

BIBLIOTEKA
POLSKIEGO KRÓTKOFALOWCA

67

KRZYSZTOF DĄBROWSKI
OE1KDA

MIESZANKA FIRMOWA
TOM 2

WIEDEN 2023



© Krzysztof Dąbrowski OE1KDA
Wiedeń 2023

Opracowanie niniejsze może być rozpowszechniane i kopiowane na zasadach niekomercyjnych w dowolnej postaci (elektronicznej, drukowanej itp.) i na dowolnych nośnikach lub w sieciach komputerowych pod warunkiem nie dokonywania w nim żadnych zmian i nie usuwania nazwiska autora. Na tych samych warunkach dozwolone jest tłumaczenie na języki obce i rozpowszechnianie tych tłumaczeń.

Na rozpowszechnianie na innych zasadach konieczne jest uzyskanie pisemnej zgody autora.

**Mieszanka firmowa
Tom 2**

Krzysztof Dąbrowski OE1KDA

**Wydanie 1
Wiedeń, marzec 2023**

Spis treści

Wstęp	6
1. System LoRa z rozpraszaniem widma	8
1.1. Znaczenie parametrów transmisji	15
1.2. Rozwiązania stacji	18
1.2.1. Eksperymenty dla początkujących	22
1.2.2. Dydaktyczny program nadajnika APRS w systemie „LoRa”	25
1.3. LoRa APRS	34
1.4. Wyposażenie stacji w systemie „LoRa”	40
1.5. Sieć systemu „LoRa MeshCom”	41
1.5.1. Przykładowe zastosowania sieci	44
1.5.2. Instalacja i uruchomienie oprogramowania	45
1.5.3. Wymiana meldunków z innymi systemami	48
1.6. Odbiór satelitów „LoRa”	48
1.7. Próbnik połączeń „LoRa”	50
2. Sondy meteorologiczne	53
3. Przenoszenie rozmów w sieci DMR	58
4. Szybka transmisja danych w trybie DD	61
5. Tłumienie wolnej przestrzeni	64
6. QDX – radiostacja z cyfrową obróbką sygnałów	68
7. „SDR-Control” – program sterujący dla radiostacji Icom	71
Literatura i adresy internetowe	75
Spis tomów „Biblioteki polskiego krótkofalowca”	77

Sommaire

Mélange de divers 2

Préface	6
1. Système LoRa à spectre étalé	8
1.1. Les paramètres de LoRa et leurs relations	15
1.2. Constructions des postes	18
1.2.1. Expériences pour débutants	22
1.2.2. Logiciel didactique de poste APRS „LoRa”	25
1.3. LoRa APRS	34
1.4. Équipement des postes „LoRa”	40
1.5. Le réseau „LoRa MeshCom”	41
1.5.1. Exemples d'applications	44
1.5.2. Installation et démarrage de logiciel	45
1.5.3. Échange de messages avec autres systèmes	48
1.6. La réception des satellites „LoRa”	48
1.7. Testeur des liaisons „LoRa”	50
2. Les sondes météorologiques	53
3. Itinérance vocale en réseaux DMR	58
4. Transmission de données par le mode DD	61
5. L'atténuation en espace libre	64
6. Le poste DSP „QDX”	68
7. Le logiciel télécommande pour les postes d'ICOM „SDR-Control”	71
Bibliographie et les pages web	75
Liste des volumes de la „Bibliothèque de radioamateur polonais”	77

Wstęp

Obecny tom zawiera wybrane tematy poruszane wcześniej w Świecie Radio lub w pewnej części w poprzednich tomach „Biblioteki polskiego krótkofalowca” oraz materiały nie wystarczające (na razie) objętościowo na oddzielny tom. Zostały one pogrupowane tematycznie, zaktualizowane i uzupełnione o dodatkowe informacje zaokrąglające całość.

Tematem aktualnym i coraz bardziej zyskującym na znaczeniu nie tylko w krótkofalarstwie jest radiowy system nadawania danych oparty o transmisję z rozpraszaniem widma sygnału – *LoRa*. Jest on wykorzystywany m.in. do transmisji danych telementrycznych albo do komunikacji w lokalnych lub bardziej rozległych sieciach. Rozpraszanie widma powoduje zwiększenie szerokości pasma sygnału nadawanego w stosunku do sygnału modulującego. Stosunek szerokości pasm sygnału nadawanego i modulującego określa zysk systemowy, dzięki któremu możliwe jest stosowanie niskich lub bardzo niskich mocy nadawania. Leżą one przeważnie nawet poniżej 100 mW, a w sytuacjach w których nie zapewniają one pożądanego zasięgu pomocne stają się sieci o mniejszym lub większym zagęszczeniu stacji (ich oczek). W sieci krótkofalarskiej wykorzystywane jest oprogramowanie *MeshCom* oparte na opracowanym przez firmę *Meshtastic*. Dla szerokich rzesz użytkowników powstały sieci Internetu Przedmiotów (IoT) typu *LoraWAN* itp. Temat jest silnie rozwojowy i na pewno będzie poruszany w dalszych publikacjach. Być może w przyszłości zostanie mu poświęcony jeden z kolejnych tomów „Biblioteki”.

Urządzenia nadawczo-odbiorcze *LoRa* są nie tylko tanie, ale też na tyle uniwersalne, że mogą być stosowane w innych systemach transmisji. W połączeniu z odpowiednim oprogramowaniem możliwy jest odbiór danych telementrycznych z satelitów *LoRa* albo z balonów naukowo-badawczych i przekazywanie ich do sieci *TinyGS*. Krótkofalowcy wykorzystują tę technikę m.in. w transmisji danych APRS. Komunikaty APRS też zresztą mogą być uzupełniane o dane telementryczne. Do wyboru jest szeroka gama modeli różniących się wyposażeniem: wyświetlaczami, wyposażeniem w odbiorniki GPS, w pojemniki dla akumulatorów itp.

Sprawą istotną dla użytkowników systemu *LoRa*, a tym bardziej dla eksperymentatorów jest należyte zrozumienie znaczenia parametrów transmisji, ich wzajemnych zależności i ich wpływu na przepustowość kanału albo na uzyskiwane zasięgi i niezawodność transmisji. Sprawy te zostały dokładniej przedstawione w pierwszym rozdziale.

Codziennie o ustalonych międzynarodowo godzinach wypuszczane są do atmosfery balony meteorologiczne. Sondy umieszczone na ich pokładach pozwalają na pomiar ciśnienia atmosferycznego, temperatury, punktu rosy i wilgotności otoczenia. Dane te są przesyłane na ziemię drogą radiową. Po osiągnięciu zadanej wysokości balony pękają, a sondy opadają na ziemię na niedużych spadochronach. Śledzenie sond i poszukiwanie ich na ziemi stało się stosunkowo nową dziedziną zainteresowań krótkofalowców (i nie tylko – do odbioru nie jest wymagane żadne pozwolenie). Niektóre modele sond dają się nawet wykorzystać do celów amatorskich. Zwrot wielu modeli sond do instytutów meteorologicznych jest nawet niepożądany gdyż ich ponowna kalibracja i przystosowanie do wysłania w następny lot wypada drożej niż wysłanie fabrycznie nowego egzemplarza. Wymagane jest odsyłanie przez znalazców jedynie niektórych specjalnych typów sond. Pozostałe należy wyrzucać zgodnie z zasadami obowiązującymi dla złomu elektronicznego i baterii. Ich wykorzystanie do celów krótkofalarskich jest więc całkiem legalne.

W sieci DMR możliwe jest przenoszenie rozmów (ang. *roaming*) z jednego z przemienników znajdujących się na trasie przebywanej przez operatora stacji na następny. Wymaga to wprawdzie pewnego nakładu pracy na konfigurację radiostacji, ale rzecz nie jest aż taka trudna. Niestety nie wszystkie modele radiostacji pozwalają na korzystanie z tej możliwości. Warunkiem jej wykorzystania jest nadawanie sygnałów identyfikacyjnych (radiolatarni) przez stacje przemiennikowe – co nie jest mile widziane przez niektórych z ich operatorów ze względu na zajmowanie przepustowości kanałów radiowych.

Następnym ciekawym tematem jest szybka transmisja danych (DD) w systemie D-STAR. Ze względu na szerokość pasma zajmowaną przez sygnał danych jest ona możliwa w paśmie 23 cm i w wyższych pasmach mikrofalowych. Obecnie pozwalają na to radiostacje IC-9700, ale w najbliższym czasie ma być dostępna radiostacja IC-905 pokrywająca kolejne pasma mikrofalowe aż do 6 cm włącznie. Łączności można prowadzić na trasach bezpośrednich albo przez specjalne przemienniki dostosowane do tego standardu transmisji. W Polsce obecnie nie ma żadnego przemiennika tego rodzaju, dlatego też autor ograniczył się do omówienia łączności bezpośrednich. Transmitowaną mogą być dane o dowol-

nym znaczeniu i pochodzeniu, mogą być to więc dane komputerowe, cyfrowy głos dowolnie zakodowany, obrazy itp.

Rozważając zasięgi osiągnięte w różnych zakresach fal stykamy się ze zjawiskiem zwanym tłumieniem wolnej przestrzeni. Nie ma ono nic wspólnego ze stratami energii w środowisku rozchodzenia się fal radiowych, ale wiąże się ze stosunkiem powierzchni skutecznej anteny do długości fali i z zależnościami geometrycznymi dotyczącymi odległości między anteną nadawczą i odbiorczą i rozkładaniem się energii na powierzchnię rosnącą w miarę odległości. Stacje odbiorcze mogą więc wychwytywać coraz mniejsze części nadawanej energii. Sprawie tej poświęcamy piąty rozdział opracowania.

Technika cyfrowej obróbki sygnałów wchodzi coraz szerzej nie tylko do urządzeń fabrycznych ale i do zestawów konstrukcyjnych prostych radiostacji. Jednym z takich rozwiązań jest przedstawiona w tym tomie radiostacja QDX przeznaczona do pracy niektórymi emisjami cyfrowymi na falach krótkich. Natomiast proste rozwiązania radiostacji fonicznych UKF możliwych do skonstruowania i uruchomienia we własnym zakresie, opartych na modułach cyfrowej obróbki sygnałów DRA818V i DRA818U, opisano w tomach 36 i 57 „Biblioteki”.

