

## Software - Vorgaben für Raspberry UFO-Station.

Die Software wird mit Python erstellt.

Generell: Das Systemprogramm ist in Funktionsmodule unterteilt.

- 1- Funkuhr - Auswertung
- 2- GPS – Auswertung
- 3- Sensor – Auswertung
- 4- Bildverarbeitung Raspi - Kameramodul
- 5- Videoverarbeitung USB – Grabber Videokamera , möglichst XVGA
- 6- Optional Bewegungserkennung (wird ev. von Wolfi beigesteuert)
  - Alle Messwerte , die Bilder und Videos werden auf die SD-Karte gespeichert.
  - Es werden zu keiner Zeit aus Platzgründen Daten überschrieben.
  - Ist die SD-Karte voll, dann schaltet sich der Raspi ab.
- 7- Optional eine Datenaufzeichnung zusätzlich über USB-Festplatte.

Nach dem Einschalten werden erstmalig nach 30 Sekunden die Sensoren gemessen.  
Für dieser Zeit Blinkt ein (GPIO) als Marker, das System ist noch nicht bereit.

Funktionsablauf und Details:

- 1- Einschalten (Spannung anlegen), Raspi läuft automatisch und starte die Programme automatisch.
- 2- Bei Unterspannung oder Stromabschaltung geht Raspi aus, um bei wiederkehrender Spannung erneut selbständig zu starten, wie unter Punkt 1.
- 3- Die Messwertdaten , 3x Magnetfeld, 3x Beschleunigung, Temperatur, Luftfeuchte, werden in eine Datei geschrieben (Excel oder anderes gut verwertbares Format).  
Die Abspeicherung erfolgt als Messwertgruppe mit Datum/Uhrzeit/GPS-Position.  
Es gibt zwei Arten den Datenaufzeichnung.

**-Standart:** Alle 2 Minuten eine Messwertgruppe abspeichern.

**-Alarm:** Die Auslösung dieser Funktion ist auf zwei (drei) Arten möglich.

1. Der externe Alarmeingang (GPIO 21) ist aktiv.
  2. Über- oder Unterschreiten der aktuellen Meßwerte (hierzu siehe Tabelle M).
  3. Ein Alarm-Flag wird von der Bewegungserkennung gesetzt (Optional).
- Für die Zeit des anstehenden Alarms wird eine (GPIO 16) geschaltet.

Bei erkanntem **Alarm** erfolgt die Aufzeichnung der Meßwertgruppe mit mindestens 20 Messungen in der Sekunde. Dies Alarmaufzeichnung erfolgt für 2 Minuten. Bei einer folgenden weiteren Meßwertänderung (wie **2.**) wird diese Zeit um jeweils weitere 2 Minuten verlängert.

### 4- Einzelaufzeichnung der Raspi – Systemkamera.

- Immer 1Bild/Sekunde, möglichst mit angepaßter Belichtungszeit.

Diese Funktion bei **Alarm** bzw. im einstellbaren **Zeitfenster** (hierzu siehe Tabelle Z).

- In die Bilder werden die jeweils aktuellen Meßwerte und Datum/Zeit eingeblendet.
- Bei **Alarm** werden auch die 20 vorgehenden Bilder gespeichert. Tritt der **Alarm** im Zeitfenster auf, dann ist das nicht notwendig.

## 5- Videoaufzeichnung über USB-Grabber.

-- In die Videobildern wird Datum/Uhrzeit eingeblendet.

- **Standart:** 1 Bild/Sek. , diese Funktion im einstellbaren **Zeitfenster** (hierzu siehe Tabelle Z).
- **Alarm:** 25 Bilder/Sek. , im oder außerhalb des **Zeitfensters**.  
Die 20 Sekunden vor dem **Alarm** werden ebenfalls mit 25 Bilder/Sek. gespeichert.

-----  
Funktions- und Bedien- Einstellungen für den Raspberry werden über einfache Textdateien bestimmt. Diese Textdateien werden auf einem USB-Stick abgespeichert. Geändert und Erweitert werden diese mit einem gängigen Textprogramm unter Windows oder Linux., auf jedem beliebigen Computer.

