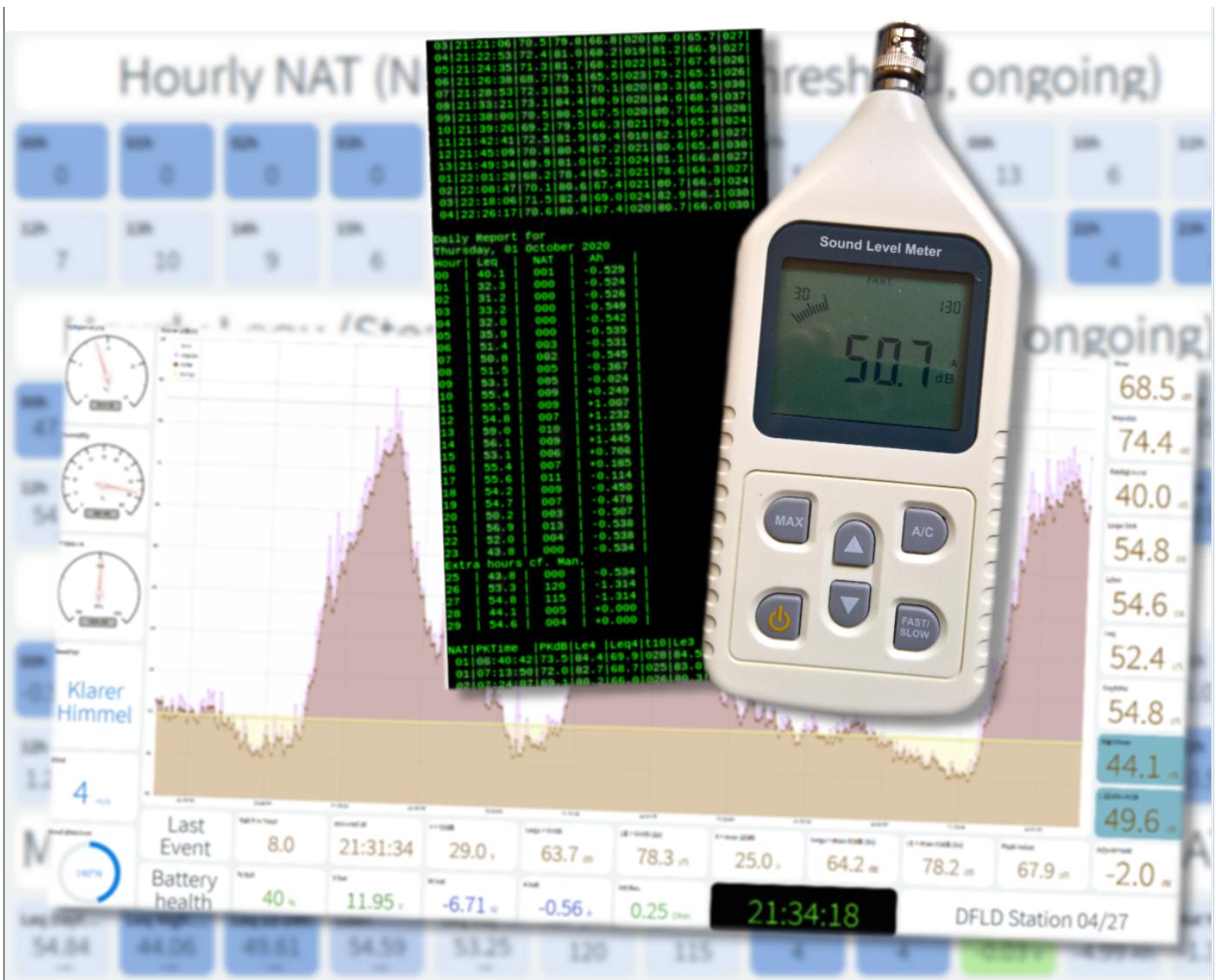


Netzwerk-Erweiterung

für einen Schallpegelmesser

(die eierlegende Wollmilchsau)😊



Zusammenfassung

Es gibt eine große Auswahl an Schalldruckpegelmessgeräten auf dem Markt, von ziemlich billigen bis zu sündhaft teuren. Obwohl sie nicht für eine rechtliche Durchsetzung geeignet sind, weisen die Billigen häufig eine ausreichende Genauigkeit für viele Zwecke aus.

Die meisten haben jedoch entweder keine, oder nur äußerst primitiven Reporting-Fähigkeiten.

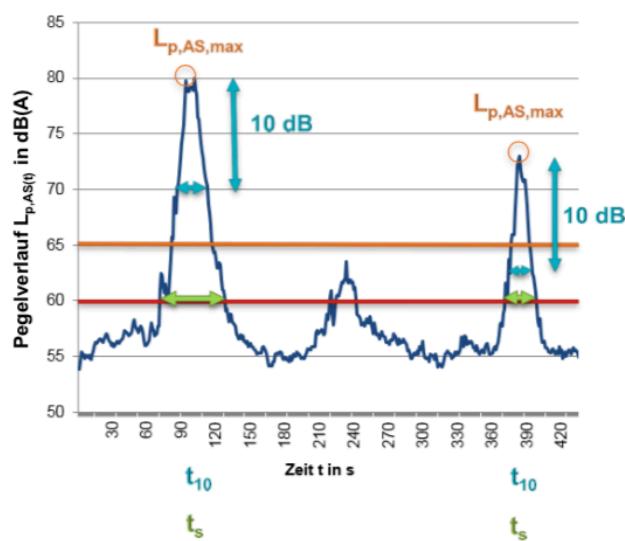
Der Zweck dieser Entwicklung besteht darin, Netzwerkfunktionen für verschiedene Schalldruckmessgeräte der Verbraucherklasse bereitzustellen und fortschrittliche akustische Metriken bereitzustellen.

Diese kommen den Anforderungen IEC 61672-1: 2013 und DIN 45643 2011 an die Lärmüberwachung von Fluglärm sehr nahe :

vgl: <https://www.ecac-ceac.org/documents/10189/51566/01.+Doc29+4th+Edition+Volume+1.pdf/bfde6e09-b46b-44e1-b73f-388fc3527aa>
sowie:https://www.fraport.com/content/dam/fraport-company/images/umwelt/schallschutz/flug%C3%A4rm/dokumente/Umstellung%20des%20Messverfahrens.pdf/_jcr_content/renditions/original/Umstellung%20des%20Messverfahrens.pdf

Dazu wird dem Schallpegelmessgerät ein ESP8266-WLAN-Mikrocontroller hinzugefügt, der weniger als 4 € kostet.

Die Modifikation bietet USB- und WiFi-Konnektivität und kann programmiert werden, um die coolsten Dinge zu tun, die nur sehr teure High-End-Geräte bieten: erweiterte Messpegelzeitkonstanten, zeitbezogene Aggregationen und Ereigniserkennung unter Verwendung fester / schwebender Schwellenwerte gemäß ECAC-Standards:



Die Software berücksichtigt mögliche starke Schwankungen des Geräuschpegels und integriert innerhalb der Analysezeit aufeinanderfolgende Grenzüberschreitungen.

Das gegebene Programm ist hochgradig konfigurierbar, intensiv dokumentiert und kann leicht an andere Anforderungen zur Überwachung des Verkehrslärms, wie z.B. Straßenlärm angepasst werden

- Eisenbahnlärm
- öffentliche Veranstaltungen und so weiter ...

Der Schalldruck wird intern alle 125ms ausgewertet und verarbeitet, um (gleichzeitig) die folgenden Zeitreaktionsstandards bereitzustellen:

- Fast (Attack t = 125 ms, Decay t = 125ms)
- Slow (Attack t = 1S, Decay 4,3 dB / s)
- Impuls (Attack t = 125 ms, Decay 2,9 dB / s)
- Realer Spitzenwert pro Minute (125ms Auflösung, nicht das Maximum der Messwerte)
- Hintergrundpegel (t = 2000 Sekunden, dabei bleiben Ereignisse über dem Schwellenwert unberücksichtigt)

Die Verarbeitung und die Berechnung der Metriken erfolgen unter Verwendung einer mathematisch korrekten Mittelung der Schallenergie gemäß der folgenden Gleichung:

$$L_{p,A,eq,T} = 10 \lg \left(\frac{t_0}{T} \sum_{i=1}^N 10^{L_{p,A,E,i}/10 \text{ dB}} \right) \text{dB}$$

Die Software bietet Metriken gemäß den Lärmstandards für Fluglärmessungen:

Laufend	Exposition / zeitlich gemittelt / kumulativ			Einzelereignisse
Jede Sekunde	Im Minutentakt *	Im Stundentakt*	Um Mitternacht *	Ereignisgesteuert*
Alle oben genannten Werte (abgefragt).	Leq	Leq	Leq 24h	Höchstwert
	Maximum in der Minute	Anzahl über festem Schwellenwert (NAT)	Leq Night (22:00 - 06:00)	Leq relativ zur festen Schwelle
	Minimum in der Minute	Wetterbedingungen von openweathermap.org	Leq Day (6:00 - 22:00)	Leq relativ zum variablen Schwellenwert max-10dB
	Hintergrundpegel		Leq (22:00 - 24:00 Uhr)	LE relativ zur festen Schwelle
			Lden	LE relativ zum variablen Schwellenwert max-10dB
			NAT Night (22:00 - 06:00 Uhr)	Zeit über dem festen Schwellenwert
			NAT-Tag (6:00 - 22:00 Uhr)	Zeit. über variablem Schwellenwert max-10dB
			NAT (22:00 - 24:00 Uhr)	Zeit der Ereignisspitze
			Tageszusammenfassung Leq für jede Stunde	
			Tageszusammenfassung Nat für jede Stunde	

* 365 Tage Langzeitspeicherung und Excel-Export sind dabei möglich

Die Software bietet Berichte und eine Schnittstelle zum Plotten/Berichten sowie Dashboards über den kostenlosen Cloud-Dienst <http://thinger.io>

Zurück zu den Wurzeln des gesamten Projekts kann die Software optional eine Schnittstelle zu www.dfld.de/. www.eans.net, herstellen.

DFLD ist ein Netzwerk aus mehr als 600 Fluglärmessstationen, das über 15 Jahre lang langfristig Daten und Statistiken aufzeichnet. (Dazu wird ein Gateway (z. B. ein Raspberry Pi) zusätzlich benötigt).

Das Programm ruft auch Wetterinformationen von openweathermap.org ab und zeichnet zu den Messwerten die entsprechenden meteorologischen Bedingungen auf, wodurch die Ereigniserkennung und die damit verbundenen Fehlalarme bei übermäßigem Wind verhindert werden.

Hardware

Neben der Software selbst, dokumentiere ich einige kostengünstige Hardwarelösungen, von relativ einfach mit nur drei zu lötende Drähten bis hin zu dem vollständigen Konzept eines wetterfestes Außengeräts mit überwachter netzunabhängiger solarbetriebener Energie.

Ein Kalibrierungsverfahren an zwei festen Punkten 94dB und 47dB stellt sicher, dass der Eingangsmessbereich (inkl. Evtl. Ein Offset) mit dem DC-Ausgangssignal des Geräts übereinstimmt.

Mit zusätzlicher Hardware ist auch eine vollständige Solarbatteriemanagementfunktion für einen drahtlosen Betrieb realisierbar. (Diese optionale Funktionalität wird separat beschrieben, Einstellungen dazu und Berichte dafür werden in **Roter Schrift** gekennzeichnet und können ggf. ignoriert werden).

Berichte

Das Programm kann Ergebnisse verschiedener Arten ausgeben.

Zur Auswahl bietet die Software über die serielle USB-Leitung, oder jetzt bevorzugt über eine drahtlose Telnet-Verbindung, ein einfaches Befehl-Menü bestehend aus einzelne Zeichen, die über [return] abgeschlossen werden.

Geräteverwaltung

'Z': ESP-Gerät zurücksetzen

'C': 94dB-Kalibrierung anwenden

'c': 47dB-Kalibrierung anwenden /

'U': 94 & 47dB-Standardeinstellungen abrufen

'+'. Offset um 1 dB erhöhen

'-': Offset um 1 dB verringern

'?': Parameter auflisten

'~': Einstellungen für WLAN & Funkparameter auflisten.

Diese Befehle sind beliebig anreihbar:

Sie können mehrere Zeichen eingeben, dann [return] drücken; diese Befehle werden sofort hintereinander ausgeführt:

Beispiel dafür: [U ++](#) bedeutet: „Standardeinstellungen von 94 und 47 dB abrufen und den Offset um 3 dB erhöhen.

Lokaldisplay

Das nutze ich nicht mehr, bei Bedarf könnte es wieder implementiert werden. Es würde ein Display (OLED, oder LCD) benötigen. Thinger ist jetzt deutlich leistungsfähiger und benötigt keine Hardware, die gelötet werden müsste...

'0': // Anzeigemodus 0... '3': // Anzeigemodus 3

Berichte über die serielle USB-Leitung

'A': // AK-Ausgabe (dies ist kein druckbarer Bericht, es wird jede Sekunde ein Byte auf der serielle Schnittstelle ausgegeben, um die DFLD-Website mit Daten zu versorgen)

Normalerweise legt ein Großbuchstabe eine Funktion fest und der Kleinbuchstabe setzt die Funktion zurück.

'P': // Periodische Berichte an 'p': // Periodische Berichte aus

Auswahl für regelmäßige Berichte:

'D': // Tagesbericht an

'd': // Tagesbericht aus

'H': // Stundenbericht an

'h': // Stundenbericht aus

'M': // Minutenbericht an *

'm': // Minutenbericht aus

'S': // Sekundenbericht an *

's': // Sekundenbericht aus

'E': // Ereignisbericht an

'e': // Ereignisbericht aus

* Diese Berichte liefern Arduino-Serial-Plotter kompatible Ergebnisse.

Beispiel 1 :[PDHmsE](#) bedeutet: Berichte an, täglich, stündlich, ohne Minute, ohne Sekunde, mit Ereignissen

Beispiel 2: [p](#) (klein) bedeutet: Berichte aus.

Beispiel 3: [P](#) (groß) bedeutet: Berichte mit den letzten Optionen fortsetzen

Beispiel 4: [Sd](#) bedeutet: Sekundenbericht an, Tagesbericht aus, sonst gelten die letzten Optionen

'J': // Überwachung der Jobdauer 'j': // Rücksetzen des Maximums für die jeweilige Jobdauer

Einmalberichte

'L': // Leq-Bericht bis 24 Uhr

'N': // NAT-Bericht bis 24 Uhr

'B': // Batteriebericht über 24h *

'b': // Batteriebericht (momentane Messungen) *

'W': // Wetterbericht

Diese Berichte unterbrechen periodische Berichte, fahren Sie mit „P“ fort, um zum periodischen Drucken zurückzukehren.