Zdalne sterowanie radiostacji przez lokalną sieć w domu albo przez Internet bywa wygodne w wielu sytuacjach i dlatego cieszy się zainteresowaniem krótkofalowców. W obecnym tomie zaprezentowano program „SDR-Control” przeznaczony do zdalnej obsługi nowszych modeli radiostacji Icoma. Ogólne rozważania dotyczące zdalnego sterowania radiostacji zawiera pierwszy tom mieszanki firmowej (tom 66 „Biblioteki”).

Krzysztof Dąbrowski OE1KDA

Wiedeń

2 marca 2023

1. System „LoRa” z rozpraszaniem widma

Oznaczenie „LoRa” jest skrótem od określenia „Long Range” mającego oznaczać telemetrię daleko-sięzną, przy czym o ile w porównaniu z rozpowszechnionymi systemami transmisji danych pomiarowych na dystansach kilku lub kilkudziesięciu metrów zasięgi dochodzące w korzystnych warunkach do 20 lub więcej km można uznać za dalekie. W oczach krótkofalowców nie są to oczywiście żadne DX-y, ale przecież nie o to najbardziej chodzi.

System „LoRa” korzystający z transmisji z rozpraszaniem widma sygnału (ang. *spread spectrum*) został opracowany w 2013 roku we francuskiej firmie „Cycleo” wykupionej następnie przez firmę „Semtech”. Obecnie na rynkach dostępne są moduły dla pasm 433 i 868 MHz, a w USA także pracujące w paśmie 915 MHz. Do zastosowań krótkofalarskich nadają się oczywiście moduły na zakres 433 MHz (typu RFM98W-433S2 itp.), a do zastosowań nie wymagających licencji amatorskiej – moduły dla pasma 868 MHz (RFM95W-868S2 itp.). W paśmie 868 MHz w zależności od podzakresu obowiązuje ograniczenie mocy do 10 lub 25 mW i nadawania do 0,1% lub 1% czasu pracy – liczonego w skali godzinowej.

Pomimo niskich mocy nadawania – od kilkunastu do maksimum 120 mW – osiągalne są, dzięki zastosowaniu transmisji z rozpraszaniem widma, stosunkowo dalekie zasięgi. W porównaniu z transmisją z kluczkowaniem FSK lub GMSK są one około dziesięciokrotnie dalsze. Zależny od współczynnika rozpraszania widma zysk systemowy wynosi tutaj około 20 dB.

Niskie moce nadawania pozwalają na bateryjne zasilanie urządzeń. Obecnie produkowane moduły pobierają przy nadawaniu od 20 mA przy mocy 5 mW do 120 mA przy mocach rzędu 120 mW.

Krótkofalowcy mogą, w odróżnieniu od użytkowników nielicencjonowanych korzystać z dowolnych anten zewnętrznych i nie obowiązują ich ograniczenia czasu nadawania, a moce są ograniczone jedynie warunkami licencji. Dozwolone jest korzystanie z dodatkowych wzmacniaczy mocy.

Dopuszczalne szybkości transmisji wynoszą wprawdzie od 180 bit/s do 37,5 kbit/s, ale dla uzyskania możliwie największej czułości stosowane są szybkości z dolnego zakresu, co praktycznie jest w pełni wystarczające ponieważ ilość transmitowanych danych jest raczej niewielka. Dla minimalnych szybkości osiągnęte są czułości -148 dBm, natomiast przykładowo dla 1200 bit/s już tylko -119 dBm. Możliwa jest też automatyczna zmiana szybkości transmisji w zależności od jakości łącza (ang. *adaptive data rate* – ADR).

Maksymalna długość transmitowanego pakietu wynosi 256 bajtów, włącznie z danymi korekcyjnymi FEC (korekcji wyprzedzającej). Jest ona ograniczona długością rejestru przesuwonego zawartego w obwodzie scalonym modemu.

Współczynnik rozpraszania widma (stosunek szerokości pasma sygnału nadawanego do szerokości widma kanału podstawowego) leży w granicach 6 – 12, a pasmo sygnału transmitowanego zajmuje od 7,8 do 500 kHz. Najczęściej stosowane jest pasmo 125 lub 250 kHz.

Moduły radiowe „LoRa” są stosowane w profesjonalnych systemach telemetrycznych, w zdalnym odczycie liczników elektrycznych i innych przyrządów, w układach zdalnego sterowania, w systemach bezpieczeństwa, alarmowych, w śledzeniu przesyłek, bagażu na lotniskach albo innych obiektów itp.

Do spraw problematycznych należy wprawdzie kwestia zabezpieczenia transmitowanych danych, ale w zastosowaniach amatorskich sprawa ta nie ma większego znaczenia.

Moduły „LoRa” pozwalają dodatkowo na transmisję danych z kluczkowaniem FSK, GFSK, MSK, GMSK i OOK. Możliwe jest więc wykorzystanie ich w innych systemach transmisji i sieciach, śledzenie sond meteorologicznych itp.

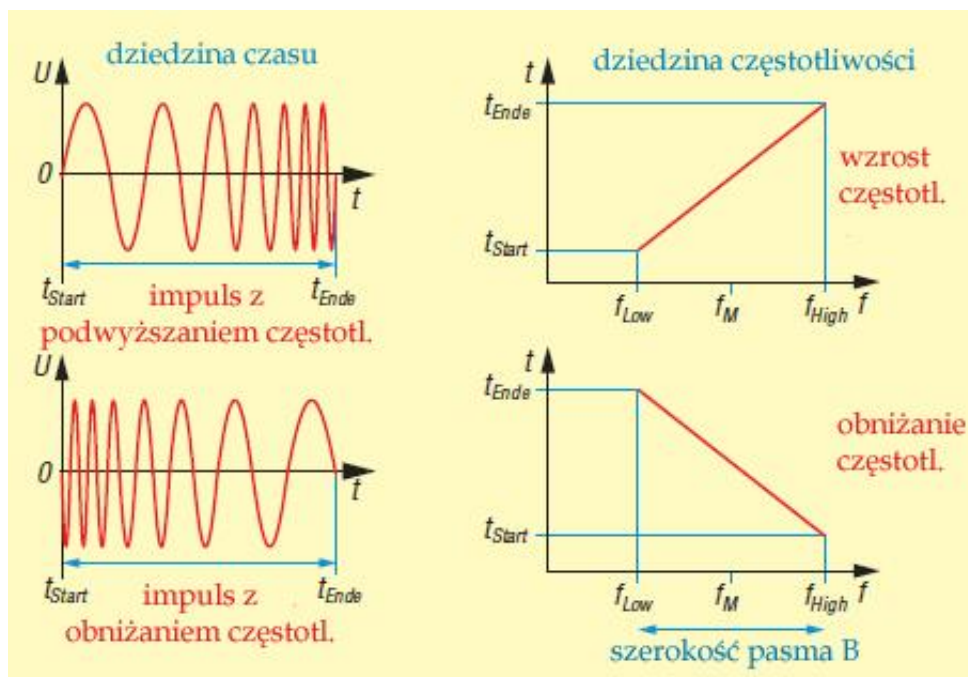
Do najczęściej stosowanych metod rozpraszania widma sygnału należą kluczkowanie częstotliwości (ang. *frequency hopping* – FH), kluczkowanie fazy kodem losowym (ang. *direct sequence spread spectrum* – DSSS) i linowa zmiana częstotliwości (ang. *chirp*). Do rozpraszania widma stosowane są kody pseudoprzypadkowe. W najprostszym przypadku są one generowane za pomocą rejestrów przesuwanych z odpowiednio dobranymi sprzężeniami zwrotnymi, ale mogą być także generowane programowo. Istnieją całe rodziny takich kodów, a nowe kody mogą być też generowane przez kombinację istniejących. Teoria kodów pseudoprzypadkowych jest dość rozbudowana i skomplikowana.

Zysk systemowy jest zależny od stopnia rozproszenia widma czyli stosunku szerokości pasma sygnału rozproszonego do sygnału podstawowego. Sygnał o rozproszonym widmie jest odbieralny przez klasyczne odbiorniki wąskopasmowe jako szum. Dodaje się on do szumów pochodzących z innych

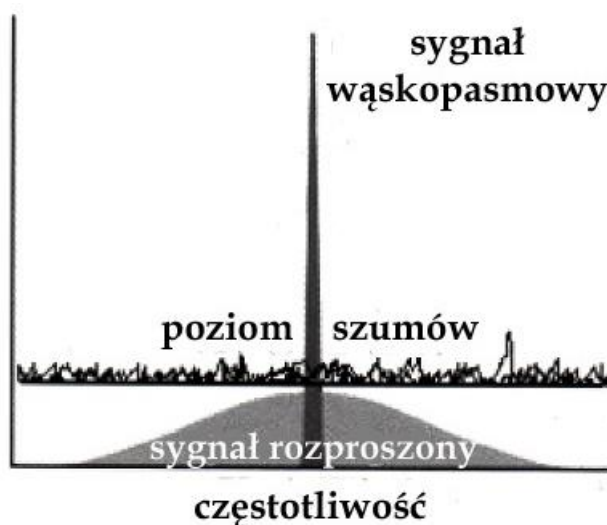
źródeł. Przy wielu zachodzących na siebie sygnałach o rozproszonym w dowolny sposób widmie selektywny wybór możliwy jest dzięki stosowaniu różnych kodów. Dla prawidłowego odbioru konieczna jest oprócz zgodności kodów także zgodność ich faz (w sygnale odbieranym i w kodzie generowanym w odbiorniku), a także zgodność ich częstotliwości zegarowych. Sprawa zapewnienia synchronizacji należy do najważniejszych problemów w komunikacji z rozpraszaniem widma sygnału.