Ist der USB-Stick mit einer solchen Dateien beim Systemstart im Raspberry angesteckt, so erkennt das Programm dies und lädt die TxT Dateien (Tabellen) für den Programmablauf. Eine bereits im System vorhandene TxT Datei wird überschrieben. Das lesen und eine Fertigmeldung werden über (GPIOs) an LEDs angezeigt.

### #Tabelle M

### # Kommentar zur Information

# Beispiel: Code. MgX xxx //xxx kann 000 bis 255 sein.  
# Die Zahl zeigt den Bruchteil vom Startwert an.  
# Ist zB. 025 eingetragen, dann erfolgt ein **Alarm** wenn die Abweichung > 10%  
# Ist 000 eingetragen dann erfolgt ohne Unterbrechung **Alarm**.  
# Ist 255 eingetragen dann erfolgt zu keiner Zeit ein **Alarm**.

Code. MgX 025 # Eine Zahl zwischen 000 und 255 Magnetfeld X  
Code. MgY 025 # Eine Zahl zwischen 000 und 255 Magnetfeld Y  
Code. MgZ 025 # Eine Zahl zwischen 000 und 255 Magnetfeld Z  
Code. GvX 005 # Eine Zahl zwischen 000 und 255 Gravitation X  
Code. GvY 005 # Eine Zahl zwischen 000 und 255 Gravitation Y  
Code. GvZ 005 # Eine Zahl zwischen 000 und 255 Gravitation Z

Die restlichen Meßwerte wie Luftdruck, Temperatur... werden nicht berücksichtigt.

### #Tabelle Z

# besteht nur aus einer Zeile welche das Zeitfenster in vollen Stunden angibt .

#Beispiel: Code. Zeit. 00-01 Einschaltfenster 00Uhr bis 01 Uhr.  
Code. Zeit. 02-17 Einschaltfenster 02Uhr bis 17 Uhr .  
Code. Zeit. 12-19 Einschaltfenster 12Uhr bis 19 Uhr.  
Code. Zeit. 11-12 Einschaltfenster 11Uhr bis 12 Uhr  
Code. Zeit. 19-24 Einschaltfenster 19Uhr bis 24 Uhr  
Code. Zeit. 17-10 Einschaltfenster 17Uhr bis 10 Uhr  
Code. Zeit. 00-00 Einschaltfenster 00Uhr bis 00Uhr, (24 h Dauerbetrieb.)  
Code. Zeit. 99-99 Es gibt kein Einschaltfenster (0 h Dauerbetrieb.)  
Code. Zeit 17-07 Das Einschaltzeitfenster ist von abends 17 Uhr bis morgens 7 Uhr

# Festlegung der GPIO

## **Funktionskontrolle USB-Stick:**

GPIO 24    LED grün    Dauerlicht für 10 Sekunden wenn eine gültige TxT-Datei gelesen wurde. Stick kann dann entfernt werden.

GPIO 23    LED rot        blinkt wenn TxT-Datei auf dem USB-Stick fehlerhaft ist.  
Dauerlicht wenn keine TxT-Datei auf dem USB-Stick ist

## **Möglich ist auch statt GPIOs LEDs aus der LED-Matrix des Sense HAT zu benutzen.**

GPIO 16    Funktionsausgang, schaltet H für die Zeit in welcher ein Alarm erkannt wird.

GPIO 20    Funktionsausgang, schaltet H mit 1 Hz. (Sekundentakt).

GPIO 19    Funktionsausgang, schaltet H mit 1 Hz. (Sekundentakt) für die Alarmzeit.

GPIO 21    Funktionseingang, wenn L dann schalte in Alarm.

Vorhandene Programme - Module die verwendet werden können.

[http://www.netzmafia.de/skripten/hardware/RasPi/RasPi\\_Kamera.html](http://www.netzmafia.de/skripten/hardware/RasPi/RasPi_Kamera.html)

<http://raspberry-at-home.com/video-grabber-for-raspberry-pi/>

<http://www.raspberry-pi-geek.de/Magazin/2014/01/Raspberry-Pi-als-Funkuhr-und-NTP-Stratum-1-Zeitserver>

<https://www.raspberrypi.org/documentation/hardware/sense-hat/>