Berichte (Beispiele)

Minute Stunde Tag und Ereignisse

Menübefehl MHEP:

```
08: 55: 59.490 -> Bat_Volt-10: 1.928 Bat_Watt: -0.593 Bat_Level: 7.200
08: 56: 59.461 -> Bat_Volt-10: 1.927 Bat_Watt: -0.593 Bat_Level: 7.200
08: 57: 59.462 -> Bat_Volt-10: 1.926 Bat_Watt: -0.593 Bat_Level: 7.200
08: 58: 59.502 -> Bat_Volt-10: 1.925 Bat_Watt: -0.593 Bat_Level: 7.200
08: 59: 59.489 -> BatAxBat: 0.000 A0dBLeQ: 58.8 Windgeschwindigkeit: 0 Richtung: 1072693248
08: 59: 59.536 -> Bat_Volt-10: 1.924 Bat_Watt: -0.593 Bat_Level: 7.200
09: 00: 59.478 -> Bat_Volt-10: 1.923 Bat_Watt: -0.593 Bat_Level: 7.200
09: 01: 59.466 -> Bat_Volt-10: 1.922 Bat_Watt: -0.593 Bat_Level: 7.200
09: 02: 47.466 -> PKTm: 09:01:37 PKdB: 52.8 ATdB: 50.2 ATsec: 48 PK-10dB: 49.7 PK-10sec: 41 NAT: 1
09: 02: 59.489 -> Bat_Volt-10: 1.921 Bat_Watt: -0.593 Bat_Level: 7.200
09: 03: 59.488 -> Bat_Volt-10: 1.920 Bat_Watt: -0.593 Bat_Level: 7.200
```

Menübefehl L:

Leq: für Samstag, 08. August 2020
Stunde | 00 | 01 | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 10 | 11 |
Leq dB | 58,1 | 58,3 | 58,3 | 58,4 | 58,4 | 58,5 | 58,6 | 58,7 | 58,8 | 61,1 | 65,4 | 65,8 |
Stunde | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 |
Leq dB | 66,0 | 66,2 | 66,3 | 62,9 | 63,2 | 63,6 | 63,8 | 54,0 | 56,3 | 56,6 | 57,3 | 57,8 |

Menübefehl N:

NAT: für Samstag, 08. August 2020
Stunde | 00 | 01 | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 10 | 11 |
NAT | 00 | 00 | 01 | 00 | 01 | 00 | 00 | 00 | 01 | 24 | 23 |
Stunde | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 |
NAT | 20 | 21 | 19 | 06 | 12 | 09 | 34 | 12 | 13 | 01 | 01 | 00 |

Menübefehl B: *

Batterieverlauf:
Stunde | 00 | 01 | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 10 | 11 |
Fledermaus Ah | -0,045 | -0,047 | -0,048 | -0,048 | -0,048 | -0,048 | -0,048 | -0,049 | -0,049 | +0.000 | +0.000 | +0.000 |
Stunde | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 |
Fledermaus Ah | +0.000 | +0.000 | +0.000 | +0.000 | +0.000 | +0.000 | +0.000 | +0.000 | -0,044 | -0,044 | -0,044 |

- Rot bedeutet „mit Batteriemanagement“, siehe Anhänge

Laufende Berichterstattung:

Die Ereignis- und Mitternachtszusammenfassung stellt bereit:

- NAT: (Anzahl der über dem Schwellenwert liegenden Stunden;
- PKTime: (Spitzenzeit des Ereignisses)
- Leq4: (Pegeläquivalent für die Zeit, die durch max-10dB bis max-10dB auf der anderen Seite abgegrenzt ist)
- t₁₀: (Zeit definiert durch max-10dB bis max-10dB auf der anderen Seite)
- Leq3: (Pegeläquivalent für die Zeit die über dem Schwellenwert abgegrenzt ist)
t_s(tAT): (Zeit über dem Schwellenwert)

NAT | PKTime | PKdB | Leq4 | t10 | Leq3 | tAT |

15	21: 52: 29	74,7	70,5	026	70,1	057
01	22: 01: 33	74,6	71,2	017	70,3	043
02	22: 03: 44	71,2	67,3	019	66,7	045
03	22: 13: 46	72,3	69,2	026	68,9	055
04	22: 16: 25	71,8	69,3	021	68,8	048
05	22: 28: 19	69,3	66,7	024	66,7	048

...

Tagesbericht für Dienstag, 15. September 2020

Stunde | Leq | NAT | Ah |

00 | 35,6 | 000 | -0,427 |

01 | 33,2 | 000 | -0,391 |

02 | 28,8 | 000 | -0,385 |

03 | 29,3 | 000 | -0,399 |

04 | 32,3 | 000 | -0,388 |

05 | 38,8 | 000 | -0,379 |

06 | 49,8 | 005 | -0,379 |

07 | 54,1 | 007 | -0,025 |

08 | 53,1 | 006 | -0,290 |

09 | 61,2 | 020 | -0,219 |

10 | 57,2 | 005 | +0,116 |

11 | 61,8 | 012 | +1,050 |

12 | 56,9 | 006 | +1,594 |

13 | 57,4 | 008 | +1,824 |

14 | 59,4 | 010 | +1,392 |

15 | 57,6 | 007 | +1,080 |

16 | 54,2 | 005 | +1,091 |

17 | 54,6 | 007 | +1,023 |

18 | 58,3 | 008 | +0,290 |

19 | 56,2 | 012 | -0,142 |

20 | 54,1 | 005 | -0,494 |

21 | 59,8 | 015 | -0,461 |

22 | 53,9 | 005 | -0,430 |

23 | 31,8 | 000 | -0,407 |

Zusätzliche „Stunden“ vgl. Handbuch.

25 | 31,8 | 000 | -0,407 |

26 | 56,0 | 143 | +4.651 |

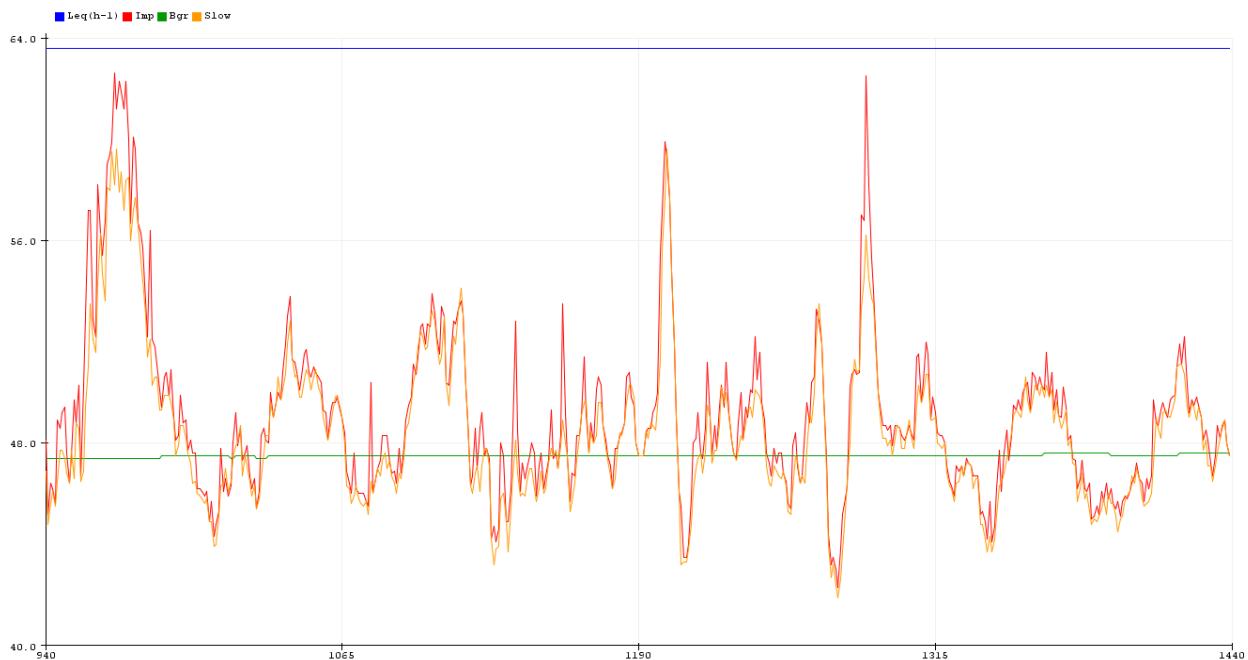
27	57,6	138	+4.651				
28	45,2	005	+0.000				
29	56,9	005	+0.000				
NAT PKTime PKdB Leq4 t10 Leq3 tAT							
01	05: 58: 09	72,0	68,0	019	67,5	044	
01	06: 07: 10	71,7	69,0	028	68,8	059	
01	07: 05: 42	61.1	58.7	018	59.0	031	
...							

Diese Berichte werden standardmäßig über die serielle USB-Verbindung zum seriellen Monitor der Arduino IDE weitergeleitet. Alternativ auch über Telenet.

Grafische Ausgabe (Plotten):

Über den Menübefehl "ESP" (*Nomen est Omen*) können Sie einen Bericht mit Daten im Sekundentakt abrufen, dessen Daten in einem für die serielle Darstellung geeigneten Format vorliegen.

Über den Arduino Serial Plotter können Sie eine grafische Ausgabe des Lärmpegelverlaufs erhalten:



Die Plotter- und Berichtsfunktionen des Arduino IDE sind jedoch begrenzt, und Sie können auch nicht mehrere Plots gleichzeitig abrufen.

Cloud-Service Thinger

Durch die Nutzung der bedingt kostenlosen Cloud-Dienste von <http://thinger.io> können besonders leistungsstarken Berichtsfunktionen erreicht werden :

Jedermann kann dort bis zwei Geräte kostenlos registrieren, um Informationen zu zeichnen und Dashboards auf einer sehr vielseitigen Weise zu entwerfen.

- Schnelle Dashboards, die im Sekundentakt aufbauen.

- Historische Dashboards, in denen Informationen angezeigt werden, die von thinger.io in laufend gesammelt wurden.

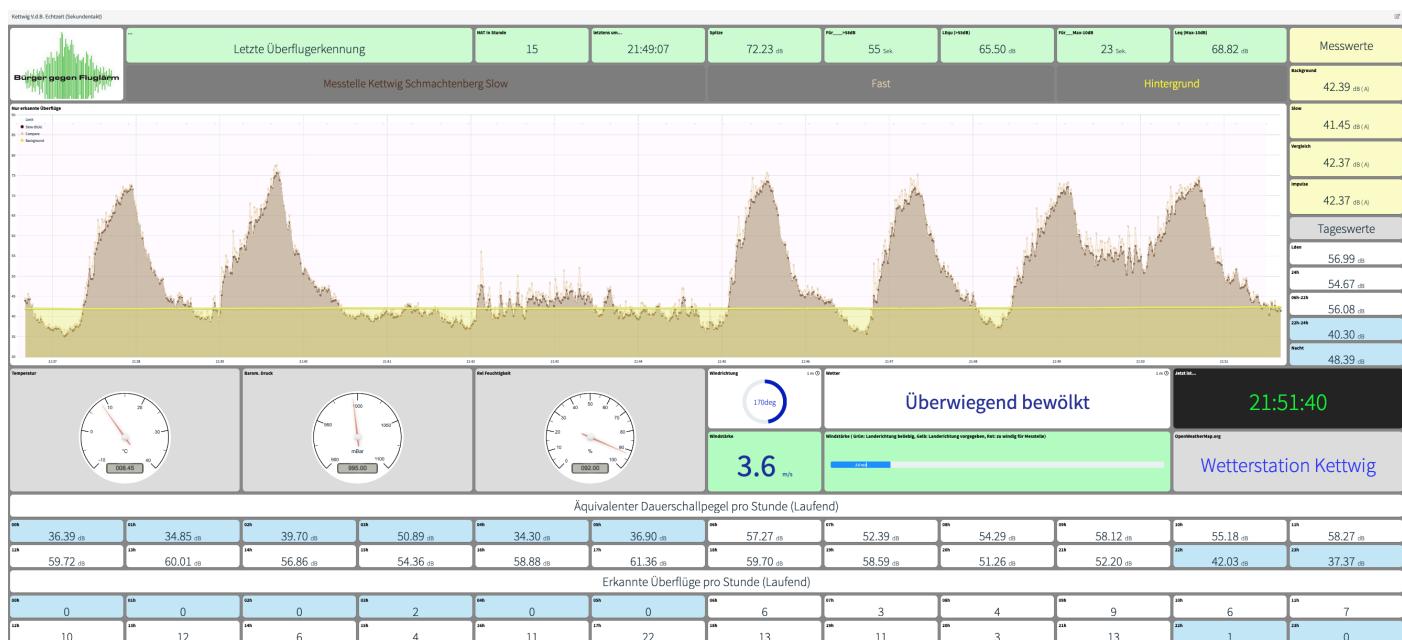
- Berichte, in denen periodische Informationen über einen Zeitraum von bis zu 365 Tagen gesammelt werden. Sie können diese als CSV-Daten in z. B. Excel herunterladen.

- Regeln, um Warnungen über zB E-Mail weiterzuleiten.

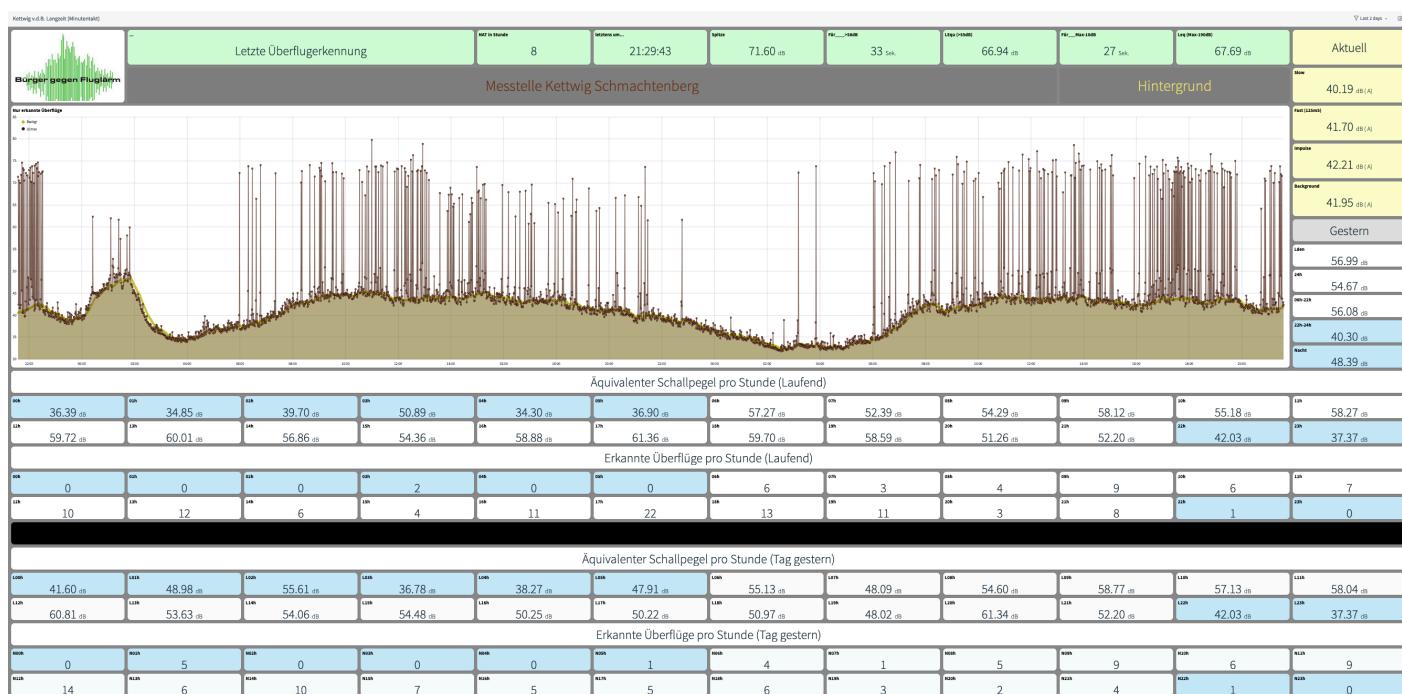
- Mit dem Gerät interagieren, um ein Relais anzusteuern oder die Kalibrierung einzuleiten

- und vieles mehr...