Oprócz radiokomunikacji transmisja z rozpraszaniem widma jest stosowana m.in. w zdalnym sterowaniu i w technice pomiarowej. Pomiar przesunięcia kodu odbieranego w stosunku do nadawanego pozwala, zwłaszcza przy użyciu odpowiednio długich kodów, na dokładne pomiary odległości.

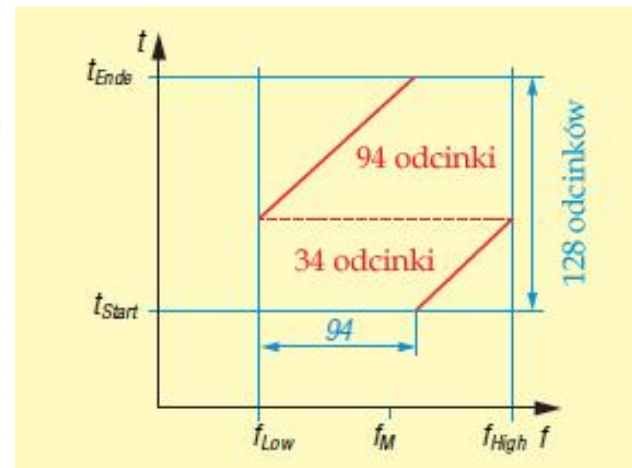
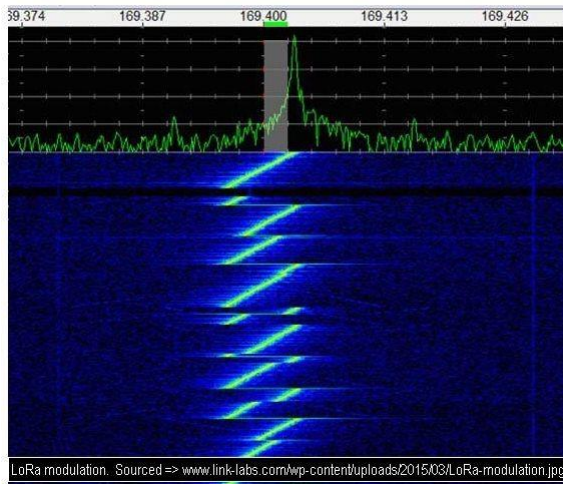
W modułach „LoRa” stosowany jest uproszczony system liniowej zmiany częstotliwości (ang. *chirp* – CSS) oparty na standardzie IEEE 802.15.4.



Rys. 1.1. Przebieg sygnałów modulowanych liniowo w dziedzinie czasu i częstotliwości (źródło: *Funkamateurl* 11/2021)

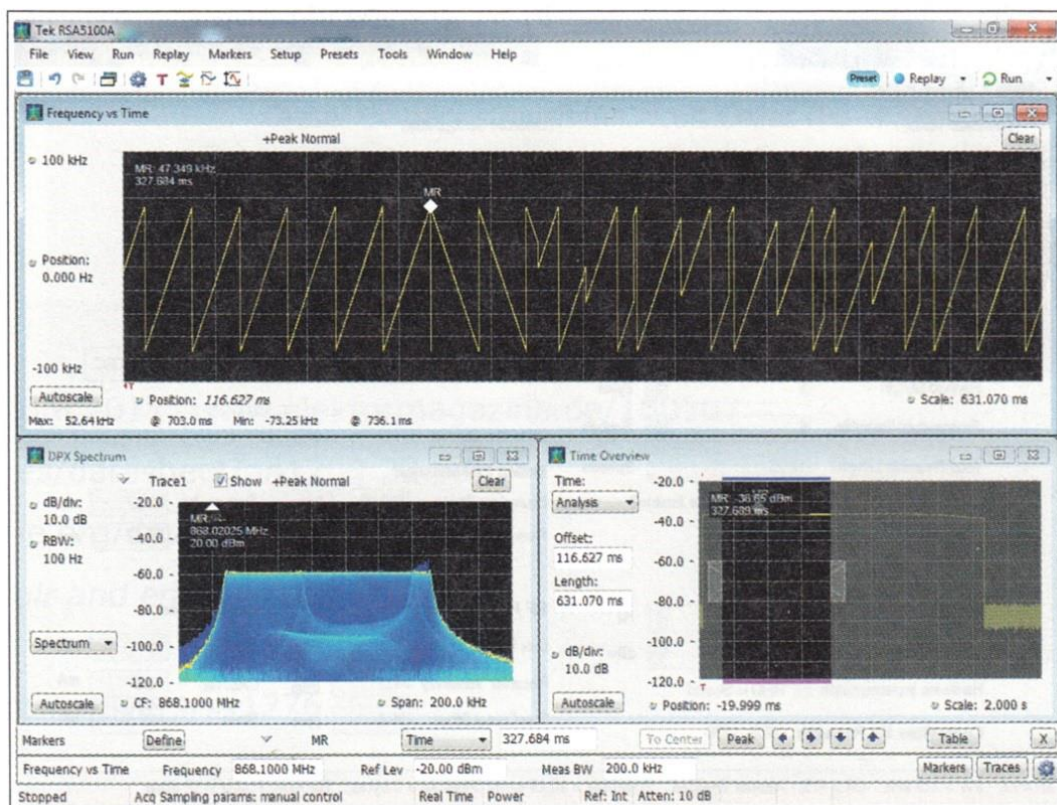


Rys. 1.2. Transmisja sygnału o rozproszonym widmie. Sygnał użytkowy jest po stronie nadawczej rozpraszany, a po stronie odbiorczej skupiany w identyczny sposób i przy użyciu identycznego kodu dla odzyskania danych w paśmie podstawowym. Ewentualnie odebrane sygnały wąskopasmowe znajdujące się w tym samym paśmie ulegają w odbiorniku rozproszeniu i nie powodują zakłóceń



Rys. 1.3a. Przebieg sygnału systemu „LoRa” w funkcji czasu i częstotliwości (źródło: dokumentacja LoRa)

Rys. 1.3b. Zasada kodowania na przykładzie symbolu o wartości dziesiętnej 94 (źródło: Funkamateurl1/2021)



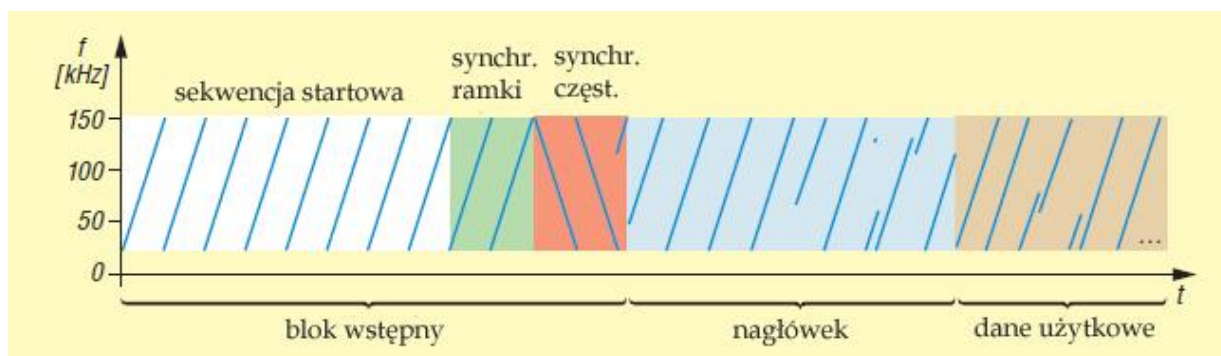
Rys. 1.4. Sygnał systemu LoRa zmierzony analizatorem widma RSA5106 firmy Tektronix (źródło: Elektor maj-czerwiec 2017)

Na ilustracji 1.4 przedstawiona jest dokonana w laboratorium *Elektora* analiza sygnału systemu „LoRa” dla pasma o szerokości 125 kHz, współczynnika rozpraszania 12, FEC 4/5 i dewiacji również 125 kHz. W bloku synchronizacyjnym przebiegi są odwrócone: częstotliwość zmniejsza się, a nie wzrasta w funkcji czasu. Blok synchronizacyjny składa się z 8 symboli, przy czym do synchronizacji odbiornika wystarczają 4 z nich.

Bitów zgrupowane są w symbole 12-bitowe – przyjmują więc wartości od 0 do 4095. Czas trwania symbolu wynosi 32,8 msek. Przepustowość brutto równa jest 366,2 bit/s, co przy FEC 4/5 daje przepustowość netto 292,9 bit/s.

Maksymalna dewiacja 125 kHz jest podzielona na 212 odcinków: $125000/4096 = 30,52$ Hz. Dewiacja chwilowa równa jest wartości symbolu $\times 30,52$ Hz. Transmisja 8 bajtów użytkowych trwa 925 msek.

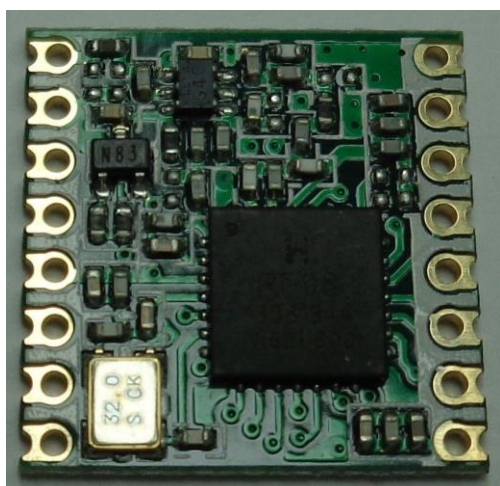
Najczęściej system jest stosowany do niezbyt częstych transmisji niewielkich ilości danych. Transmisja z dużymi szybkościami nie opłaca się. Oprócz zastosowań naziemnych w praktyce amatorskiej interesująca jest transmisja danych z balonów. Dzięki znacznym wysokościami lotu uzyskuje się duże zasięgi stacji. Inną interesującą dziedziną może być śledzenie zwierząt wyposażonych w nadajniki.



