate LON	8.91262
iGate Altitude	483
Sensor BME280	False
BME280 Intervall	300
Temperatur	1.0
Luftdruck	1.0
Luftfeuchtigkeit	1.0
Last Message	--- None ---
RSSI	0
Pkt RSSI	0
SNR	0
Packet Err	0
APRS-IS Message	1
RX Count	0
Wx-Data	False
WX Intervall	300
Beacon Intervall	600
Beacon Message	-x
Uptime	0h 0m 39s
Version	0.9.7
DirtyFlag	False

2023-08-18 12:52:47

Copyright © [hb9pae@swiss-artg.ch](mailto:hb9pae@swiss-artg.ch)

## Allgemeines

Der LoRa Empfänger empfängt Positionssignale und APRS-Meldungen im Amateurradio LoRa-Band (433.775 MHz). Die decodierten APRS-Meldungen werden anschliessend an die APRS Datenbank weitergeleitet. Diese können anschliessend unter <http://aprs.fi> abgerufen und angezeigt werden.

Die empfangenen Daten werden im Status-Report auf dem OLED-Display oder über einen Webbrowser :5000 (<http://192.168.0.123:5000>) abgerufen werden.

Das Python Programm IGate.py steuert das LoRa RPI-Board der SWISS-ARTG mit dem RF95W Chip RF-Hope als LoRa-Empfänger. Weitere Informationen zum RPI-Board unter: <https://www.swiss-artg.ch/index.php?id=174> ( Menu: Digital Data > LoRa-APRS > LoRa Gateway)

## Neues in der Version 1.3

- Die Konfigurationsdaten sind mit einem Passwort geschützt (Web-Interface).
- Dialoge auf dem OELD-Display überarbeitet.
- Neue Tastenbelgung
- Umstellung Betriebssystem auf Debian 12 / Python 3.11.
- Ersatz der veralteten Python Bibliotheken.
- Anpassungen lora.c Compileroptionen
- Modul HMI.py in Button.py und Display.py aufgeteilt.
- Button.py: neue Lib Igoio, Tastenbelegung neu 1:Status, 2: Pkt Info, 3: Config 4: Menu up, 4 Menu Down
- Display.py: neue Lib oled-txt (Ersatz für die veraltete Adafruit-SSD1306 Lib)

## Copyright

Das hier dokumentierte Programm ist Open Source, der Programmcode ist frei verfügbare.

## Projektdokumentation

- Quellprogramme [https://github.com/swiss-artg/LoRa-APRS\\_RPI-iGate](https://github.com/swiss-artg/LoRa-APRS_RPI-iGate)
- Beschreibungen unter <https://cloud.hoststar.ch/s/GytsTsQSYSoYAXk>
  - Imagedatei zum Raspberry PI Version 3+ / 4:
  - Schema, Stückliste, Aufbauanleitung
  - Kurzbeschreibung und Installationsanleitung

## Bausatz LoRa APRS-iGate RPI-Aufsteckplattone

Die SWISS-ARTG stellt bietet einen Bausatz (Platine und alle Bauteile, solange Vorrat) an. Interessenten melden sich unter [info@swiss-artg.ch](mailto:info@swiss-artg.ch) oder beim Autor.

### Bestückung RPI-Board

Schema, Stückliste und Aufbauanleitung sind auf der SWISS-ARTG Webseite oder im Git-Repository verfügbar. Eine Bestückungsanleitung ist im Git Repository verfügbar.

### Modifikation RPI-Board

Die neue Version vom RPI-Board muss nicht modifiziert werden. Versions-Beschriftung oberhalb Diode D1: «231125».

Die erste Generation des RPI-Boards der SWISS-ARTG muss für den Interrupt-Betrieb modifiziert werden. Verbinde dazu Pin 14 vom RFM96W (DIO 0) mit Pin 11 (BCM17) vom RPI 40-pol Header.

### Wetterstation

Falls ein Sensor BME280 angeschlossen und aktiviert ist ("SENSOR BME280 = TRUE" wird der Sensor alle 5 Minuten abgefragt. Ist das FFlag "Wx-Data = TRUE" werden die Sensordaten auch an APRS.FI gesendet. Die Sensordaten werden intern im Menue "WETTER" angezeigt.

## Installation

Für die Installation wird ein IMAGE zur Verfügung gestellt. Der Link zur aktuell verfügbaren Version ist auf der SWISS-ARTG Webseite (Menu: Digital Data > LoRa APRS > LoRa Gateway) zu finden.

Kopiere das Image-File mit einem Imager-Programm (z.B. Raspberry Pi Imager) auf eine SD-Karte (mind. 8 GB).

## Manuelle Installation

Der Programmcode kann auch direkt vom Git Repository geladen werden. Dabei müssen die erforderlichen Bibliotheken installiert werden. Weitere Hinweise zur manuellen Installation sind im INSTALL.md dokumentiert.

## Inbetriebnahme

- Das modifizierte LoRa RPI-Board auf dem Raspberry PI aufstecken.
- Die programmierte SD-Karte im RPI einstecken.
- RPI mit dem lokalen Netzwerk verbinden.
- PRI mit der Stromversorgung (5 VDC) versorgen.

Nach dem ersten Start des Raspberry PI wird das Filesystem auf der SD-Karte expandiert, es folgen mehrere Restarts. Nach dem erfolgreichen Start des Programmes erscheint der Welcome-Bildschirm auf dem OLED-Display.

## Konfiguration

Die Konfiguration der persönlichen Daten (Rufzeichen, Koordinaten etc.) erfolgt über einen Web-Browser:

- Adresse <http://:5000>, Reiter Konfiguration
- Abfrage Benutzer und Passwort. Als Benutzername wird das iGate Rufzeichen verwendet.
  - Default Benutzer: "NOCALL", Passwort "geheim".
  - Das Passwort kann im Konfigurationsmenu geändert werden.
- Trage iGate Rufzeichen, den Passcode und die Standortdaten im Konfigurations-Formular ein und speichere die Daten. Die Parameter werden im File /opt/RPI-iGate-LoRa/igate.ini abgespeichert.

## Spezielle Parameter

- EN\_APRSIS: True / False: Bestimmt, ob das iGate Daten an das APRS-IS System übermittelt.
- EN\_BME280: True / False: BME280 Sensor angeschlossen
- EN\_WXDATA: True / False: Bestimmt ob die BME280 Sensordaten als WX-Bake an das APRS-IS geliefert werden.

## Bedienung über das Terminal

Das Terminal (Konsole) kann über SSH oder direkt mit Bildschirm und Tastatur am RPI erreicht werden.

### Start Befehl

- `sudo systemctl start igate.service`

### Automatischer Start nach dem Booten

- `sudo systemctl enable igate.service`

## Funktion der 5 Tasten

Die drei unteren Tasten auf dem RPI-Board wählen den Anzeigemodus

- Taste links > Status
- Taste mitte > Konfiguration
- Taste rechts > Letzte Meldung

Die beiden oberen Tasten unter dem Display (Menu vor/zurück) ermöglichen den sequentiellen Aufruf aller Dialoge.

## Hardwarekonfiguration

Der LoRa Treiber erwartet folgende Hardwarekonfiguration (File LORA/lora.c)

- `int ssPin = 10; // ChipSelect BCM 8`
- `int dio0 = 21; // IRQ BCM 5`
- `int RST = 22; // RESET BCM 6`