Thinger Second-Lärmplotter und Wetterinformationen



Thinger Minute Lärmplotter



Thinger Ereignisaufstellung

Bucket Data

Date	NAT	Above Thresh Dur...	Above Thresh Leq	Less10d B...	Less10d BLeq	Peak Time	Peak Value
2020-09-15T17:58:53.615Z	6	50	63.759464263916016	25	63.759464263916016	17:57:46	66.83229064941406
2020-09-15T17:55:44.613Z	5	52	60.45946502685547	33	59.72776794433594	17:54:41	63.62342071533203
2020-09-15T17:36:51.593Z	4	70	65.37852478027344	33	65.58905792236328	17:35:39	68.84777069091797
2020-09-15T17:32:05.590Z	3	60	62.27629089355469	31	62.16830825805664	17:30:52	65.78192901611328
2020-09-15T17:25:03.581Z	2	23	60.09807205200195	14	59.51743698120117	17:24:04	62.880531311035156
2020-09-15T17:18:42.575Z	1	45	55.47260284423828	36	54.57136535644531	17:17:42	58.484275817871094
2020-09-15T16:43:27.547Z	17	53	55.736690521240234	44	54.831336975097656	16:42:24	59.37811279296875
2020-09-15T16:29:55.536Z	16	54	58.39207458496094	33	57.83545684814453	16:28:52	60.47865295410156
2020-09-15T16:26:30.533Z	15	94	58.1973762512207	60	57.705810546875	16:25:20	62.396209716796875
2020-09-15T16:25:18.532Z	14	78	58.256988525390625	51	57.75934982299805	16:25:17	62.28064727783203
2020-09-15T16:23:46.532Z	13	105	63.85183334350586	53	63.81764602661133	16:23:30	66.52909088134766
2020-09-15T16:22:25.531Z	12	70	60.97349548339844	42	60.431671142578125	16:21:54	63.992164611816406

Deutscher Fluglärdienst.

Die Software kann auch, zusätzlich zu Thinger und alternativ zur seriellen Ausgabe ein einzelnes Byte pro Sekunde gemäß einem proprietären "AK-Modulbus-Protokoll" an ein Feeder-Programm ausgeben, das z. B. auf einem Raspberry Pi ausgeführt wird und Daten in Stundentakt an das europaweite Fluglärmnetz <http://www.eans.net> weiterleiten.

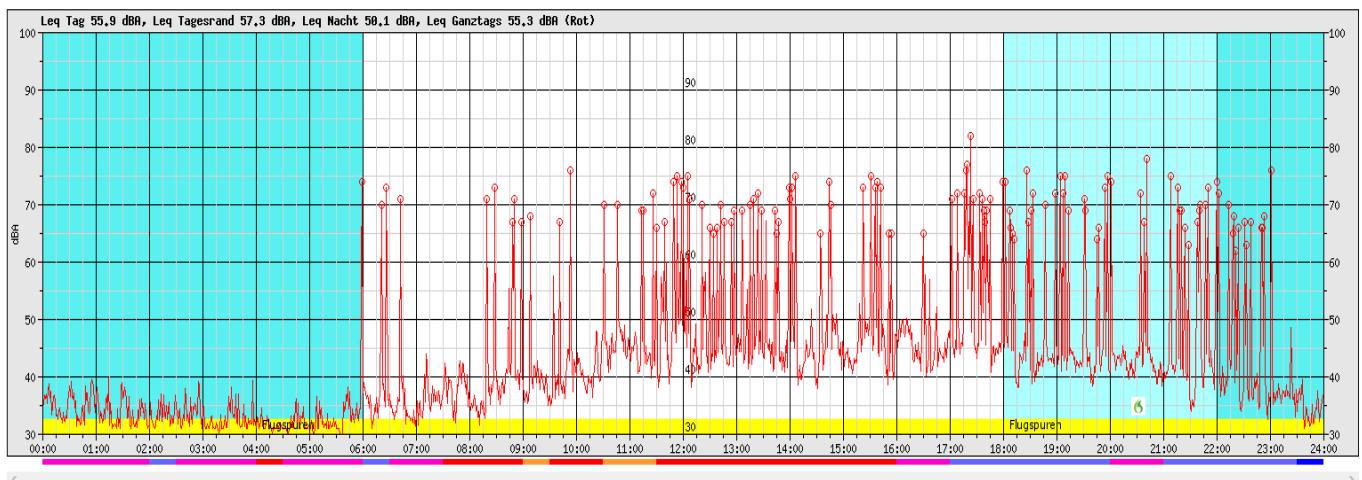
Dieses Netzwerk bietet eine sehr lange (über 10 Jahre) Lobby-unabhängige Speicherung von Flug- / Eisenbahn lärminformationen, die von Anwohnern und Gemeinden verwaltet werden.
Derzeit sind insgesamt rund 700 privat und kommunal betriebene Lärmstationen in ganz Europa verfügbar.
Geräuschaufzeichnung von DFLD / EANS. (Beispiele)

Kettwig/Schmachtenberg

Regions Menü

02.08.2020 (Sonntag)

« 02.08.2020 » - Andere Messstation ▾



Tägliche Statistiken von DFLD / EANS

Anz. erkannter Überflüge

Aufgeschlüsselt nach 5 dB_A großen Maximalpegelbereichen

Zeitraum	Überflüge												Max. [dB _A]	Dauerschallpegel L _{den} ⁽¹⁾	Gesamtschallpegel L _{eq3}		
	<50	50-54	55-59	60-64	65-69	70-74	75-79	80-84	85-89	90-94	95-99	≥100					
Tag	0	0	0	0	4	12	5	0	0	0	0	0	21	77	50.8	56.3	
Tagesrand	0	0	0	0	0	0	0	0	0	0	0	0	0	---	---	39.7	
Nacht	0	0	0	0	0	0	0	0	0	0	0	0	0	---	---	39.4	
Σ	0	0	0	0	4	12	5	0	0	0	0	0	21	77	49.0	49.0	54.6

< >

Jahresstatistik von DFLD / EANS

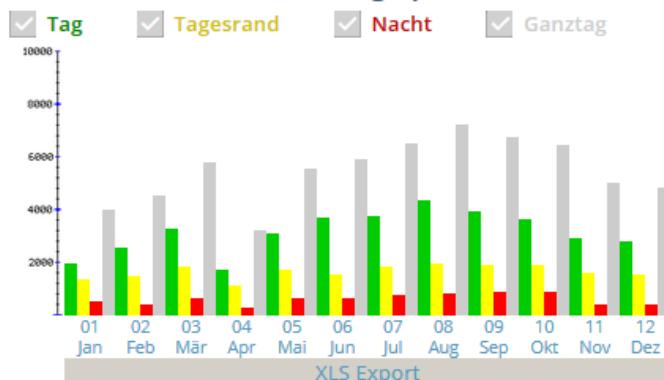
Jahres-Statistik

Kettwig/Schmachtenberg, 2019

Stundenansicht Wochenansicht Monatsansicht 6 verkehrsreichste Monate

« 2019 » Kettwig/Schmachtenberg +

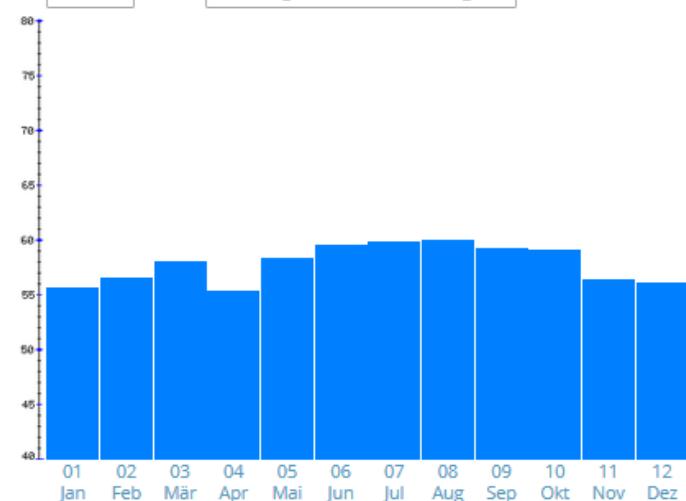
Anzahl erkannter Überflüge pro Monat:



Dauerschallpegel L_{den} (Diagramm):

L_{den} L_{eq3} L_{den} (Nur Überflüge) L_{den} (Gesamtlärm)

« 2019 » Kettwig/Schmachtenberg +



Ein detailliertes Beispiel für die Konfiguration des DFLD / EANS-Gateways (Raspberry Pi) finden Sie in einem separaten Dokument. (in Arbeit)

Hardware

Option 1

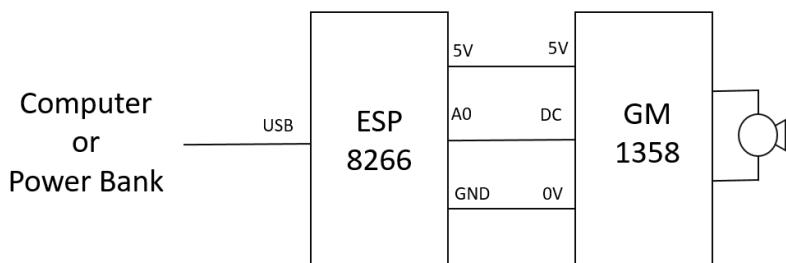
Wenn Sie noch keinen Schallpegelmesser haben, darf ich dieses Modell vorschlagen?
GM 1358 (siehe Stückliste), unglaublich preiswert und ausreichend genau!

Dieses Gerät hat die einzigartige Eigenschaft, einen linearen 0...1-V-Gleichstromsignalausgang bereitzustellen, das an GND gebunden ist und es startet automatisch sobald 5V vorliegt.

Es kann mit einem ESP 8266 mit nur 3 Drähten verbunden werden, was wirklich ein großer Vorteil ist und alles einfacher macht.

Schema

Einfache Version mit GM1358



(Weitere erweiterten Versionen (wetterfest und solar) werden in den Anhängen beschrieben.)

Stückliste

Sie können GM1358 Schallpegelmessgeräte auf eBay.com finden, aber Sie finden die besten Angebote auf AliExpress / Temu zu stark unterschiedlichen Preisen. Alles unter 30 € ist in Ordnung...

RZ GM1358 30-130dB Digital sound level meter meters
A/C FAST/SLOW dB screen New

★★★★★ 5.0 ✓ 2 Reviews 6 orders

€ 16,80 €19,76 -15%

Ships From:
CHINA Russian Federation

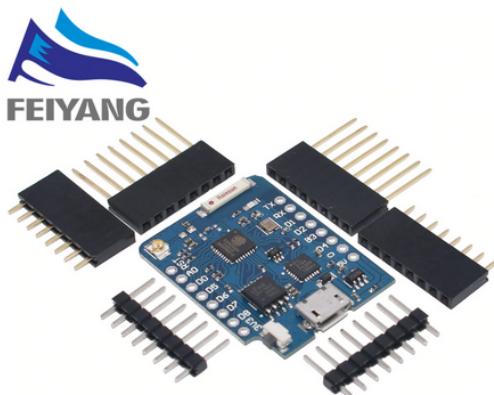
Quantity:
1 Additional 3% off (3 pieces or more)
484 pieces available

Please select the country you want to ship from

Buy Now Add to Cart

Momentan sind sie schwer zu finden und es gab eine massive Preiserhöhung, aber sie gibt es immer noch.

Das ESP8266 dazu finden sie hier am günstigsten:



1PCS WeMos D1 mini - Mini NodeMcu 4M bytes Lua WiFi Internet board based ESP8266 NODEMCU

★★★★★ 4.9 ~ 836 Reviews 1492 orders

€ 1,59 - 2,88 € 1,88 - 3,39 -15%

€ 0,91 off on € 45,41 Get coupons

Color:



Quantity:

- 1 + Additional 2% off (10 pieces or more)
43018 pieces available

Shipping: € 1,44

to Germany via Cainiao Super Economy ~

Estimated Delivery on 09/15 ⓘ

Buy Now

Add to Cart



90-Day Buyer Protection
Money back guarantee

<https://www.aliexpress.com/item/32831353752.html>

Schneller geht es, wenn Sie es in Deutschland bei den üblichen Verdächtigen kaufen (eBay, Amazon ...)

Änderung ESP8266 im GM1358

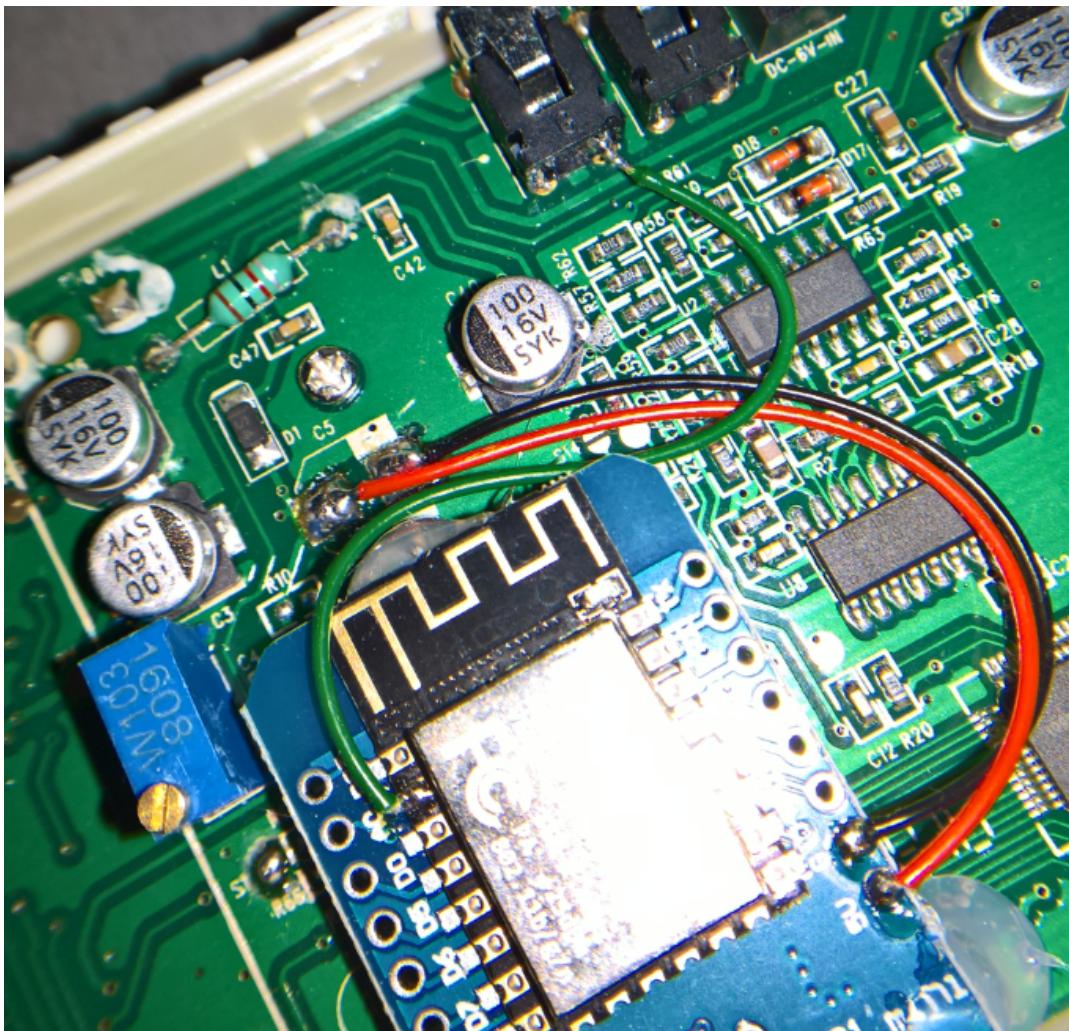
Der GM1358 ist leicht zu zerlegen. Entfernen Sie 4 Schrauben: 2 oben und 2 im Batteriefach, und Sie können das Gehäuse öffnen. Schleifen Sie eine Öffnung in das Kunststoffgehäuse, um Platz für den USB-Stecker zu schaffen.