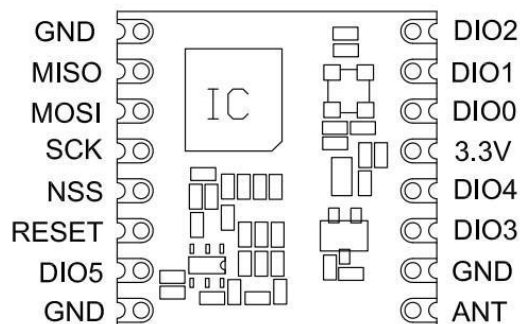
Rys. 1.5. Orientacyjna struktura pakietu „LoRa” (źródło: *Funkamateur* 11/2021)

Moduły RFM98W i podobne komunikują się z mikrokomputerem za pomocą złącza SPI (*Serial Peripheral Interface*), w którego skład wchodzi sygnały MOSI (*Master Output Slave Input*), MISO (*Master Input Slave Output*), SCK (sygnał zegarowy) i SS (*Slave Select*). Sygnał selekcji urządzenia SS musi być połączony z wejściem NSS modułu „LoRa”. Jako wyjście selekcyjne po stronie *Arduino* może służyć dowolne wyjście logiczne.

Kontakty DIO0 – DIO5 modułu są wykorzystywane przez bibliotekę LMIC do różnych celów j.np. informacji o pracy urządzenia. Moduły pracujące w paśmie 433 MHz zawierają przeważnie nadajniki-odbiorniki SX1278 firmy „Semtech”. Ich syntezer pokrywa zakres częstotliwości 137 – 525 MHz. Podobny do niego SX1268 pokrywa zakres 410 – 525 MHz. W zakresie 860 – 1000 MHz pracują układy SX1272/1273, a SX1276 pokrywa zakres 433 MHz oraz pasma przemyślne 868 i 915 MHz. Moduły SX1280/1281 pracują w paśmie 2,4 GHz.



Fot. 1.6. Moduł RFM dla pasma 433 MHz...



Rys. 1.7. ...i jego wyprowadzenia

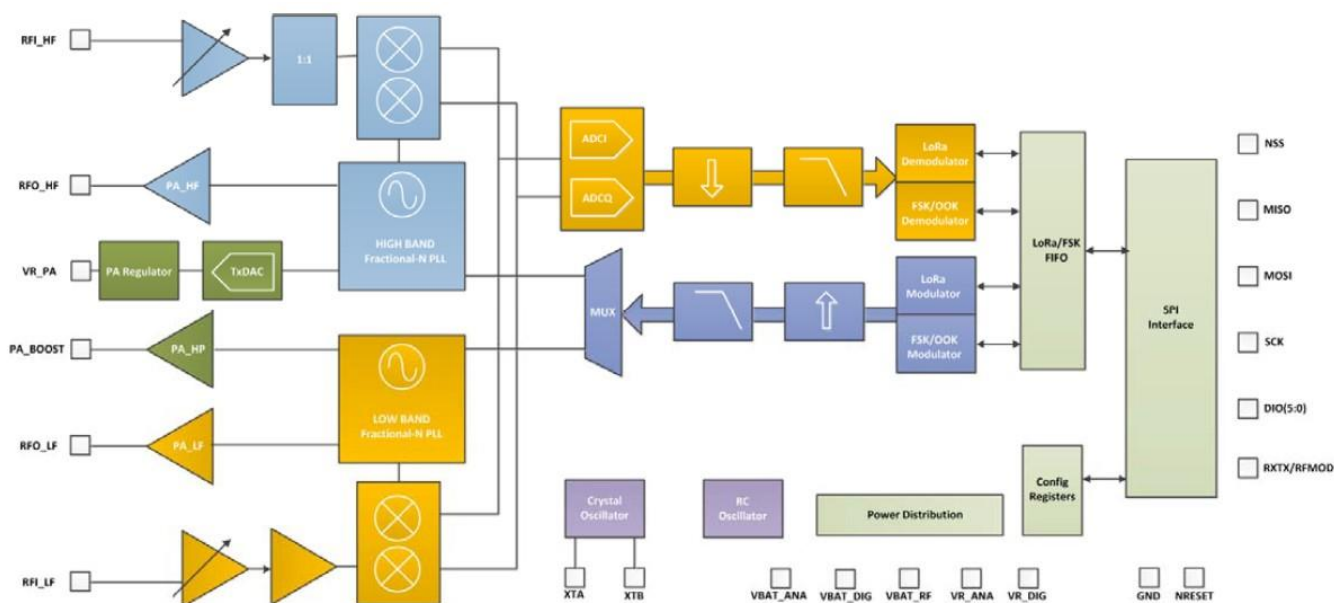
Tabela 1.1

Minimalny stosunek sygnału do szumu w zależności od współczynnika rozpraszania dla modułu SX1278 (GMSK dla porównania)

Współcz. rozpr.	Stos. sygn./szum [dB]	Współcz. rozpr.	Stos. sygn./szum [dB]
6	-5,0	10	-15,0
7	-7,5	11	-17,5
8	-10,0	12	-20,0
9	-12,5	GMSK	9

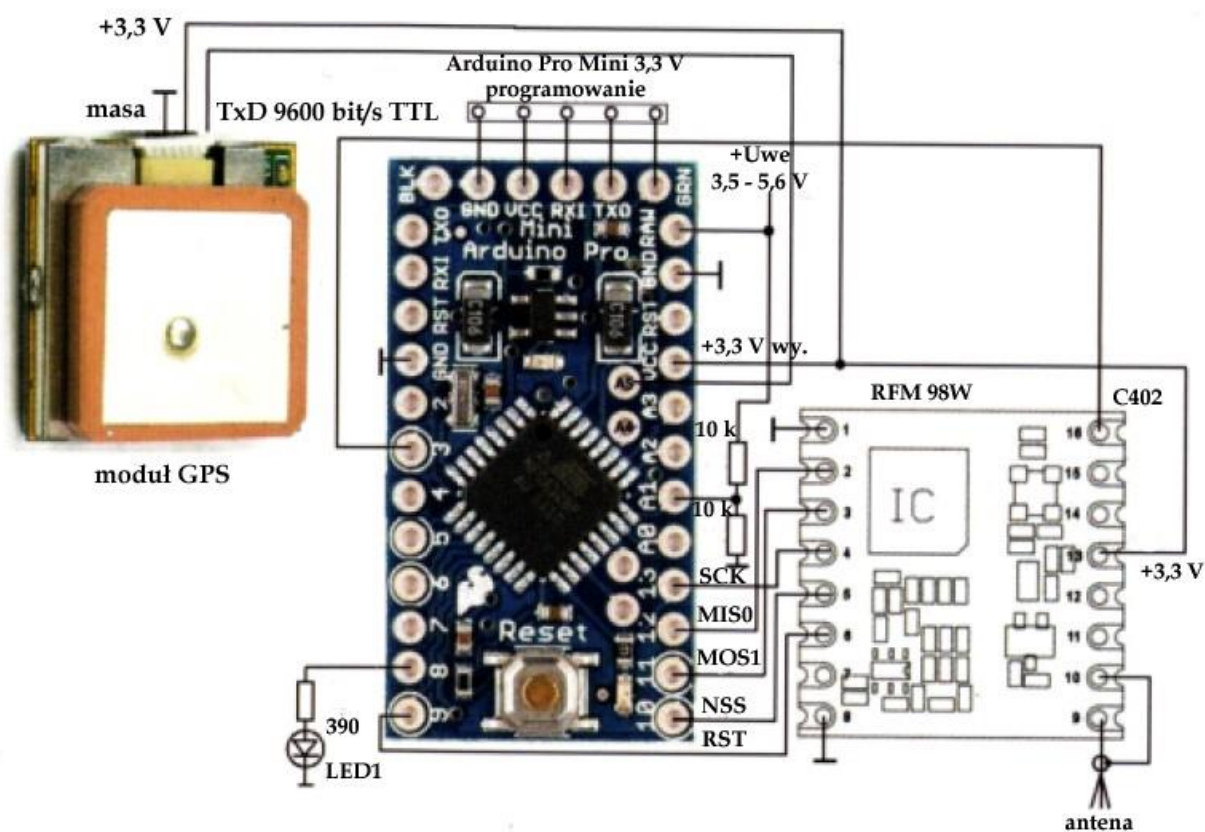
Czułość odbiornika oblicza się ze wzoru:

$ERX = 10 \log(B) + WS + SSS - 174 \text{ dBm}$, gdzie B jest szerokością pasma sygnału w Hz (dla 125 kHz $10 \log(125000) = 51$, WS – współczynnikiem szumów odbiornika (równym 6 w paśmie 70 cm), a SSS – współczynnikiem sygnał/szum według tabeli.

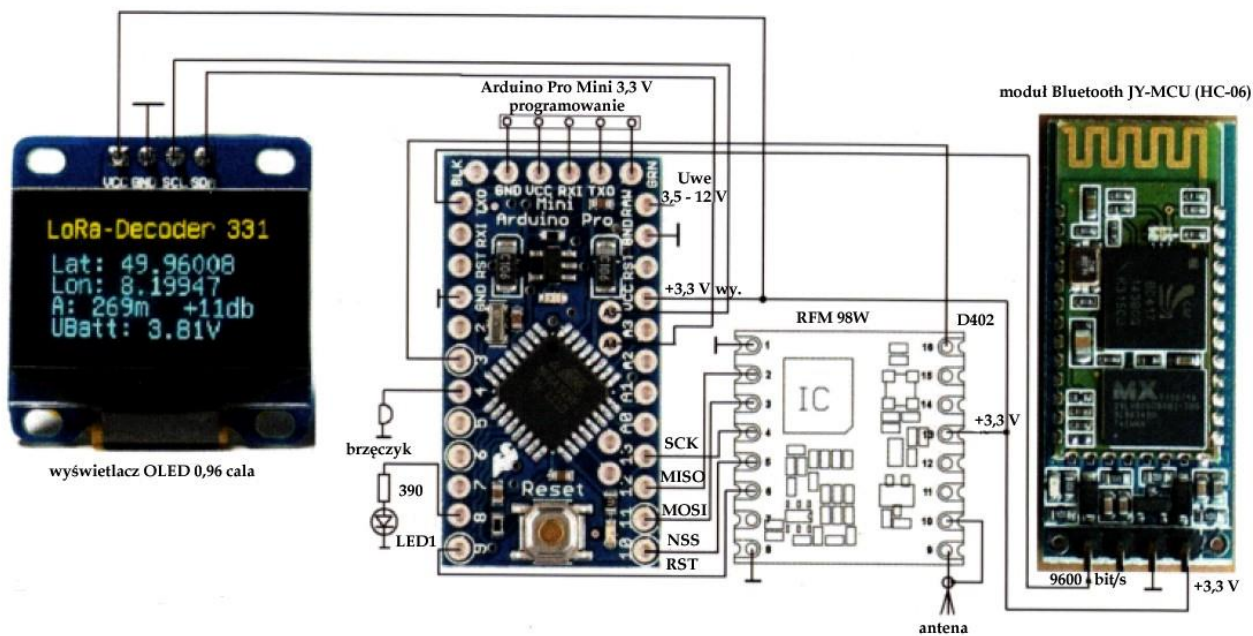


Rys. 1.8. Schemat blokowy modułu z serii SX

W przedstawionym na ilustracji 1.9 nadajniku APRS zastosowano mikrokomputer Arduino Mini zasilany napięciem 3,3 V, co ułatwia połączenie obu części, a także modułu odbiornika GPS. Oprócz danych APRS nadawana jest informacja o napięciu zasilania. Napięcie to jest doprowadzone do wejścia A1 „Arduino” przez dzielnik 1:2 złożony z dwóch oporników 10 kΩ. Krótkofalowcy eksperymentowali również z transmisją danych meteorologicznych: ciśnienia atmosferycznego, temperatury i wilgotności. W pasującym do tego odbiorniku APRS zastosowano wyświetlacz OLED połączony z mikrokomputerem za pomocą magistrali I2C (sygnał zegarowy SCL pochodzi z wyjścia A5 „Arduino”, a sygnał danych SDA – z wyjścia A4). Moduł *Bluetooth* służy do połączenia odbiornika z komputerami lub telefonami komórkowymi. Dla telefonów i komputerów androidowych dostępne są m.in. programy *BluTerm*, *W2APRS* i *Lotus Map Pro*. Pierwszy z nich jest programem terminalowym pozwalającym na odczyt danych w postaci tekstowej, a następnie umożliwiają przedstawienie pozycji odbieranej stacji na mapie.



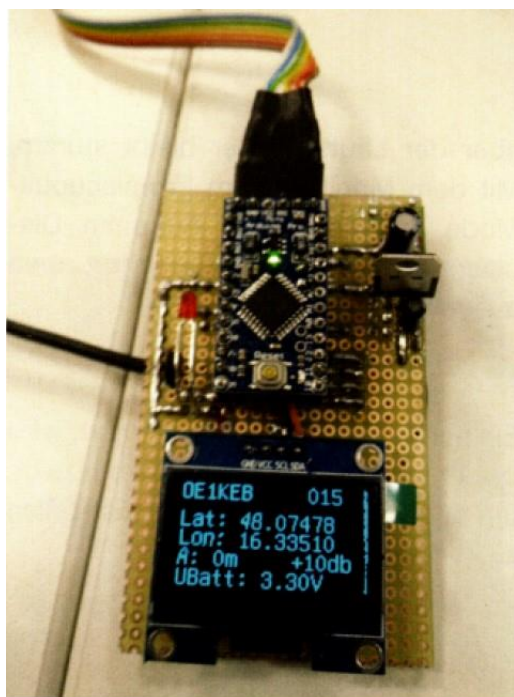
Rys. 1.9. Nadajnik APRS z modulem „LoRa” (źródło: [1.4] i QSP 4/2017)



Rys. 1.10. Odbiornik APRS (źródło: [1.4] i QSP 4/2017)

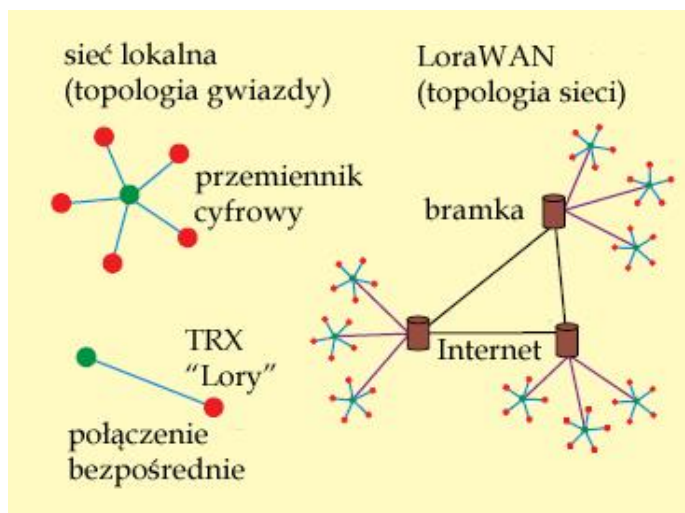


Fot. 1.11. Transmisja danych meteorologicznych: ciśnienia, temperatury, wilgotności względnej i wysokości zmierzonych za pomocą czujnika BME280 firmy „Bosch”. Dodatkowo wyświetlana jest informacja o sile odbieranego sygnału (odstępie poziomu sygnału od szumu)



Fot. 1.12. Odbiornik konstrukcji OE1KEB (źródło: QSP 4/2017)

Rys. 1.13. Możliwe rodzaje połączeń (topologie) w systemie „Lora” (źródło: Funkamateur 11/2021)



Wyposażony w SX1276/1278 moduł z ilustracji 1.14 dysponuje mocą wyjściową 100 mW (pobór prądu wynosi wówczas 120 mA, przy 20 mW – 29 mA), charakteryzuje się wysoką odpornością na blokowanie, czułością -148 dBm i niskim poborem prądu. W trakcie odbioru pobór prądu wynosi około 11 – 12 mA. Moduł jest kompatybilny z płytkami Arduino UNO, MEGA, DUE i Leonardo zasilanymi napięciem 3,3 V i 5 V.

Oprócz transmisji w systemie „Lora” do dyspozycji są modulacje FSK, GFSK, MSK, GMSK i kluczowanie amplitudy OOK (od zera do pełnej mocy jak w telegrafii). Dzięki temu możliwy jest odbiór sygnali pracujących w innych sieciach lub systemach, przykładowo sond radiowych wysyłanych na pokładach balonów meteorologicznych. Syntezator częstotliwości pozwala na nastawienie częstotliwości pracy z rozdzielczością 61 Hz. Przy szerokościach pasma sygnału rozproszonego poniżej 125 kHz kwarcowy generator sterujący należy zastąpić stabilizowanym temperaturowo – TCXO.

Płytkę ma wymiary 62 x 43 x 23 mm i masę 22 g.



Fot. 1.14. Moduł nadawczo-odbiorczy z RFM98W-433S2 dla „Arduino” (źródło: exp-tech.de)

1.1. Znaczenie parametrów transmisji

Zmiany parametrów transmisji w systemie LoRa nie pozostają bez wpływu na szybkość transmisji i zasięg odbioru stacji. Oczywiście stacja odbiorcza musi nie tylko być nastrojona na częstotliwość pracy nadajnika, ale mieć także nastawione identycznie pozostałe parametry” szerokość pasma (BW), współczynnik rozpraszania (SF) i stopę danych korekcyjnych (CR).

Tabela 1.1.1

Porównanie możliwych parametrów transmisji dla różnych modułów nadawczo-odbiorczych RFM

Typ	Zakres [MHz]	Współczynnik rozpraszania	Pasma [kHz]	Szybkość transmisji netto	Czułość [dBm]
RFM95W	868/915	6 – 12	125 – 500	0,293 – 37,5 kbit/s	-111 – -136
RFM96W/ 98W	433/470	6 – 12	62,5 – 500	0,1465 – 37,5 kbit/s	-112 – -140
RFM98W	169	6 – 12	31,25 – 125	73,24 – 9375 bit/s	-118 – -143

Rozpraszanie widma sygnału powoduje zwiększenie odporności transmisji na wpływ szumów. W zależności od modułu do dyspozycji są następujące szerokości pasma (BW) rozproszonego sygnału (dewiacje): 31,25 kHz, 41,7 kHz, 62,5 kHz, 125 kHz, 250 kHz i 500 kHz. Wybór niższych szerokości pasma powoduje przedłużenie czasu transmisji. Pasma poniżej 125 kHz wymagają wyposażenia stacji w TCXO ze względu na wymaganą większą stabilność częstotliwości pracy. Dla sieci LoRaWAN przyjęto standardowo pasmo 125 kHz (w sieciach krótkofalarskich także 250 kHz). W paśmie 2,4 GHz stosowane są szerokości pasma 500 kHz – 1,6 MHz. Ogólnie rzecz biorąc węższe pasmo sygnału oznacza zmniejszenie szybkości transmisji i zwiększenie zasięgu stacji. Szersze pasmo oznacza zwiększenie szybkości transmisji i jednocześnie zmniejszenie zasięgu.