Kleben Sie den ESP8266 mit einem USB-Anschluss wie gezeigt an, um das Modul zu mit Heißkleber befestigen.

- Löten Sie einen Draht zwischen der 5-V-Klemme und dem schmalen Lötpad „C3“ (siehe nächste Abbildung)
- Löten Sie einen Draht zwischen der Klemme „G“ und dem breiten Lötpad „C3“ (siehe nächste Abbildung)
- Löten Sie einen Draht zwischen dem „Briefmarken“anschluss des ESP-Moduls vor „A0“ und dem Innenkontakt der DC-Buchse.

(*Löten Sie nicht an den Anschluss „A0“ der blaue Baugruppe, die einen 3-V-Spannungsteiler bietet, den wir hier nicht brauchen.*)



Schrauben Sie das Gehäuse zurück. Erledigt!

Energieüberlegungen zum Stromverbrauch

Der GM1358 kann für einen drahtlosen Betrieb den eingebauten ESP8266 mit Strom versorgen.

Der eingebaute 9-V-Alkalibatterieblock versorgt den Schalldruckpegelmessgerät allein für etwa 24-Stunden (bei 11 mA).

Mit dem nachgerüsteten ESP8266 beträgt der Gesamtverbrauch ca. 48 mA, wodurch im SP-Messgerät alleine, der Batterieblock innerhalb von nur ca. 8 Stunden komplett entladen wird.

Glücklicherweise können Sie das System auch über die USB-Buchse des ESP8266 den GM1358 ganz ohne Batterieblock mit Strom versorgen:

- Am Computer über USB angeschlossen
- Autark an eine Power-Bank angeschlossen
(eine 3000mA Lithium Zelle 18360 bietet über zwei Tage Dauerbetrieb)
- Mit geeignetem 5V Spannungswandler an eine 12V Blei-Säure-Batterie, wesentlich mehr.

Hardwareoption 2: Upgrade für eine vorhandene AK-MODUL-BUS-Konverter-Lösung.

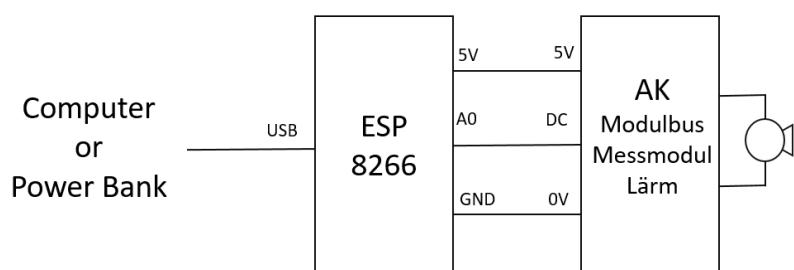


Dies ist eine gute Option für diejenigen von uns, die bereits einen AK-Modulbus-Schallwandler zusammen mit einer dedizierten Schnittstelle db-Online zu DFLD verwenden.

https://www.ak-modul-bus.de/stat/messmodul_laerm.html

Ein ESP8266 kann den dB-Online-Konverter komplett ersetzen und emulieren (+ erheblich verbessern). Damit sparen Sie die dB-Online-Schnittstelle, das Netzteil und den Seriell-USB-Wandler. Und viel Platz und Kabelsalat!

Schema



Stückliste

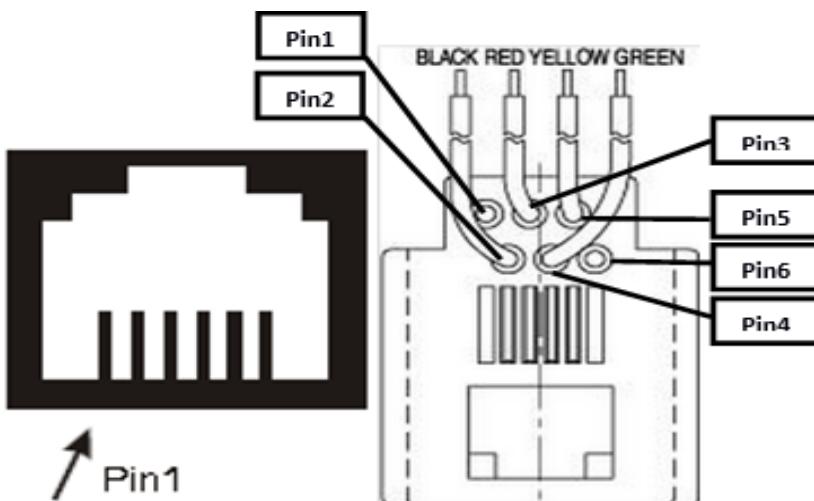
Das ESP8266 finden Sie hier am günstigsten:

1PCS WeMos D1 mini - Mini NodeMCU 4M bytes Lua WiFi Internet board based ESP8266 NODEMCU
★★★★★ 4.9 836 Reviews 1492 orders
€ 1,59 - 2,88 € 1,88 - 3,39 -15%
€ 0,91 off on € 45,41 Get coupons
Color:
Blue, Black, Silver
Quantity:
1 Additional 2% off (10 pieces or more)
43018 pieces available
Shipping: € 1,44
to Germany via Cainiao Super Economy
Estimated Delivery on 09/15
Buy Now Add to Cart
90-Day Buyer Protection
Money back guarantee

<https://www.aliexpress.com/item/32831353752.html>

Schneller geht es, wenn Sie es in Deutschland bei den üblichen Verdächtigen kaufen (eBay, Amazon ...)

Sie benötigen noch eine RJ12-Verlängerung und schneiden davon den RJ12 Stecker ab.



Wenn Sie ein normales deutsches Standardkabel verwenden (normalerweise mit umgekehrter Verkabelung), verdrahten Sie wie folgt:

- roter (brauner) Draht (Pin3) zu A0 *
- grüner Draht (Pin4) nach Masse
- gelber Draht (Pin5) auf +5 V

Schneiden Sie die anderen Drähte ab.

- Sollten Sie die (ungewöhnliche) direkte Verkabelung vorfinden, verdrahten Sie wie folgt:

- grüner Draht (Pin3) zu A0
- schwarzer Draht (Pin4) gegen Masse
- roter Draht zu (Pin5) +5V

* Hinweis:

In diese HW-Variante wird das rote (braune) Kabel an den A0-Anschluss auf der blaue Leiterplatte angelötet. und nicht an das ESP , wie dies bei einem GM1358 der Fall war. (da wir dieses Mal den einge- bauten 3V-1V Spannungsteiler verwenden möchten.)

Verwenden Sie etwas Heißkleber, um die Drähte zu befestigen.

Es steht Ihnen frei, ein Gehäuse in Streichholzschachtelgröße Ihrer Wahl zu verwenden, mit einem Schrumpfschlauch schützen, oder sogar die Baugruppe wie hier frei zu lassen.



Software

Weiter geht es mit den Arbeitsschritten zur Softwareeinrichtung:

Registrierung bei Internetdienste

Bevor Sie an der Konfiguration herangehen, müssen sie sich noch bei den Diensten, die Sie verwenden werden, anmelden und sich die Anmeldedaten merken.

1. Erstellen Sie ein kostenloses Konto bei [www.thinger.io <http://www.thinger.io>](http://www.thinger.io)

Richten Sie Gerät wie folgt ein (Sie können zwei Geräte kostenlos verwalten):

<https://console.thinger.io/#!/console/devices/add>

The screenshot shows a form titled "Device Details". It contains four input fields: "Device Type" with the value "Generic Thinger Device (WiFi, Ethernet, GSM)", "Device Id" with the value "Schallpegel", "Device description" with the value "Mein Schallpegelmessgerät mit überirdischen Funktionen", and "Device credentials" with the value "dY%TH9pHk&l9". Below the fields is a button labeled "Generate Random Credential".

merken Sie sich diese Daten, sowie Ihr Anwendername und Passwort.

2. Richten Sie die folgenden „Daten Buckets“ ein:

Bucket	Name	Description
MIN	Minute recording	Sound level: Min, Slow, Max
EVENT	Event List	Above Threshold, Battery warnings
DAY	Daily report	LDEN, LDaytime, LNighttime, NAT24h, NAT22-24
HOUR	Hourly Data	LEqu, Weather, Battery Normal

3. Erstellen Sie ein kostenloses Konto bei https://home.openweathermap.org/users/sign_up

um Wetterinformationen für Ihren Standort zu erhalten.

Nach der Anmeldung, geben Sie Ihre Wohnort hier ein:



Merken Sie sich die Ortsnummer in der URL: <https://openweathermap.org/city/2928810> , sowie Ihr Anwendername und Passwort.

4. Für bereits Mitglieder des DFLD, merken Sie sich die Region und Stationsnummer.

5. Für den Betrieb benötigen Sie eine WLAN-Verbindung und eine Internetverbindung, aber das haben Sie ja schon...

Einrichtung

Der ESP8266 wird über eine USB-Verbindung mit dem kostenlosen Open-Source-Arduino-IDE programmiert.

Wenn Sie es noch nicht installiert haben, sollten Sie dafür die Arduino-Programmierschnittstelle (IDE) von

www.arduino.cc auf Ihrem Lieblings-PC gemäß Anweisungen herholen:

<https://randomnerdtutorials.com/installing-the-esp32-board-in-arduino-ide-windows-instructions/>

Die aktuellen Dateien meines Programms werden auf GitHub gehostet:

<https://github.com/rin67630/SPL-Booster>

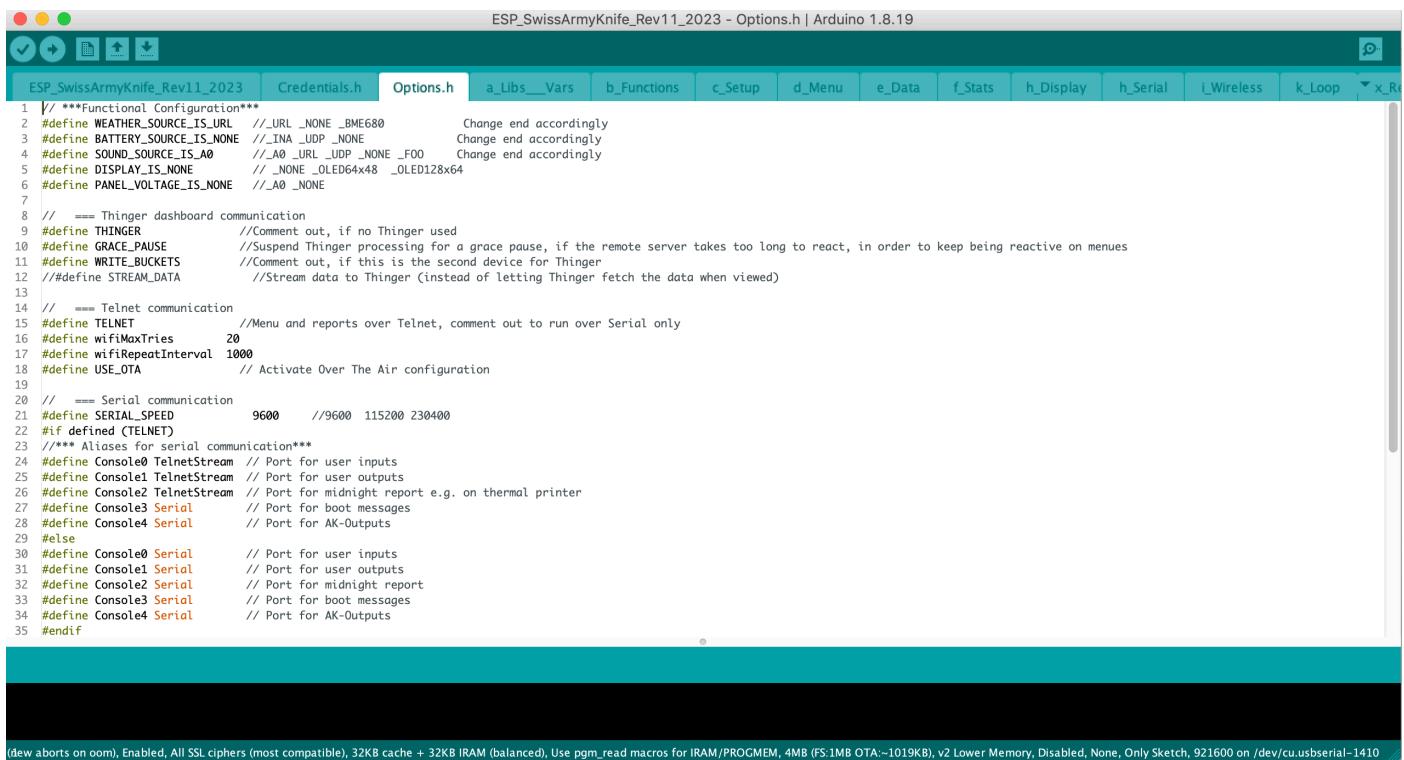
 Code ▾

Sie können alle Dateien mit der Taste

in einer .Zip-Datei auf ihrem Computer herunterladen.

Entpacken Sie die Datei.

Anschließend kopieren Sie den Ordnername von „SPL-Booster-master“ in „ESP_SwissArmyKnife_Rev11_2023“ im Arbeitsbereich des Arduino IDE und doppelklicken Sie auf „EESP_SwissArmyKnife_Rev11_2023.ino“ in diesen Ordner, um das Arduino IDE zu starten:



```
ESP_SwissArmyKnife_Rev11_2023 - Options.h | Arduino 1.8.19
ESP_SwissArmyKnife_Rev11_2023 Credentials.h Options.h a_Libs__Vars b_Functions c_Setup d_Menu e_Data f_Stats h_Display i_Wireless k_Loop x_Resources
1 // ***Functional Configuration***
2 #define WEATHER_SOURCE_IS_URL //URL _NONE _BME680 Change end accordingly
3 #define BATTERY_SOURCE_IS_NONE //INA _UDP _NONE Change end accordingly
4 #define SOUND_SOURCE_IS_A0 //A0 _URL _UDP _NONE _FOO Change end accordingly
5 #define DISPLAY_IS_NONE //NONE _OLED64x48 _OLED128x64
6 #define PANEL_VOLTAGE_IS_NONE //A0 _NONE
7
8 // --- Thing言 dashboard communication
9 #define THINGER //Comment out, if no Thing言 used
10 #define GRACE_PAUSE //Suspend Thing言 processing for a grace pause, if the remote server takes too long to react, in order to keep being reactive on menus
11 #define WRITE_BUCKETS //Comment out, if this is the second device for Thing言
12 //##define STREAM_DATA //Stream data to Thing言 (instead of letting Thing言 fetch the data when viewed)
13
14 // --- Telnet communication
15 #define TELNET //Menu and reports over Telnet, comment out to run over Serial only
16 #define wifiMaxTries 20
17 #define wifiRepeatInterval 1000
18 #define USE_OTA // Activate Over The Air configuration
19
20 // --- Serial communication
21 #define SERIAL_SPEED 9600 //9600 115200 230400
22 #if defined (TELNET)
23 //*** Aliases for serial communication***
24 #define Console0 TelnetStream // Port for user inputs
25 #define Console1 TelnetStream // Port for user outputs
26 #define Console2 TelnetStream // Port for midnight report e.g. on thermal printer
27 #define Console3 Serial // Port for boot messages
28 #define Console4 Serial // Port for AK-Outputs
29 #else
30 #define Console0 Serial // Port for user inputs
31 #define Console1 Serial // Port for user outputs
32 #define Console2 Serial // Port for midnight report
33 #define Console3 Serial // Port for boot messages
34 #define Console4 Serial // Port for AK-Outputs
35 #endif
(dew aborts on oom), Enabled, All SSL ciphers (most compatible), 32KB cache + 32KB IRAM (balanced), Use pgm_read macros for IRAM/PROGMEM, 4MB (FS:1MB OTA:~1019KB), v2 Lower Memory, Disabled, None, Only Sketch, 921600 on /dev/cu.usbserial-1410
```

Die Arduino IDE bietet Registerreiter, mit denen das Programm in gut strukturierte Unterabschnitte aufgeteilt wird.