Również zmiana wartości współczynnika rozpraszania (SF) określająca liczbę symboli (ang. *chirps*) reprezentujących każdy z bitów danych. Przy niskich współczynnikach rozpraszania wzrasta w efekcie liczba transmitowanych bitów danych czyli szybkość transmisji. Jednocześnie powoduje to zmniejszenie czułości strony odbiorczej, co oznacza zwiększenie prawdopodobieństwa utraty danych w kanale transmisyjnym (większą zawodność transmisji). Większe współczynniki oznaczają z kolei zmniejszenie szybkości transmisji danych użytkowych, ale daje to wzrost czułości odbiorników i wzrost niezawodności transmisji. Skutkuje to jednocześnie przedłużeniem czasu transmisji i wzrostem średniego poboru mocy przez nadajnik, co może być istotne w niektórych zastosowaniach (czujnikach zasilanych bateryjnie i umieszczonych w miejscach trudno dostępnych). Tak więc i tutaj wszystko sprowadza się do wy-

boru między szybkością i niezawodnością transmisji (czułością odbiornika). Dla zakresów częstotliwości poniżej 1 GHz dopuszczalne są współczynniki rozpraszania SF6 – SF12. W praktyce najniższym stosowanym współczynnikiem jest SF7 gdyż SF6 wymaga użycia TCXO jako generatora sterującego. Oznaczają one potęgę dwójki dająca liczbę symboli reprezentujących każdy z bitów. Przykładowo SF7 oznacza 2^7 czyli 128 symboli, a SF11 – 2^{11} czyli 2048 symboli. Przy współczynniku rozpraszania SF7 transmisja 64 bajtów danych trwa około 120 ms, a przy SF11 – ponad sekundę (patrz.tab. 1.1.2).

Tabela 1.1.2

Orientacyjny czas transmisji bloku 64 bajtów danych w zależności od współczynnika rozpraszania przy paśmie 125 kHz

Współczynnik rozpraszania	Przybliżony czas transmisji [ms]
SF7	120
SF8	220
SF9	400
SF10	700
SF11	1300
SF12	2400

Szerokość pasma i współczynnik rozpraszania definiują fizyczne parametry modulacji. Natomiast stopa danych korekcyjnych CR określa stopień korekcji wyprzedzającej (FEC) pozwalającej na skorygowanie błędnie odebranych danych dzięki danym nadmiarowym (redundantnym) zawartym w nadawnych pakietach. Dane nadmiarowe są obliczane po stronie nadawczej i dodawane do danych użytkowych. Większy udział danych korekcyjnych w całości pozwala na skorygowanie większej liczby przekłamań, zwiększenie niezawodności i zasięgu, ale jednocześnie oznacza, że dane użytkowe zajmują mniejszą część pakietu, czyli zmniejszenie efektywnej szybkości transmisji. Standard LoRy dopuszcza cztery możliwe stopy korekcji 4/5, 4/6, 4/7 i 4/8. Przy wartości 4/5 udział danych użytkowych w całości wynosi 4/5 czyli 80%, a przy następnych odpowiednio mniej aż do 50% przy stopie 4/8. Wybierając stopę korekcji warto pamiętać, że w wielu przypadkach korzystniejsze może być dopuszczenie niewielkiej liczby przekłamaných pakietów zamiast stałego znacznego obniżenia skutecznej szybkości transmisji. Zmiany stopnia korekcji nie wpływają na pozostałe właściwości kanału co oznacza, że odbiorniki skonfigurowane na stopę różną od nadawczej mogą go w dalszym ciągu prawidłowo dekodować.

Tabela 1.1.3

Orientacyjne wartości czułości odbiornika i szybkości transmisji w kanale LoRy w zależności od parametrów transmisji dla szerokości pasma 125 kHz wg [1.1.1]

Współczynnik rozpraszania	Stopa korekcji	Czułość [dBm]	Szybkość transmisji netto
7	4/5	-124,5	5,47 kbit/s
7	4/6	-124,5	4,56 kbit/s
7	4/7	-124,5	3,91 kbit/s
7	4/8	-124,5	3,42 kbit/s
8	4/5	-127,0	3,13 kbit/s
8	4/6	-127,0	2,60 kbit/s
8	4/7	-127,0	2,23 kbit/s
8	4/8	-127,0	1,95 kbit/s
9	4/5	-129,5	1,76 kbit/s
9	4/6	-129,5	1,46 kbit/s
9	4/7	-129,5	1,26 kbit/s
9	4/8	-129,5	1,10 kbit/s
10	4/5	-132,0	977 bit/s
10	4/6	-132,0	814 bit/s

10	4/7	-132,0	698 bit/s
10	4/8	-132,0	610 bit/s
11	4/5	-134,5	537 bit/s
11	4/6	-134,5	448 bit/s
11	4/7	-134,5	384 bit/s
11	4/8	-134,5	336 bit/s
12	4/5	-137,0	293 bit/s
12	4/6	-137,0	244 bit/s
12	4/7	-137,0	209 bit/s
12	4/8	-137,0	183 bit/s
11 (250 kHz)	4/6	-131,5	895 bit/s

Przy podwojeniu szerokości pasma ze 125 kHz do 250 kHz czułość zmniejsza się o 3 dBm, a szybkość transmisji wzrasta dwukrotnie w stosunku do podanych w tabeli dla każdej z kombinacji parametrów. Wybór pasma 500 kHz powoduje odpowiednio zmianę czterokrotną szybkości transmisji i czułości o 6 dBm.

W tabeli kolorem żółtym zaznaczono kombinacje parametrów stosowane w transmisjach APRS w Austrii i niektórych innych krajach zachodnich (12; 4/5) i w Polsce (9; 4/7). Jak wynika z tabeli kombinacja ze współczynnikiem rozpraszania 12 oznacza większą czułość, zasięgi i niezawodność transmisji kosztem przedłużenia czasu transmisji komunikatu APRS do około 3 sekund (szybkość transmisji jest zbliżona do stosowanej w tradycyjnym wariancie na falach krótkich). Natomiast kombinacja ze współczynnikiem rozpraszania 9 oznacza prawie ośmiokrotne pogorszenie czułości (a co za tym idzie zasięgu i niezawodności transmisji), ale przy szybkości transmisji zbliżonej do stosowanych w tradycyjnym systemie APRS 1200 bit/s daje w przybliżeniu czterokrotne skrócenie jej czasu.

W odróżnieniu od zainstalowanych w miejscach niedostępnych czujników telemetrycznych i innych minimalizacja poboru energii w stacjach amatorskich ma mniejsze znaczenie. Bramki radiowo-internetowe są przeważnie zainstalowane w stałych lokalizacjach i mogą być zasilane z sieci energetycznej a nadajniki danych pozycyjnych przenoszone lub przewożone w samochodzie są łatwo dostępne i dzięki temu nie ma trudności w naładowaniu ich akumulatorów lub wymianie baterii. Dużo większe oszczędności można osiągnąć dopasowując odstęp między transmisjami do szybkości poruszania się stacji – czyli przez ograniczenie liczby transmisji do sensownego minimum. Zapobiega to również nadmiernemu zatłoczeniu kanału.

W sieciach „MeshCom” w Austrii stosowane jest obecnie pasmo 250 kHz (poprzednio było to 125 kHz), współczynnik rozpraszania 11 i stopa korekcji 4/6 (ostatnia linia w tabeli z różowym tłem). Jak widać również i w tym przypadku położono nacisk na uzyskanie możliwie dużej czułości i zasięgów. W Niemczech stosowany jest częściowo również współczynnik 10.

Wybór częstotliwości pracy w paśmie 70 cm oznacza, że krótkofalowców nie obowiązują ograniczenia mocy nadawania i procentowego udziału czasu nadawania w całości czasu pracy wymagane od stacji nielicencjonowanych (ISM). Moc nadawania jest ograniczona jedynie warunkami licencji co oznacza, że krótkofalowcy mogą stosować dodatkowe wzmacniacze mocy. Przeważnie jednak moce nadawania nie przekraczają 1 W.

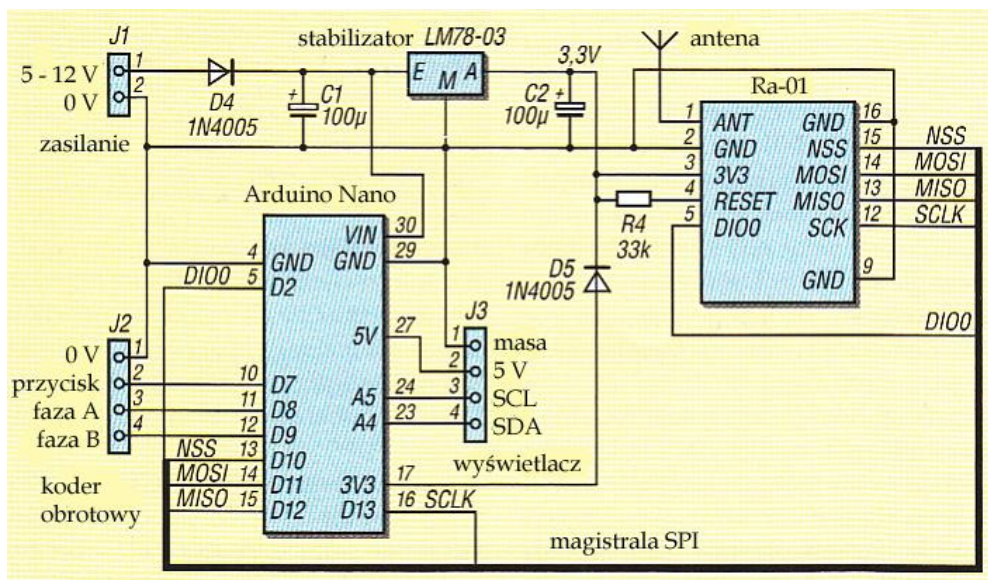
Natomiast w paśmie 868 MHz stacje nadawcze obowiązują ograniczenie mocy do 25 mW i procentowego czasu transmisji do 0,1% lub 1 % w zależności od podzakresu pasma.

1.2. Rozwiązania stacji



Fot. 1.2.2. TRX na modułach Ra-01 lub Ra-02 (źródło: DL2MEE)

Radiostacja systemu *Lora* opracowana przez DL2MEE została opisana w numerze 3/2022 miesięcznika „Funkamateu”. Zawiera ona moduł nadawczo-odbiorczy Ra-01 firmy *Thinker* i sterujący go mikrokomputer *Arduino Nano* zasilany napięciem 3,3 V. Zamiast modułu Ra-01 można zastosować Ra-02. Pierwszy z nich posiada antenę drukowaną i ma możliwość podłączenia anteny śrubowej (helikalnej) wchodzącej w skład kompletu, a drugi posiada gniazdko μ FL/IPEX do podłączenia anteny zewnętrznej. Są one wyposażone w układ nadawczo-odbiorczy SX1278.