Als Anwender brauchen Sie sich nur um die Register `Credentials.h` und `Options.h` zu kümmern.

Dort werden Zugangsdaten und Optionen parametert:

Parametrierung

Basisnutzung

-die Software erhält die Wetterinformationen aus dem Internet

-Es gibt kein Batteriemanagement

-Der Eingang des Lärmpegels kommt von einem Schalldruckpegel über den Analogeingang A0.

-Sie verwenden die Thing言-Dienste mit nur einem Gerät

-Alle seriellen Ausgänge werden entweder nicht verwendet oder gehen an die serielle Standardschnittstelle

Sie müssen dafür nur im ersten Teil der Registerkarte "Konfiguration" Anmeldeinformationen und Optionen eingeben.

Hardwarekombinationen (für Fortgeschrittene)

Die Software kann jedoch zusätzliche Betriebsmodi bereitstellen, einige mit zwei oder mehr ESP8266:

- ein ESP8266 ohne zusätzliche Hardware, die Soundinformationen von einer URL von DFLD abruft, um Dashboards anzuzeigen / Statistiken bereitzustellen.
- Split-Betrieb: ZB erfasst ein primärer ESP8266 draußen den Schalldruck und sendet der über WLAN-UDP die Geräuscheinformationen an einen sekundären ESP8266 im Innenbereich.
- Split-Betrieb: ZB ist ein primärer ESP8266 außen am Schalldruckpegelmesser angeschlossen, der mit Solarenergie betrieben wird und über Batteriemanagement-Hardware verfügt.
Der sendet der über WLAN-UDP die Geräuscheinformationen und die Daten der Batterie an einen sekundären ESP8266 im Innenbereich
- Split-Betrieb: ZB gibt es einen ESP8266 für das Batteriemanagement und einen zweiten, der den Schalldruckpegel verarbeitet. Einer von ihnen leitet die Informationen an thinger.io weiter
- Ein primärer ESP8266 verarbeitet das Batteriemanagement und das Schalldruckpegelsignal und leitet die Werte ohne zusätzliche Hardware zum Drucken von Berichten und zur Interaktion mit Thinger an einen zweiten Empfänger ESP8266 weiter.
- Multifunktion: Ein einziger ESP8266 verarbeitet allein das Batteriemanagement und das Schalldruckpegelsignal.
- zwei unabhängige Lärmprozessoren auf demselben Konto. (nicht empfohlen, aber möglich: nur einer sollte auf Eimer schreiben)
- und noch mehr andere Kombinationen von Aufgaben zwischen ESP8266 können eingerichtet werden.

Für diese Betriebsarten, müssen Sie dann im zweiten Teil der Registerkarte Konfiguration Konfigurationsinformationen eingeben.

Aber lasst uns klein anfangen... ☺

Configuration

Zuerst laden Sie das Programm, wie im Kapitel Einrichtung beschrieben.

Um die Optionen zu konfigurieren, werden nun Änderungen an Registerkarte "a0 Parameter" vorgenommen.
(und nur dort! es sei denn, Sie wissen genau, was Sie tun!)

Hier ist ein Beispiel für den Inhalt der Credentials.h-Parameter:

```
// ***Credentials***  
#define WIFI_SSID      "Public_SSID"          //default SSID  
#define WIFI_PASS      "Secret_Password"       //default Password  
#define UDP_TARGET     "192.168.?.?"        // Destination IP for your own Raspberry Pi  
#define HOST_NAME      "Thinger_Device"        // Must match Thinger device name  
#define THINGER_USERNAME "Thinger_User"         // Must match Thinger user name  
#define UDP_TARGET     "192.168.001.001"       // Destination IP for Raspberry Pi (enter real local IP)  
#define UDP_PORT        4321                  // Port to send data, next port to send reports  
#define THINGER_CREDENTIALS "Thinger_Credentials" // Must match Thinger device credentials (Not user password)  
#define DFLD_REGION    "???"                 // Airport region at DFLD  
#define DFLD_STATION   "???"                 // Station number at DFLD  
#define Ao94           747                   // 747 Input AK with offset and 2,5v  
#define Ao47           461                   // 461 Input AK with offset and 2,5v  
// ***Event parameters***  
#define WIND_LIMIT      15                   // upper limit of Wind speed to record NATs  
#define UPPER_LIMIT_DB  88                   // upper limit of Thinger plots  
#define LOWER_LIMIT_DB 31                   // lower limit of Thinger plots  
#define EVENT_THRESHOLD_LEVEL 58             // Begin of Exceedance level  
#define NAT_THRESHOLD_LEVEL 58             // Begin of Exceedance level  
#define MEASUREMENT_THRESHOLD_LEVEL 55        // Begin of measurement level  
#define MIN_EXCEEDANCE_TIME 20              // Minimum duration of an event  
#define MAX_EXCEEDANCE_TIME 80              // Maximum duration of an event  
#define LISTENING_TIME 20                  // mimimum time between events  
  
// ***Time zones***  
#define NTP_SERVER      "de.pool.ntp.org"  
#define MYTZ TZ_Europe_Paris               // (utc+) TZ in hours  
#define TZ 1  
  
// ***Weather server***  
#define OPEN_WEATHER_MAP_APP_ID    "7f02173c19eed016c4797f4ca4251fcc"  
#define OPEN_WEATHER_MAP_LOCATION_ID "2876401"          // Lohausen  
#define OPEN_WEATHER_MAP_LANGUAGE   "de"  
#define OPEN_WEATHER_MAP_UNITS      "metric"
```

Einfache Funktionsweise

Bitte geben Sie die folgenden Informationen und Anmeldeinformationen in Grün ein:

- a) ein frei gewählter Hostname in Ihrem Netzwerk (für Ihr WiFi und zB für Thinger)
- b) Ihre WiFi-Anmeldeinformationen *
- c) die Anmeldeinformationen und Parameter für OpenWeatherMaps
- d) die Anmeldeinformationen und Parameter für Thinger
- e) die Zeitzone
- f) Die Standardkalibrierungsparameter, um Ihren Schalldruck mit dem Analogeingang abzugleichen *
- g) die Event-Parameter für eine Ereigniserkennung (oder belassen Sie die Standardeinstellungen)
- h) Lassen Sie alles unter " // Electrical Parameters" unverändert.

* // ***Acoustical parameters***

Wenn Sie das empfohlene Schalldruckmessgerät GM1368 oder ein anderes verwenden, das einen proportionalen Gleichstromausgang von 10 mV pro dB (Bereich 0..1,0 V) liefert, lassen Sie die Parameter unverändert .

Wenn Sie ein Schalldruckmessgerät verwenden, das einen proportionalen Gleichstromausgang von 2,5 mV pro dB (Bereich 0..2,5 V) liefert, ändern Sie die Parameter in :

```
#define Ao94 712 und #define Ao47 356.
```

Wenn Sie einen Schallwandler von AK Modulbus verwenden, der ein Gleichstromausgang von 25 mV pro dB (Bereich 0..2,5 V) mit Offset liefert, ändern Sie die Parameter in :

```
#define Ao94 747 und #define Ao47 461
```

Wenn Sie ein anderes Schalldruckmessgerät verwenden, das einen proportionalen Gleichstromausgang mit Offset liefert, dann gehen Sie zum Kapitel Hinweise „Parameter für andere Schallpegelmessgeräte einstellen“

Hier ist ein Beispiel für den Inhalt der Options.h-Parameter:

```
// ***Functional Configuration***  
#define WEATHER_SOURCE_IS_URL    // URL _NONE _BME680           Change end accordingly  
#define BATTERY_SOURCE_IS_NONE   // INA _UDP _NONE             Change end accordingly  
#define SOUND_SOURCE_IS_A0       // _A0 _URL _UDP _NONE _FOO  Change end accordingly  
#define DISPLAY_IS_NONE          // _NONE _OLED64x48 _OLED128x64  
#define PANEL_VOLTAGE_IS_NONE    // _A0 _NONE  
  
// === Thinger dashboard communication  
#define THINGER                  //Comment out, if no Thinger used  
#define GRACE_PAUSE               //Suspend Thinger processing for a grace pause, if the remote server takes too long to react, in order  
to keep being reactive on menues  
#define WRITE_BUCKETS            //Comment out, if this is the second device for Thinger  
//#define STREAM_DATA              //Stream data to Thinger (instead of letting Thinger fetch the data when viewed)  
  
// === Telnet communication  
#define TELNET                   //Menu and reports over Telnet, comment out to run over Serial only  
#define wifiMaxTries             20  
#define wifiRepeatInterval        1000  
#define USE_OTA                  // Activate Over The Air configuration  
  
// === Serial communication  
#define SERIAL_SPEED             9600      //9600 115200 230400  
#if defined (TELNET)           //Don't change the grey parameters  
//*** Aliases for serial communication***  
#define Console0 TelnetStream   // Port for user inputs  
#define Console1 TelnetStream   // Port for user outputs  
#define Console2 TelnetStream   // Port for midnight report e.g. on thermal printer  
#define Console3 Serial         // Port for boot messages  
#define Console4 Serial         // Port for AK-Outputs  
#else  
#define Console0 Serial         // Port for user inputs  
#define Console1 Serial         // Port for user outputs  
#define Console2 Serial         // Port for midnight report  
#define Console3 Serial         // Port for boot messages  
#define Console4 Serial         // Port for AK-Outputs  
#endif  
  
// === UDP communication  
#define PUBLISH_REPORT           // Issue events&midnight reports to UDP Port + 100, comment out else  
##define PUBLISH_DFLD            // Issue DFLD byte via UDP Port to another Swiss Army receiver, comment out else  
##define PUBLISH_BATTERY          // Issue Battery information via UDP Port to another Swiss Army receiver, comment out else  
##define PUBLISH_SOUND            // Issue Sound Levels via UDP Port to another Swiss Army receiver, comment out else  
  
// ***Electrical parameters*** to be used if an INA226 monitors the battery parameters.  
##define DEVICES_FOUND INA.begin(10, 25000) //10A Max, 25milliOhm Shunt  
#define SHUNT      25000    // Shunt resistor value in microOhm  
#define AMPERE    10       // 10 or 1  
#define MIN_VOLT  11.0     // 11.0  9.6  
#define MAX_VOLT  14.4     // 14.4  12.8  
#define MIN_AMP   -6  
#define MAX_AMP   +6  
#define MIN_WATT  -40  
#define MAX_WATT  -40
```

Thinger-Konfiguration

Dashboards

Jetzt können Sie bis zu 4 Dashboards nach Ihren Wünschen erstellen.

<https://console.thinger.io/#!/console/dashboards/add>

Ich empfehle 2 Dashboards: REALTIME, MINUTE zu erstellen:

(Die Datenquellen werden in den Anhänge beschrieben)

Daten-Buckets

(entspricht Abbildung Seite 8)

Daten-Buckets sind langfristige dauerhafte Dateien in Thinger, in denen Informationen bis zu einem Jahr gespeichert werden können. Die Einrichtung haben sie ja bei der Anmeldung vorgenommen.

Der Inhalt von Daten-Buckets kann angezeigt werden und nach CSV und damit nach Excel usw. exportiert werden.

Datenquellen für Thinger

Die Software bietet verschiedene Dateninformationen zur Verwendung in Thinger als Widgets.

Das Widget "Time Series" ermöglicht die Anzeige von Plots dieser Werte aus Geräteressourcen oder Geräteeigenschaften.

Diese werden in Echtzeit aufgebaut, wenn Sie das Dashboard aufrufen.

Auf diese Weise können Sie keine Daten anzeigen, die vor dem Aufruf des Dashboards generiert wurden.

Die Aktualisierungsrate kann bis zu 1 Sekunde betragen, aber bitte beachten :

nicht zu viel Daten auf eine Seite setzen, die laufende Aktualisierung verbraucht ziemlich viel Ressourcen...

Sie können als Datenquelle auch Buckets verwenden. In diesem Fall werden die Daten aus dem Verlauf in den Buckets abgerufen und sofort angezeigt. Die Aktualisierungsrate muss ≥ 1 Minute betragen.

Andere Widgets zeigen Werte numerisch, oder als Balkendiagramme, in Tachoform usw.

Geräteressourcen

Geräteressourcen sind eine weitere Möglichkeit Daten für Widgets in einem Dashboard bereitzustellen.

Die Daten können entweder über das Dashboard abgefragt werden, wenn das Dashboard auf dem Bildschirm angezeigt wird,

Folgende Ressourcen stehen zur Verfügung:

Noise

- Min, Langsam, Max, Hintergrund, Schnell, Impuls, Limit,
jede Sekunde weitergeleitet

Energy

- Spannung, Strom, Leistung, Innenwiderstand, Prozentsatz der Ladung,
jede Minute weitergeleitet

DAY

- Ah vier die aktuelle Stunde, Spannung um vier Uhr, Spannungsdelta von einem Tag zum folgenden, gemessen um vier Uhr, Leq-Werte für jede Stunde des Tages, NAT für jede Stunde des Tages, Batterie Ah für jede Stunde des Tages. am Ende des Tages
weitergeleitet

HOUR

- Temperatur, Luftfeuchtigkeit, Luftdruck, Windgeschwindigkeit, Windrichtung, ob Beschreibung, Batteriestrom,
Batteriespannung,
am Ende der Stunde weitergeleitet

MIN

- Lärm min, max, Hintergrund, leq, Batteriespannung, Stromversorgung zur / von der Batterie
jede Minute weitergeleitet

EVENT

- Ereignisspitzenzeit, Spitzenwert, über dem Schwellenwert LE, über dem Schwellenwert LEQ, über der Schwellendauer, weniger 10 dBLE, weniger 10 dBBL, weniger 10 Dauer, NAT-Weiterleitung, wenn ein Ereignis erkannt wurde.

Geräteeigenschaften

Geräteeigenschaften sind eine zusätzliche Möglichkeit, Widgets in einem Dashboard Informationen bereitzustellen. Die Eigenschaften werden alle 6 Minuten von der Software aktualisiert. Die Eigenschaften werden bei der Initialisierung des ESP 8266 auch von Thinger zurückgelesen, zum Beispiel wenn das ESP zurückgesetzt wird, um dort weiter zu machen wo wir zuvor standen.

Folgende Ressourcen stehen zur Verfügung:

Nat

Objekt {00h: 0, 01h: 0, 02h: 0, 03h: 0, 04h: 0, 05h: 0, 06h: 16, 07h: 10, 08h: 9, 09h: 2, 10h: 10, 11h: 6, 12h: 0, 13h: 6, 14h: 8, 15h: 5, 16h: 5, 17h: 8, 18h: 10, 19h: 8, 20h: 6, 21h: 3, 22h: 0, 23h: 1, Tag: 90, Tag: 89, NAT22-24: 1, Nacht: 1}

lequ

Objekt {00h bis 23h, Tag, L-Tag, EV10, EvOT, L22-24h, Lden, Leq, L-Nacht}

BAT

Objekt {Ah Batterie stundenweise von 00h bis 23h}

Persistenz

(Der Zweck der Persistenz der Geräteressourcen besteht hauptsächlich darin, die Programmkontinuität über Neustarts hinweg sicherzustellen: Die Daten sind nicht in erster Linie für die Verwendung in einem Dashboard vorgesehen, können es aber dennoch. Die Liste wird nur zur Info angegeben.)