Rys. 1.2.2. Schemat ideowy TRX z modulem *Ra-01* i *Arduino Nano* (źródło: Funkamateu 3/2022)

Syntezer Ra-01 jest przestrajany z krokiem 61 Hz, ale wykazuje też znaczne odchyłki od nominalnej częstotliwości. Na szczęście nie mają one większego znaczenia przy szerokopasmowych transmisjach z rozpraszaniem widma. Szerokość widma sygnału wyjściowego SX1278 leży w zakresie 7,8 – 500

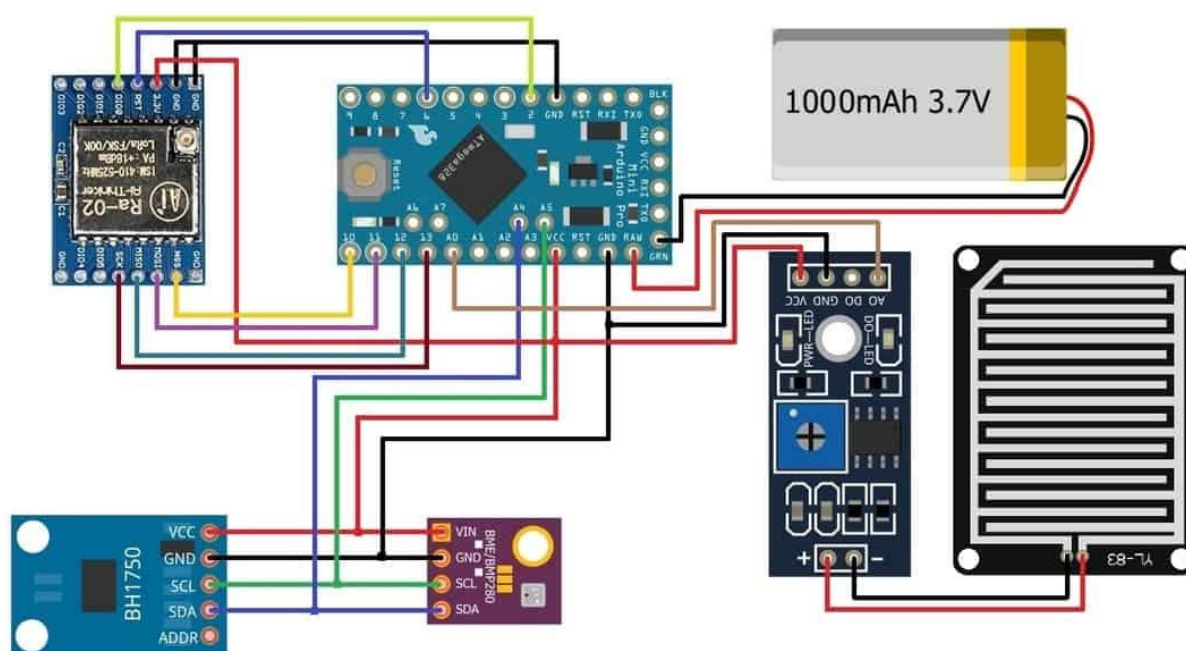
kHz, a dostępne współczynniki rozpraszania leżą pomiędzy 6 – 12. Maksymalna moc wyjściowa nadajnika wynosi 18 dBm (63 mW). Użycie mikrokomputera innego typu, zasilanego napięciem 5 V wymaga dopasowania poziomów z 5 V na 3,3 V dla modułu radiowego. Do obsługi można zastosować koder obrotowy albo przycisk, podłączane do złącza J2.

Układ może być zasilany przez gniazdko mini-USB Arduino albo przez jego kontakty 5 V. W przypadku zasilania modułu radiowego przez gniazdko USB Arduino należy ograniczyć moc wyjściową do 8 dBm, gdyż pobór prądu przy nadawaniu przekraczający 240 mA może powodować spadek napięcia zasilania mikrokomputera i zakłócenia w jego pracy – z ponownymi startami włącznika. Przy większych mocach wyjściowych konieczne jest oddzielne zasilanie napięciem 3,3 V z akumulatora albo zasilacza sieciowego.

Wyświetlacz o przekątnej 0,96 cala jest sterowany przez magistralę I2C i podłączony do złącza J3. Całość można zmontować na płytce drukowanej albo na uniwersalnej płytce eksperymentalnej. W trakcie montażu należy zwrócić szczególną uwagę na to, aby nie przegrzać modułu radiowego, gdyż spowoduje to późniejsze problemy z pracą pętli synchronizacji fazy w układzie syntezy. Może to skutkować znacznymi odchyłkami częstotliwości wyjściowej od nominalnej, dochodzącymi nawet do kilkunastu kHz.

Oprogramowanie Arduino korzysta z bibliotek dla obsługi wyświetlacza i modułu radiowego. Jedną z bibliotek Lory jest dostępna w Internecie pod adresem <https://github.com/sandeepmistry/arduino-LoRa>, a biblioteka dla wyświetlacza – pod adresem www.arduino.cc/reference/en/libraries/adafruit-ssd1306. Biblioteka Lory zawiera funkcje nadawcze i odbiorcze dla rodziny układów SX1276/77/78/79. Przykładowe programy są dostępne w strefie pobrań (download) witryny www.funkamateur.de. Jednym z przykładów jest program radiolatarni LoRa nadający tekst w odstępach 5-sekundowych. Przed uruchomieniem radiolatarni konieczne jest wprowadzenie do programu własnego znaku wywoławczego, częstotliwości nadawania i parametrów transmisji. Innym z przykładów jest program dla odbiornika sygnałów radiolatarni nadający własny tekst po naciśnięciu przycisku. Mini-TRX można także wykorzystać do transmisji własnych współrzędnych geograficznych odczytanych z odbiornika GPS.

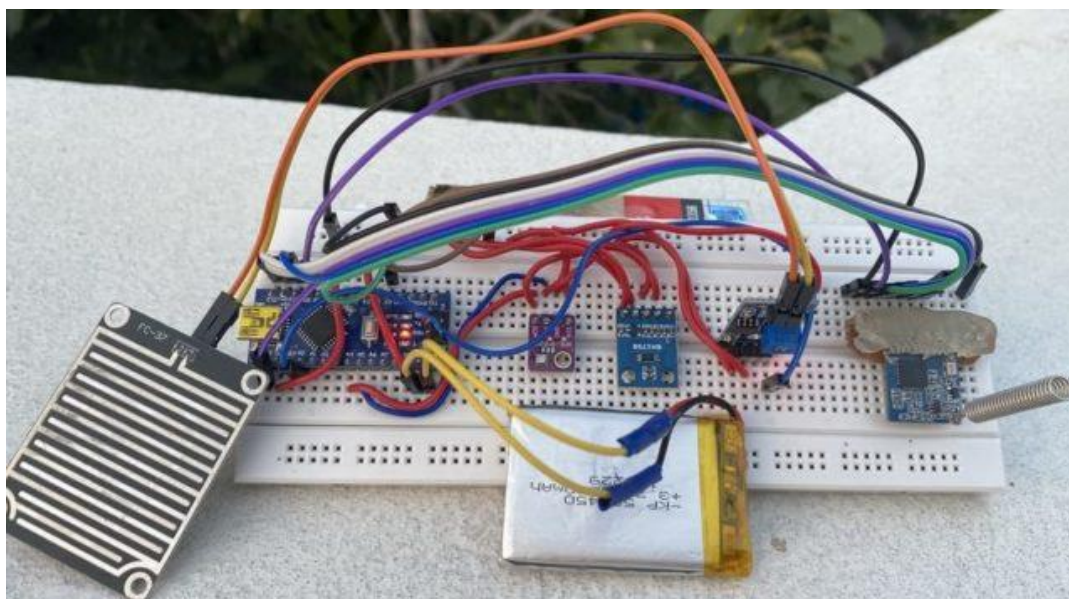
Ostatni z przykładowych programów ilustruje możliwości zdalnego sterowania. Po stronie odbiorczej (sterowanej zdalnie) dioda elektroluminescencyjna (z opornikiem szeregowym 330 Ω) jest włączona między kontaktami 5 V i SCL (A5) na złączu J3. Każdy z przykładów można modyfikować i rozszerzać zgodnie z własnymi potrzebami.



Rys. 1.2.3. Przykładowa realizacja stacji meteorologicznej

W skład stacji z rys. 1.2.3 wchodzi 3,3-voltowy mikrokomputer *Arduino Pro Mini*, czujnik barometryczny BME280, czujnik deszczu FC-37, czujnik oświetlenia BH1750 i moduł radiowy Ra-02

(z SX1278 dla 433 MHz) systemu *LoRa*. Czujniki BH1750 i BME280 są połączone z mikrokomputerem przez magistralę I2C. Do ładowania akumulatora może służyć bateria słoneczna albo moduł ładowania TP4056. Czujnik BH1750 podaje natężenie światła bezpośrednio w luksach – do wartości maksymalnej 65535.