- A047, (ADC parameter)
- A094, (ADC parameter)
- A0dBgr,
- A0dBSumExp,
- Ah / Stunde,
- Ah / gestern,
- ThreshLEint,
- corrdB: -2,
- currentInt,
- Windrichtung,
- Luftfeuchtigkeit,
- last_update,
- less10dBLEint,
- nCurrent,
- Luftdruck,
- Batterie-internes Widerstand
- Wetterzusammenfassung,
- Außentemperatur,
- Batteriespannung um4h,
- Batteriespannungsdifferenz zum Vortag,
- Windgeschwindigkeit.

Inhaltsverzeichnis

Zusammenfassung	2
Hardware	3
Berichte.....	4
Geräteverwaltung.....	4
Lokaldisplay.....	4
Berichte über die serielle USB-Leitung	4
Einmalberichte.....	4
Berichte (Beispiele).....	5
Minute Stunde Tag und Ereignisse	5
Laufende Berichterstattung:.....	5
Grafische Ausgabe (Plotten):.....	6
Cloud-Service Thinger	7
Thinger Second-Lärmplotter und Wetterinformationen	7
Thinger Minute Lärmplotter	7
Thinger Ereignisaufstellung	8
Hardware, option 1.....	8
Schema	10
Einfache Version mit GM1358	10
Stückliste.....	10
Änderung ESP8266 im GM1358.....	11
Energieüberlegungen zum Stromverbrauch	12
Hardwareoption 2: Upgrade für eine vorhandene AK-MODUL-BUS-Konverter-Lösung.....	13
Stückliste.....	13
Software	15
Registrierung bei Internetdienste.....	15
Einrichtung.....	16
Parametrierung.....	16
Basisnutzung	16
Hardwarekombinationen (für Fortgeschrittene)	17
Configuration	17
Einfache Funktionsweise	18
Dashboards	19
Datenquellen für Thinger.....	19
Noise	19
Energy	19
DAY	19
HOUR	19
MIN	19
EVENT	19

Geräteeigenschaften	20
Nat	20
lequ	20
BAT	20
Persistenz	20
Anhänge	23
Kalibrierungsprozedur	23
Einstellungshilfe ohne Kosten	23
Smartconfig	24
Erweiterte Betriebsarten (für Fortgeschrittene)	25
Deutscher Fluglärmdienst	25
Geräuschaufzeichnung von DFLD / EANS. (Beispiele)	8
Tägliche Statistiken von DFLD / EANS	9
Jahresstatistik von DFLD / EANS	9
Hardwareverbesserungen	26
Hardware Nr. 1a: GM1358 mit Außenmikrofon	26
Solarbetriebene Version (für Fortgeschrittene)	29
Betrieb mit 2 Serielle Schnittstellen	29
Netzwerk-protokollierung	29
Autarker Betrieb mit Thermobondrucker	30
Verwendung anderer SPL-Geräte	30
Normen und Vorschriften zum Verkehrslärm	31
Verwendete Lärm Metriken	31
Leq4-Metrik	31
Leq3-Metrik	32
LAE-Metrik	32
LE-Metrik	32
Tag-Nacht-Durchschnittsschallpegel (Lden)	32
Ereignisaus- und -bewertung	33
Bewertung:	33
Auswertung	33

Anhänge

Kalibrierungsprozedur

Am besten nutzen Sie einen Kalibrator, der ein 94dB Referenzpegel abliefer.

Betreiben Sie das ESP8266 an der serielle Schnittstelle und starten Sie den Seriellen Monitor.

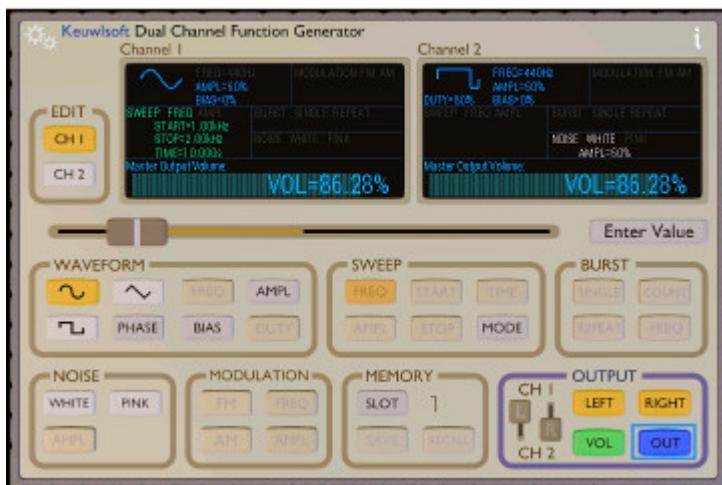
Stecken sie das Mikrofon ins Kalibrator und geben Sie „C“ (Großbuchstabe) als Kommando ein.

Das System antwortet mit Cal94=xxx und gibt den ADC-Wert dazu an.

Einstellungshilfe ohne Kosten

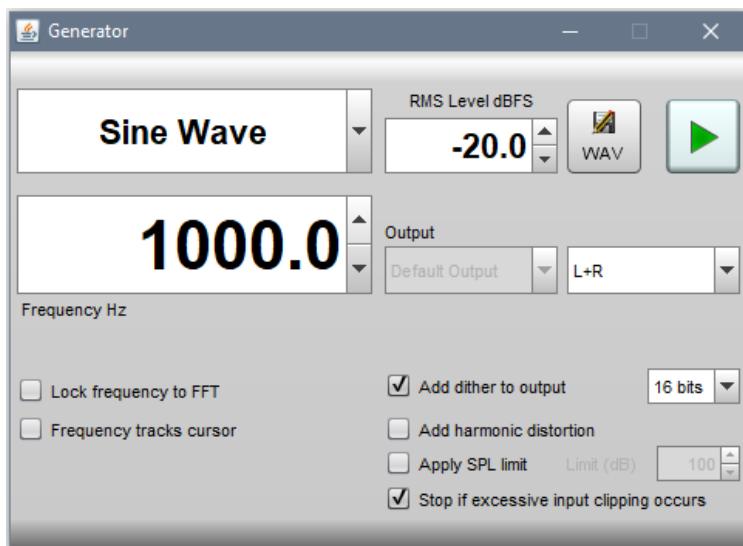
Sie brauchen (leihweise) einen Referenzschallpegelmessgerät und einen Android Smartphone :

Laden Sie den kostenlosen „Kewlsoft Function Generator“ herunter: <http://keuwl.com/FunctionGenerator/>



Alternativ können Sie am Windows-PC das Programm Room-Equ-Wizard (REW) installieren:

<https://www.roomeqwizard.com/> und von diesem Programm den Signal Generator verwenden:



Der Signal Generator von REW hat gegenüber die Smartphone-Lösung den Vorteil, dass Sie die Lautstärke direkt in dB Schritte verändern.

Das hat sich als sehr praktisch erwiesen: eine Pegelabweichung kann einfach durch die gleiche Abweichung im Feld RMS-Level dBFS addiert substanziert werden und der neue Versuch klappt sofort!

Stellen in einem sehr ruhigen Raum das Referenzschallpegelmessgerät und das einzustellendes Mikrofon mit gleichem geringem Abstand (ca. 5cm) zum Lautsprecher und stellen Sie das Generator auf Sinus 1000 Hz so laut, dass das Referenzschallpegelmessgerät 94dB anzeigt.

Justieren Sie Ihr Schallpegelmessgerät entsprechend.

Dann führen Sie die Kalibrierungsprozedur am Serial Monitor bei 94dB durch.

Wiederholen Sie die Prozedur mit dem geringeren Pegel von 47dB und die Kalibrierungsprozedur mit „c“ (Kleinbuchstabe) als Kommando ein.

Das System antwortet mit Cal47=xxx und gibt den ADC-Wert dazu an.

Parameter für andere Schallpegelmessgeräte einstellen

Wenn Sie ein anderes Schalldruckmessgerät verwenden, das einen proportionalen Gleichstromausgang zwischen 0...3 V mit Offset liefert, müssen Sie herausfinden, welche ADC-Werte den Schalldruckmessungen bei 94 dB und 47 dB entsprechen.

Dafür die Prozedur „Einstellungshilfe ohne Kosten“ durchführen und die 2 ADC-Werte (94db und 47dB) notieren und entsprechend in den Programmparametern anpassen:

```
// ***Acoustical parameters***  
#define Ao94 1050 // 752 for AK with offset 0,5..2,5v, 712 for 0..2,5V without offset, 1050 for linear 0..1V  
#define Ao47 550 // 461 for AK with offset 0.5..2,5v, 356 for 0..2,5V without offset, 550 for linear 0..1V
```

Smartconfig

Der Code bietet auch die Möglichkeit, die Software drahtlos hochzuladen und zu warten:

http://arduino.esp8266.com/Arduino/versions/2.0.0/doc/ota_updates.html#arduino-ide

Sie können Smartconfig verwenden https://docs.espressif.com/projects/esp-idf/en/latest/esp32/api-reference/network/esp_smartconfig.html um Ihre WiFi-Anmeldeinformationen einzugeben.

Dies ist die vielseitigste Methode und ermöglicht die Anpassung an verschiedene WiFi-Netzwerke, ohne sie neu zu kompilieren. Sie müssen jedoch eine App auf Ihrem Smartphone ausführen.

Alternativ können Sie Smartconfig auskommentieren und Ihre Anmeldeinformationen direkt eingeben:

```
// SMARTCONFIG definieren // (WiFi-Anmeldeinformationen über Smartphone-App SmartConfig definiert)  
  
// Alternativ zur Smartconfig App können Sie die Zeile Smartconfig auskommentieren  
// und geben Sie Ihre Anmeldeinformationen ein, um ein neues WLAN zu aktivieren:  
#define WIFI_SSID "SSID"  
#define WIFI_PASS "Passwort"
```

Erweiterte Betriebsarten (für Fortgeschrittene)

Die Parameter für erweiterte Betriebsmodi sind nur erforderlich, wenn Sie mit zusätzlicher Hardware arbeiten möchten oder bereits eine Lärmstation bei DFLD ausgeführt wird und einen ESP8266 als zusätzliches Berichtsgerät verwenden möchten.

Sie können konfigurieren:

- a) Ändern Sie das Ende entsprechend, ob Sie Wetterinformationen verwenden:
 _URL, um die Informationen von <http://openweathermap.org> _NONE zu erhalten, wenn Sie die Wetterinformationen nicht benötigen.
- b) Konfigurieren Sie die Quelle der Batterieinformationen und ändern Sie das Ende entsprechend:
 _NONE, wenn Sie keine Batterieinformationen haben.
 _INA, wenn Sie ein INA226-Modul zum Messen von Strom und Spannung haben
 _UDP, wenn Sie im selben Netzwerk einen anderen ESP8266 haben, der dieselbe Skizze wie ein Master ausführt (// # PUBLISH_BATTERY definieren)
- c) Konfigurieren Sie die Quelle der Soundinformationen
 _NONE, wenn Sie keine Batterieinformationen haben_A0, wenn der Schallpegel vom Analogeingang A0_UDP kommt, wenn Sie im selben Netzwerk einen anderen ESP8266 haben, der dieselbe Skizze wie ein Master ausführt (// # definiere PUBLISH_SOUND)
 _URL, wenn der Schallpegel von einer JSON-URL von stammt [www.dfld.de <http://www.dfld.de>](http://www.dfld.de)
(Dann müssen Sie unten die Regions- / Stationsparameter der DFLD-Station angeben, die die Toninformationen liefern.)

Wenn Sie _INA als Quelle für Batterieinformationen definiert haben, sollten Sie die Parameter unter optionalen elektrischen Parametern für das Batteriemanagement überprüfen.

Sie sollten den Wert des aktuellen Shunts Ihres Breakouts anpassen und die Messbereiche entsprechend Ihrer Batterie definieren.

Wenn Sie _URL als Quelle für Toninformationen definiert haben, sollten Sie die Region und die Stationsnummer der DFLD-Station eingeben, auf die Sie sich beziehen.

Die Zeile #define THINGER bestimmt, ob Sie die Cloud-Dienste von thinger.io verwenden. Wenn Sie dies nicht tun und nur serielle Berichte verwenden, entfernen Sie bitte den Kommentar zu dieser Zeile.

Die Zeile #define WRITE_BUCKETS bestimmt, ob dieses Gerät die Buckets schreiben darf. Wenn zwei Geräte an Thingr.io berichten, sollte nur ein Master-Gerät die Buckets schreiben und diese Zeile auskommentieren.

Die Zeilen mit #define CONSOLEx bestimmen, wo einige Teile der Skizze ausgegeben werden. Normalerweise werden Berichte an Serial ausgegeben, mit zusätzlichen UARTS-Ports. Sie können beispielsweise Serial1 für die Ausgabe verwenden. Serial1 kann beispielsweise für die gedruckte Protokollierung mit einem Thermodrucker verbunden werden.

Wenn Sie einen Netzwerkcomputer haben, z. B. als Gateway zu DFLD, sollten Sie PUBLISH_DFLD auskommentieren. Sie können auch PUBLISH_REPORT verwenden, um die Ereignisse und den Mitternachtsbericht abzurufen, die an diesen Computer ausgegeben wurden. Anschließend müssen Sie die Parameter UDP-TARGET und UDP-Port entsprechend eingeben.

Wenn Sie eine geteilte Konfiguration verwenden, geben Sie PUBLISH_BATTERY und / oder PUBLISH_SOUND auf dem primären Gerät ein, definieren Sie UDP-TARGET als IP-Adresse des Ziel-ESP8266 und stimmen Sie den UDP-Port zwischen beiden Geräten ab. PUBLISH_BATTERY und / oder PUBLISH_SOUND können nicht zugeordnet werden zu einem anderen ESP-Gerät und PUBLISH_REPORT gleichzeitig zu einem Computer.

Hardwareverbesserungen

Hardware Nr. 1a: GM1358 mit Außenmikrofon.

Wenn Sie den GM1358 24/24/365 zur Messung des Schalldrucks im Freien verwenden möchten, empfehle ich die folgende Nachbesserung am GM1358 zur Outdoor-Befähigung.

Sie benötigen eine elektrische DN20 / DN25-Kabelverschraubung, ein DN40-Abflussrohr und ein Abschlussstopfen, einen 12-mm-Edelstahl-Garnelenfilter für Aquarienliebhaber, ein Stück 5-mm-Acrylschlauch, eine Edelstahl-Topfreiniger und zwei gut abgeschirmte Koax-Kabeln mit BNC, XLR oder Jack Anschlüssen (ein kurzes und einen in der Länge zwischen den Aufstellungsort des Mikrofons und ein trockenes Ort für den GM1358).

Ich bevorzuge BNC-Steckverbinder, da sie eine gute Qualität bieten, und Sie auf dem absterbenden Markt für Analogvideo viele günstige Kabeln vorfinden. Sie können aber auch Audio-Buchsen und Kabeln in Bühnenqualität verwenden (Thomann)...



Sie löten zuerst die Mikrofonstange vom Messgerät ab und löten an der Stelle die Buchse ein.

Auf der Mikrofonseite führen Sie den 5-mm-Acrylschlauch über das kurze Kabel, verbinden und isolieren den heißen Draht mit dem roten Mikrofondraht und löten einen Teil des Geflechts mit dem schwarzen Mikrofonkabel (Sie sollten ihn nicht isolieren) und lassen Sie die anderen

Teile des Geflechts entlang des Acrylschlauchs laufen, schieben Sie dann alles in den Mikrofonausleger und kleben Sie alles heiß zusammen.

Das nicht isolierte Geflecht verbindet dann die Stangen mit der Masse, was vorteilhaft ist.

Sie haben jetzt einen abnehmbaren Mikrofonarm für die lokale Kalibrierung, das lange Kabel verlegen Sie dann fest als Verlängerung zum Aufstellungsort des Mikrofons.



Der nächste Schritt besteht darin, das Mikrofon für den Außenbereich zu härten:

Das Mikrofon muss vor Regen, Wind, Frost und Schnee geschützt werden:

Es ist keine Option, es unter ein Dach zu stellen, zwischen Mikrofon und Himmel darf kein Schall-abschirmendes Hindernis sein.

Ein einfacher Garnelenfilter für die Aquaristik kann diese Aufgabe erstaunlich gut erledigen.

Sie ziehen es über die Mikrofonstange: Wasser wird durch Kapillarität in das Edelstahlnetz aufgefangen und die Mikrofonkapsel bleibt trocken.

Es funktioniert: ich habe das an einem regnerischen Novembermonat mit einem mit Salz gefüllten Blindrohr getestet. Nach 30 Tagen draußen war das Salz nicht geschmolzen!

Jetzt müssen Sie auch noch Windgeräusche reduzieren.

Ich empfehle nicht die normalen Windkappen aus Gummischäum, wie sie in der Akustik verwendet werden: Sie würden zwar passen: die Außenabmessung des Filters beträgt ca. 25 mm. Sie würden jedoch nicht sehr lange halten:

Die UV-Strahlung zersetzt sie innerhalb von wenige Wochen.

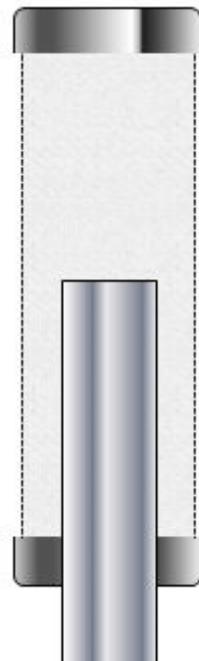
Die Akustiker-Industrie stört das nicht: sie verdient prächtig mit Serviceverträgen, um alle zwei Wochen die Hauben zu ersetzen. ;-)

Wir werden es besser machen und eine völlig wartungsfreie Lösung verwenden: Sie ziehen einfach einen haushaltüblichen Edelstahl- Topfreiniger über den Aquarienfilter!

Es wird ewig halten und keinen Regenwasser einfangen: Das Wasser läuft einfach durch.

Ich habe die Lösung in einer Windkammer an der Universität Düsseldorf getestet: bei 10 m/s betrug das Windgeräusch 56dB: das war genauso gut, wie die Indoor-Windkappe von Brüel & Kjaer!

Der Topfreiniger erfüllt vollständig die DIN Norm, unter 60 dB Windgeräusch bei 10 m/s zu erzeugen!



Der letzte Punkt besteht nun darin, einen Halt für das Mikrofon zu finden und den Stecker vor Wasser und Staub zu schützen. Eine der billigsten und effizientesten Möglichkeiten besteht darin, ein Stück DN40 Abwasserrohr und den entsprechenden Stopfen zu verwenden.

Bohren Sie ein Loch mit einem Durchmesser von 25 mm im Stopfen, Führen Sie die Mikrofonstange durch die Kabelverschraubung, schrauben Sie ihn in den Stopfen, ziehen Sie alles (einschließlich der anderen Seite der Mikrofonkabelverlängerung) zusammen, z. B. in den Halter einer Gartensonnenstuhl, oder befestigen Sie das Rohr an einer Stange im Garten. Das Mikrofon sollte zugänglich sein!

Sie werden es zu schätzen wissen, dass Sie das Mikrofon ohne Werkzeug aussstecken können, um es von Zeit zu Zeit

in Innenräumen neu zu kalibrieren, wobei die Verlängerung des Mikrofonkabels an Ort und Stelle bleibt.

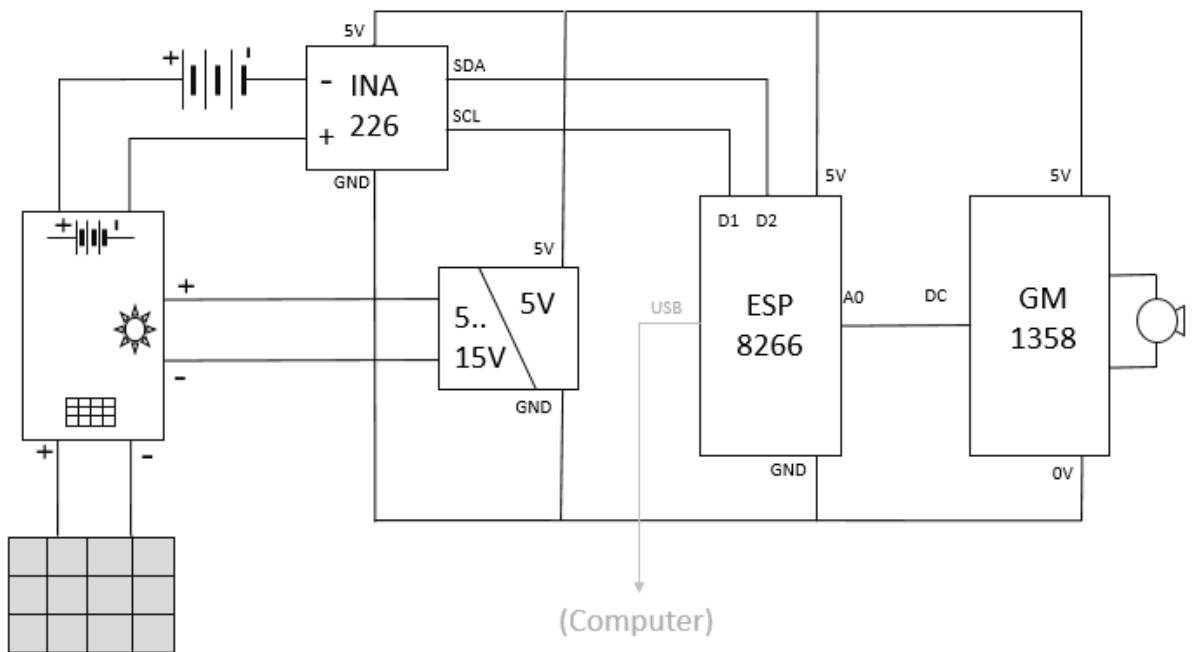


So sieht es dann aus!
Dieses Außenmikrofon trotzt den
Wetterbedingungen seit Jahren ohne
Wartung.

Mit einem gut abgeschirmten Audiokabel
können Sie das Mikrofon bis zu 6 Meter
vom Schalldruckmessgerät entfernt betreiben.
Dies sollte ausreichen, um das SPL-Gerät
wettergeschützt unter einem Dach zu
platzieren. Für eine längere Kabellänge
habe ich auch eine Lösung mit einem
Mikrofonvorverstärker, die jedoch in einem
anderen Dokument beschrieben wird, da
sie etwas komplexer ist.

Solarbetriebene Version (für Fortgeschrittene)

Wenn Sie einen netzunabhängigen 24/24/365-Betrieb mit Solarenergie wünschen, benötigen Sie je nach Standort (z. B. bei 45 ° Breite) eine 12-V-30-Ah-Batterie und (mindestens) ein 20-W-Solarpanel, da Sie sich kaum vorstellen können, wie wenig Energie ein Solarpanel an einem regnerischen Wintertag noch liefert!



Sobald Sie einen solarbetriebenen netzunabhängigen Betrieb in Betracht ziehen, müssen Sie aber auch die Ausrüstung härten, um mit den rauen Wetterbedingungen im Freien fertig zu werden: Regen, Hitze, Frost, Wind, Eis...

Ich werde eine separate Beschreibung für diese Variante bereitstellen, die deutlich mehr Handwerkerfertigkeiten erfordert und deren Beschreibung hier zu viel wäre.

Betrieb mit 2 serielle Schnittstellen

Die Berichte können von Pin D4 (GPIO2) des ESP über einen UART-USB-Konverter an einen anderen USB-Port eines Host-Computers zur Berichterstellung gleichzeitig zum DFLD-Betrieb auf dem USB-Port weitergeleitet werden.

Netzwerk-protokollierung

Die Berichte können an einen UDP-Netzwerk Port ausgegeben werden und in einer Protokolldatei auf dem Host-Computer gespeichert werden.

Autarker Betrieb mit Thermobondrucker

An Pin D4 (GPIO2) des ESP kann ein Thermobondrucker wie dieser angeschlossen werden:



Der Drucker druckt die Berichte autark aus, um somit ein System ganz ohne Rechner zu bilden.

Man kann es alleine laufen lassen und periodisch die gedruckten Ergebnisse (Überflüge und Tageszusammenfassungen) abholen.

Dafür ist es aber besser, den ESP8266 außerhalb des Messgeräts zu bauen: der Drucker braucht, wenn er druckt, kurzzeitig relativ viel Strom. Eine Stromversorgung mit 2A Spitzenleistung ist erforderlich.

Verwendung anderer SPL-Geräte

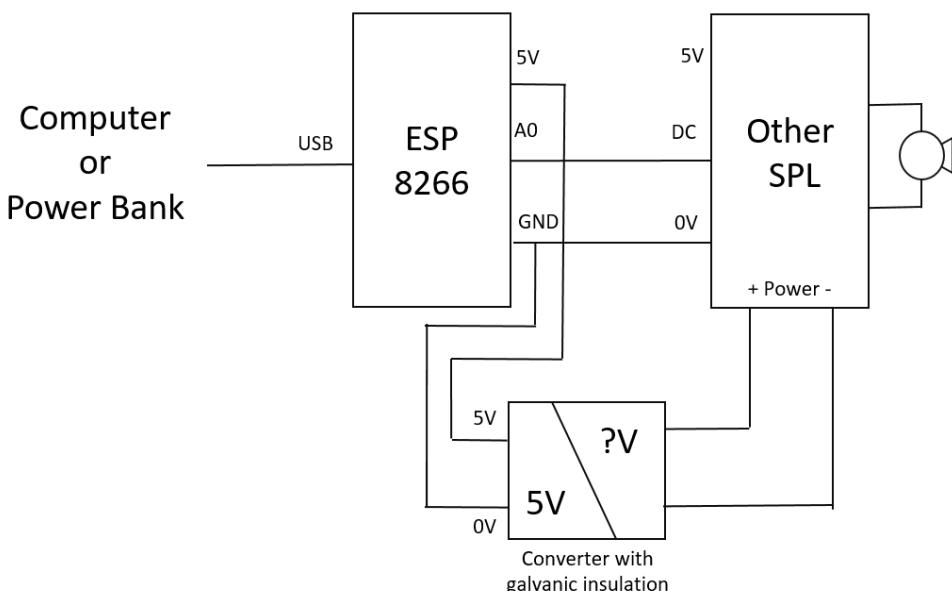
Sie können auch andere SPL-Geräte verwenden, von denen einige von höherer Qualität sind, bis hin zu Geräten der Klasse 1.

Die einzige Voraussetzung ist, dass sie einen Gleichstromausgang mit einem dB-Proportionalwert von 0..1 V oder 0..2,5 V liefern, was den Branchenstandard entspricht.

Über ein Widerstand von 80kΩ kann auch das LAF(Inst.): 0 to 4 V DC Signal von Brüel&Kjaer Messgeräte verwendet werden.

Der Nachteil ist, dass üblicherweise der Gleichstromausgang dieser Geräte in Bezug auf die Stromversorgung versetzt ist und Sie daher eine galvanische Trennung zwischen die Stromversorgung des Schallpegelmessgeräts und den Analogeingang des ESP benötigen.

Die meisten Schalldruckmesser starten auch nicht automatisch, wenn sie mit Strom versorgt werden, sondern müssen manuell über eine Ein- / Aus-Taste gestartet werden. Das macht alles etwas komplizierter und unhandlicher.



Normen und Vorschriften zum Verkehrslärm

Verwendete Lärm Metriken.

Das Programm zielt darauf ab, das internationale Abkommen der ECAC (European Civil Aviation Conference), Band 1 einzuhalten:

<https://www.ecac-ceac.org/documents/10189/51566/01.+Doc29+4th+Edition+Volume+1.pdf/bfde6e09-b46b-44e1-b73f-388fc3527aaaf>

Es werden folgende Metriken abgebildet:

Äquivalenter Schallpegel (LEQ)

Diese Metrik misst die durchschnittliche akustische Energie über einen bestimmten Zeitraum, um den kumulativen Effekt mehrerer Geräuschereignisse zu berücksichtigen. Dies könnte zum Beispiel ein Maß für den Gesamtschall an einem Ort liefern, an dem den ganzen Tag über Flugzeugüberflüge stattfinden. LEQ ist definiert als der Pegel des Dauerschalls über einen bestimmten Zeitraum, der die gleiche Energiemenge liefern würde wie die tatsächliche, variierende Schallbelastung.

Die Leq-Metriken und alle Metriken, die bei der Berechnung des Verkehrslärms verwendet werden, sind keine einfachen Durchschnittswerte, sondern das Ergebnis einer komplexen Berechnung:

Der Schallpegel wird zuerst deelogarithmiert (Exponentialfunktion), um Dezibel in den linearen Schalldruck umzuwandeln, gemittelt und anschließend erneut logarithmiert, um das Leq-Ergebnis zurückzugeben:

$$L_{p,A,eq,T} = 10 \lg \left(\frac{t_0}{T} \sum_{i=1}^N 10^{L_{p,A,E,i}/10 \text{ dB}} \right) \text{dB}$$

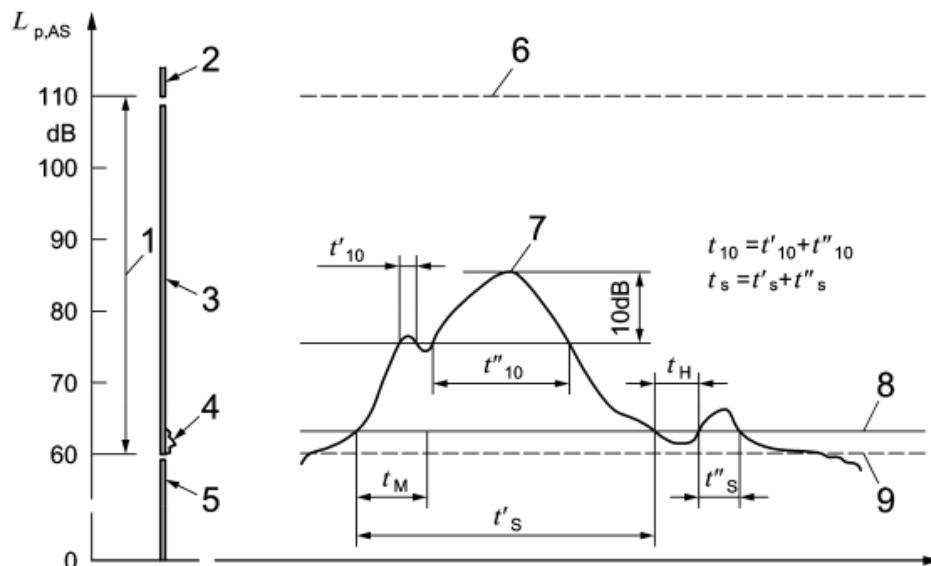
Leq4-Metrik

Die Geräuschemetrik repräsentiert (*leider nur rechtlich, nicht wirklich*) die gesamte akustische Energie eines Überflugereignisses, begrenzt durch die Grenzen seines eigenen Maximalpegels abzüglich 10 dB.

Diese Methode ist die einzige akzeptierte Metrik für das Lärmereignis

(*das real vorhandene und bereits störenden Lärm außerhalb dieses Abschnitts wird leider ignoriert*)

Die T10 (t "10) -Metrik ist die Dauer dieses Ereignisses.



In der Abbildung verwendete Schlüssel nach DIN:

- | | | |
|--|---------------------------------------|--------------------------|
| 1 Primärzeigebreich / Dynamikbereich, | 2 Überlastbereich, | 3 Bereich der Bewertung |
| 4 Bereich außerhalb der Bewertung, | 5 Bereich nicht übertragen, | 6 Obergrenze der Messung |
| 7 Maximaler Schallpegel L_p, AS, max , | 8 Messschwellenpegel $L_p, AS, MSchw$ | |
| 9 Untergrenze des Dynamikbereichs | | |
| t 10 10 dB-Ausfallzeit, H Hörzeit, M Mindestzeit, ts Überschreitungszeit | | |

Leq3-Metrik

Die Leq3-Metrik repräsentiert (leider nicht legal) die gesamte akustische Energie (auch Schalldruck genannt) eines Fluglärmereignisses, das durch einen bestimmten Schwellenwert (z. B. 55 dB) begrenzt ist, der dem tatsächlichen Ärger viel näherkommt. Die tAT (t) -Metrik ist die Dauer dieses Ereignisses.

LAE-Metrik

Die LAE-Metrik (Einzelereignis-Schallbelichtungspegel) ist der Schallpegel, den ein Ereignis definieren würde, wenn seine gesamte Schallenergie praktisch in 1 Sekunde gleichmäßig auf 1 Sekunde komprimiert würde. Dies ist ein Standard-Einzelereigniswert, der z. B. in ISO 1996 beschrieben ist.

Diese Skala berücksichtigt somit die Dauer des Ereignisses sowie seine maximale Intensität.

Lmax-Metrik

Der maximale Wert von L (t) -Slow, der während eines Ereignisses auftritt.

LE-Metrik

Die LE-Metrik ist eine spezielle Form der LEQ-Metrik. Es konzentriert die gesamte akustische Energie eines Ereignisses in einer einzigen Sekunde.

Dies hat den Vorteil, dass Ereignisse unabhängig von ihrer Dauer mit einem einzigen Wert kategorisiert werden. Das nominale Ergebnis führt jedoch zu Werten, die weit über dem Maximum des Ereignisses liegen, was für Nichtspezialisten verwirrend sein kann.

Der LE, berechnet für eine Minute statt für eine Sekunde, wäre 17,8 dB niedriger und damit akzeptabler gewesen, aber ich kann die Norm nicht ändern. ☺

Die LE ist jedoch eine bequeme Zwischenstufe, um die Schallbelastung auf Stunden, Tage oder Monate zu mitteln.

Der Hintergrundpegel

Die Hintergrundpegelmetrik ist der langfristig gemittelte Schallpegel ohne die Geräuschereignisse.

Die Geräuschereignisse müssen (legal) mindestens den Hintergrundpegel um 5dB überschreiten, was meiner Meinung nach lächerlich niedrig ist. Ich habe mindestens 10 dB Abstand zum Hintergrundpegel berücksichtigt, um das Lärmereignis zu akzeptieren.

Tag-Nacht-Durchschnittsschallpegel (Lden)

Diese Lärmmetrik spiegelt die kumulative Lärmbelastung einer Person über einen Zeitraum von 24 Stunden wider, ausgedrückt als Lärmpegel für den durchschnittlichen Tag des Jahres basierend auf dem jährlichen Flugbetrieb, bei zusätzlichen 10 dB für Ereignisse, die während der legalen Nacht auftreten (ab 22: 00 bis 06:00).

Lden ist die Standard-Lärmmetrik, die für alle FAA-Studien zur Fluglärmbelastung in Flughafengemeinden verwendet wird.

Wettereinfluss: Das in der Wetterstation verwendete Mikrofon sollte bei einer konstanten laminaren Windstärke von 10 m / s nicht mehr als 10 dB zusätzliches Lärm verursachen. Ich habe meine Mikrofone an der Technischen Universität Düsseldorf in einem Windkanal getestet und der Test hat das Design als konform erwiesen. Gemäß den Vorschriften ist bei Wind und Windböen über 10 m / s die Überflugerkennung deaktiviert. Die Wetterbedingungen werden von openweathermaps.org erfasst.

Aufgrund der Kosten und Komplexität möchte ich kein Anemometer einsetzen müssen.

NAT-Metrik

Die Anzahl über dem Schwellenwert ist die Anzahl der erkannten Ereignisse gemäß der Leq4-Metrik innerhalb eines bestimmten Zeitraums. Rechtlich gesehen werden NAT-Grenzwerte nur während der Nacht berücksichtigt.

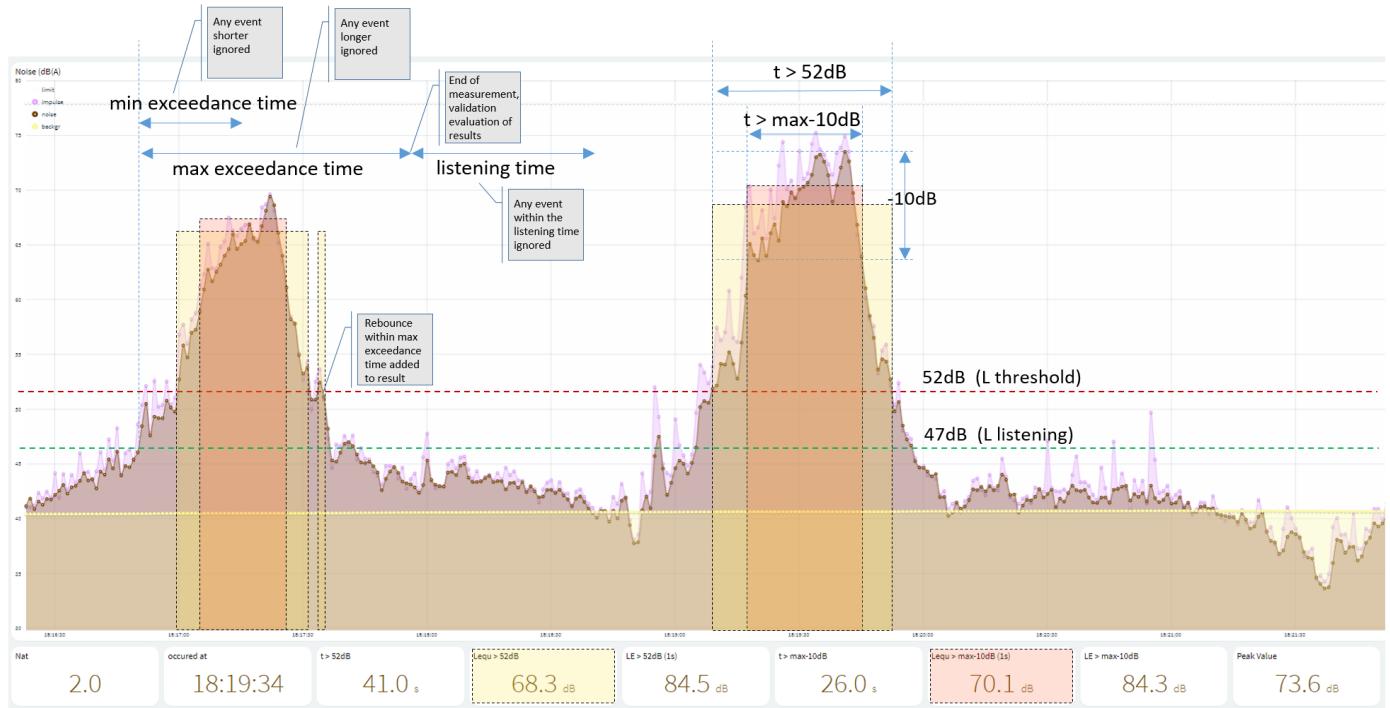
Flughäfen melden NAT normalerweise nur für Tag (6:00 bis 22:00 Uhr) und Nacht (22:00 bis 6:00 Uhr). Ich melde NAT stündlich, um die am stärksten von Lärm betroffenen Stunden zu ermitteln des Tages

Weitere Informationen zu Metriken

ICCAN: Eine Überprüfung der Fluglärmmetriken und -messungen Juli 2020:

https://iccan.gov.uk/wp-content/uploads/2020_08_11_ICCAN_review_of_aviation_noise_metrics_and_measurement.pdf

Ereignisaus- und -bewertung



Bewertung:

Die Bewertung der Geräuschereignisse erfolgt im Programm über eine „State-Machine“.

- **State o:**
Am Ende eines Erkennungszyklus befindet sich die Leerlaufsituation, in der das Programm darauf wartet, dass der Hörzeitgeber abläuft, und dann den Horchpegel verlässt → State a:
- **State a:**
Wenn der Geräuschpegel den Horchpegel überschreitet, werden die Timer für die minimale Überschreitungszeit und die maximale Überschreitungszeit gestartet, die Ereignisaufzeichnung beginnt → State b
- **State b:**
Sollte der Geräuschpegel unter den Horchpegel fallen → State e, andernfalls wird das Ereignis validiert, wenn die Ereignisdauer länger als das Minimum und kürzer als das Maximum ist. → State c
- **State c:**
Wartet, bis die maximale Dauer abgelaufen ist → State d
- **State d:**
- Der Timer für die Horchzeit wird gestartet, die Aufzeichnung wird gestoppt, die Ereignisbewertung ist erfolgreich und wird dann numerisch ausgewertet: - Die Aufzeichnung wird dann abgespielt, um nach dem maximalen Geräuschpegel des Ereignisses zu suchen und dann den maximalen Pegel zu berechnen. - 10 dB. - Die Aufzeichnung wird ausgewertet, wobei die Schallenergie für jede Probe, die den Pegel maximal -10 dB überschreitet, integriert wird. Die entsprechenden Ergebnisse werden gespeichert.
Die Aufnahme wird erneut ausgewertet, wobei die Schallenergie für jede Probe, die den festgelegten Schwellenwert überschreitet, integriert wird. Die entsprechenden Ergebnisse werden gespeichert. → State e
- **State e:**
Wartet, bis die Horchzeit für das nächste Ereignis abgelaufen ist → State o

Auswertung

Jedes Ereignis wird anhand von zwei Referenzen bewertet:

- der Referenz „Maximum minus 10 dB“ und
- der Referenz „fester Schwellenwert“.

Für jede Referenz integriert das Programm die tatsächlich zugehörige Schallenergie der Samples, die der Referenz überschreiten (einschließlich eventueller „Nachbeben“) und die Gesamtdauer der übereinstimmenden (1s) Samples.

- Die LE-Metrik ist der Logarithmus der gesamten Schallenergie
- Die Leq-Metrik ist der Logarithmus der gesamten Schallenergie geteilt durch die Ereignisdauer.

Glossar

dB(A): Dezibel, die Maßeinheit des Schalldruckpegels. Die dB-Skala ist entsprechend der Empfindlichkeit des menschlichen Gehörs für verschiedene Frequenzen logarithmisch aufgebaut. (A) steht für die Standardbewertung, die diese Frequenzabhängigkeit bei Messungen mittels eines Filters berücksichtigt.

L_{AX} : der Einzelereignispegel, der aus $L_{AS,max}$ und t_{10} berechnet wird. Bei nicht vollständig gemessener t_{10} -Zeit kann diese durch eine Dreiecksnäherung genähert werden.

L_{DEN} : der über 24 Stunden gemittelte Dauerschallpegel mit den Teilzeiten Day (06- 18 Uhr), Evening (18-22 Uhr) und Night (22-06 Uhr). Zur Berücksichtigung der erhöhten Störwirkung in der Ruhezeit bekommen die Immissionen am Abend einen Zuschlag von 5 dB, in der Nacht von 10 dB.

$L_{eq}(3)$: der energieäquivalente Dauerschallpegel, der einen gemittelten Pegel der Einzelschallpegel in einem bestimmten Zeitraum darstellt. Die Schallenergie des Dauerschallpegels ist daher äquivalent zur Schallenergie aller Einzelgeräusche. Der Halbierungspunkt $q=3$ bedeutet, dass der Dauerschallpegel bei einer Verdopplung der Vorbeiflüge an einer Messstelle um 3 dB ansteigt, bei einer Halbierung um 3 dB absinkt.

$L_{eqNacht}$: der energieäquivalente Dauerschallpegel für die Nachtstunden von 22-06 Uhr

L_{eqTag} : der energieäquivalente Dauerschallpegel für die Tagesstunden von 06-22 Uhr

$L_{p,A,E}/SEL$ (Sound-Exposure-Level): der Einzelereignispegel, dekadischer Logarithmus des Integrals über die quadratischen Schalldruckwerte während des Zeitintervalls t_s . Er kann mittels energetischer Summation über den Schalldruckpegelverlauf bestimmt werden.

$L_{p,AS,max}$: der maximale Wert im Verlauf des Schalldruckpegels eines Schallereignisses. Für ein gültiges Einzelereignis muss dieser die Messschwelle um mindestens 5 dB überschreiten.

$L_{p,AS,Mschw}$: der Messschwellenpegel, der für jede Messstation individuell bestimmt wird. Ein Geräusch muss die Messschwelle länger als die Mindestzeit tM überschreiten, damit es als ein Messereignis erkannt wird. Die Schwelle sollte mindestens 5 dB über dem Hintergrundpegel liegen. Ihr Wert beträgt zwischen 56 und 61 dB.

$L_{p,AS(t)}$: der Schalldruckpegel als Funktion der Zeit mit der Frequenzbewertung A und der Zeitbewertung S („Slow“).

N_1/N_2 : das Verhältnis der am Messpunkt ermittelten Fluggeräusche (N_1) zu den stattgefundenen Vorbei- und Überflügen, die relevant zur Schallimission am Messort beitragen (N_2). Die Erfassungsrate aller Fluggeräusche an einer Messstelle muss mindestens 50% betragen, d.h. $N_1/N_2 \geq 0,5$.

t_{10} (10-dB-down-time): eine Messgröße für die Dauer eines Flugzeuggeräusches, während der der Schalldruckpegel eines Geräusches nicht mehr als 10 dB unter dem Maximalpegel des Fluggeräusches liegt.

t_H : die Horchzeit, die zur Trennung verschiedener Einzelschallereignisse festgelegt wird. Ein Ereignis ist beendet, wenn der Pegel nach Unter- schreiten der Messschwelle $L_{p,AS,max}$ innerhalb der Horchzeit nicht wieder über die Schwelle steigt. Sie beträgt in der Regel 5 Sekunden.

t_M : die Mindestzeit, die ein Geräusch den Messschwellenpegel $L_{p,AS,max}$ übersteigen muss, damit es als Einzelereignis gezählt wird. Kurzzeitige Fremdgeräusche werden so nicht als Fluggeräusch interpretiert. Die t_M beträgt in der Regel 5 Sekunden.

t_s : die Länge eines Schallereignisses. Sie entspricht der Dauer der Überschreitung der Messschwelle $L_{p,AS,max}$.