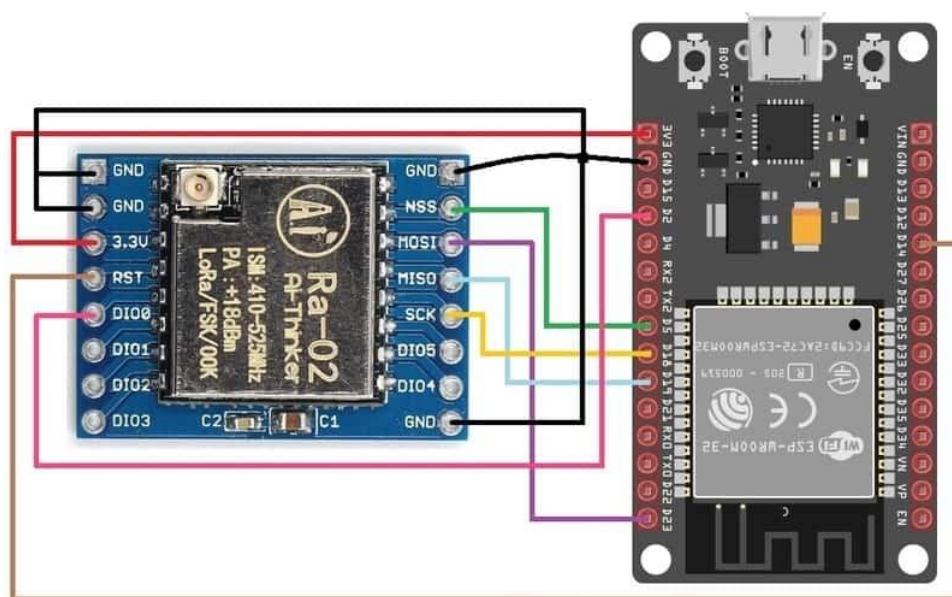


Fot. 1.2.4. Realizacja stacji na płytce próbnej

Przykładowe oprogramowanie stacji meteorologicznej i bramki WiFi wraz niezbędnymi bibliotekami dla Arduino znajduje się pod adresem

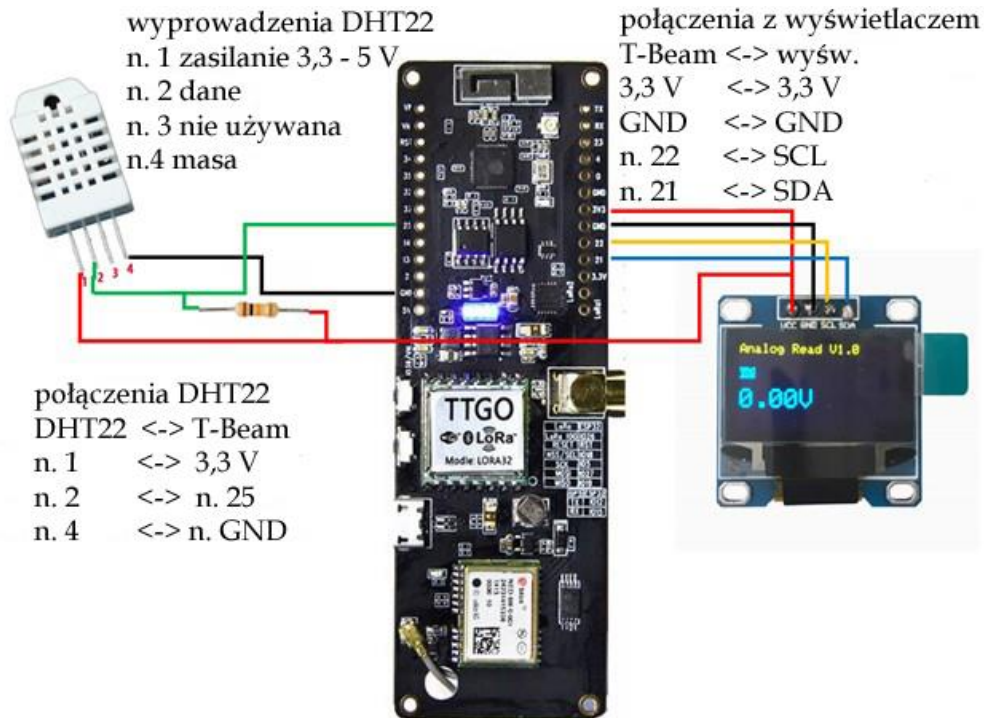
<https://how2electronics.com/lora-based-wireless-weather-station-with-arduino-esp32/>.

Bramka internetowa łączy się w podanym przykładzie z serwerem *Blynk* co pozwala na odczyt danych pomiarowych przez oprogramowanie *Blynk* na telefonie komórkowym. Bramka nie jest potrzebna w przypadku transmisji danych jedynie drogą radiową.



Rys. 1.2.5. Realizacja bramki internetowej na Ra-02 i module WiFi ESP32 (źródło: <https://how2electronics.com/lora-based-wireless-weather-station-with-arduino-esp32/>)

Prosta stacja meteorologiczna konstrukcji OE3CJB składa się z modułu *LoRa* TTGO T-Beam, czujnika temperatury i wilgotności względnej DHT22 i wyświetlacza polimerowego OLED 128 x 64 punkty (<https://oe3.oevsv.at/adl304/technikecke/technik/LoRa-APRS-Tracker/>). Moduł TTGO T-Beam pracuje w paśmie 433 MHz. Zawiera on procesor ESP32, odbiornik GPS NEO-6M i jest zasilany z akumulatora litowo-jonowego typu 18650.

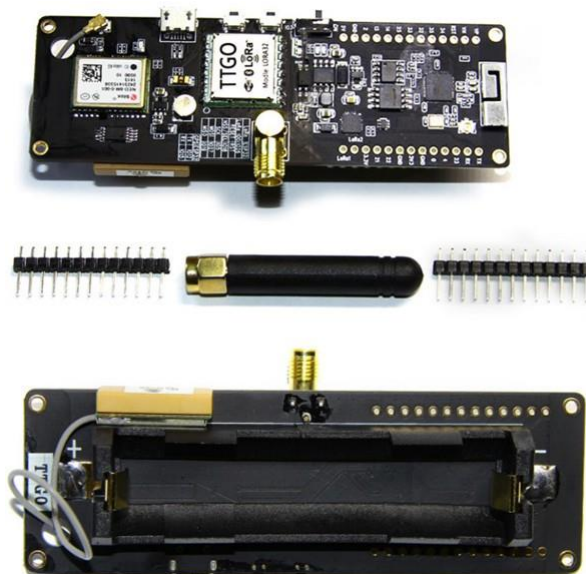


Rys. 1.2.6. Prosta stacja meteorologiczna na *TTGO T-Beam* konstrukcji OE3CJB

W wersji podstawowej bez czujnika DHT22, wyświetlacza i ze standardowym oprogramowaniem może służyć do publikowania współrzędnych geograficznych APRS operatora. Oprogramowanie stacji meteorologicznej z DHT22 jest dostępne w Internecie pod adresem:

<https://github.com/oe3cjb/TTGO-T-Beam-LoRa-APRS>

Jest to wersja eksperymentalna i może służyć za podstawę do dalszych prac i rozbudowy o dodatkowe funkcje.



Fot. 1.2.7. Moduł *TTGO T-Beam*

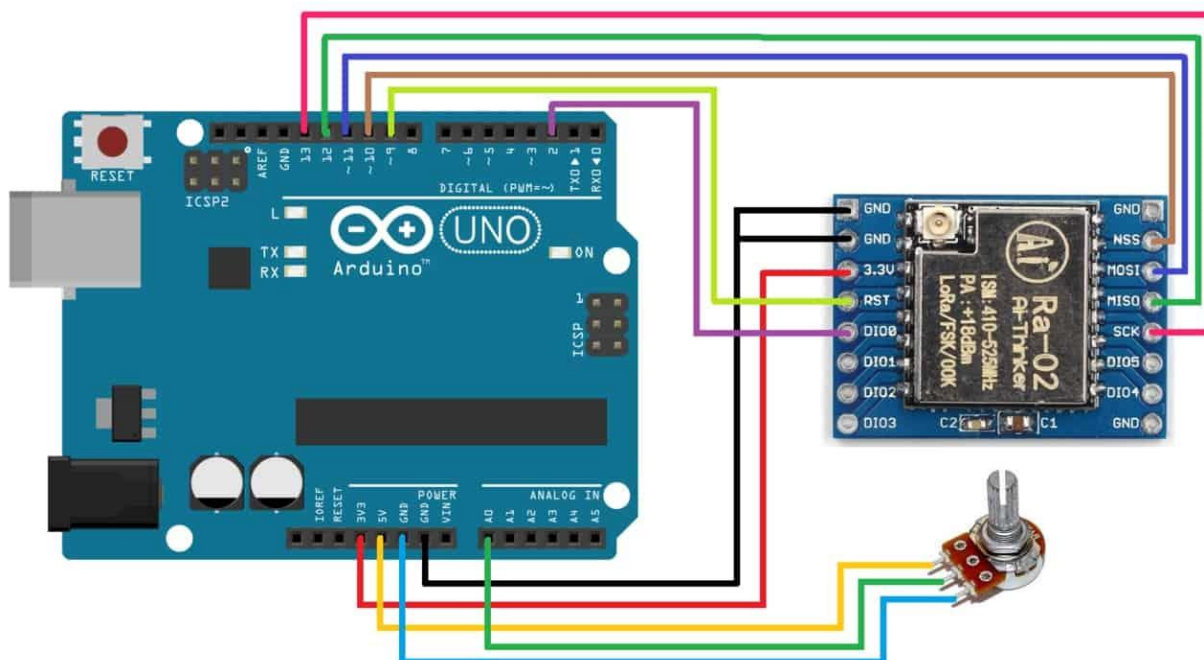
Czujniki są najczęściej połączone z mikrokomputerem za pomocą magistrali I2C lub jej nowszej wersji – Magistrali I3C. W odróżnieniu od wyjść nadawczych z otwartym drenem lub kolektorem wymagających stosowania w układach oporników podciągających (ang. *pull-up*) w magistrali I3C stosowane są wyjścia przeciwsołne (ang. *push-pull*). Zapewnia ona większą szybkość transmisji, pozwala na użycie w sieci nie tylko jednego mikrokomputera zarządzającego (nadrzędnego; ang. *master*) ale także kilku nadrzędnych mikrokomputerów pomocniczych oraz na łączenie elementów współpracujących z magistralą I2C i I3C. W obu magistralach do komunikacji służą przewody SCL i SDA.

Bardziej złożone elementy sieci takie jak wyświetlacze, nadajniki systemu *LoRa* itd. są podłączane za pomocą magistrali SPI. Występują w niej przewody MISO, MOSI, SCK i NSS. Do programowej obsługi obydwu magistral służą gotowe biblioteki dla różnych typów mikrokomputerów.

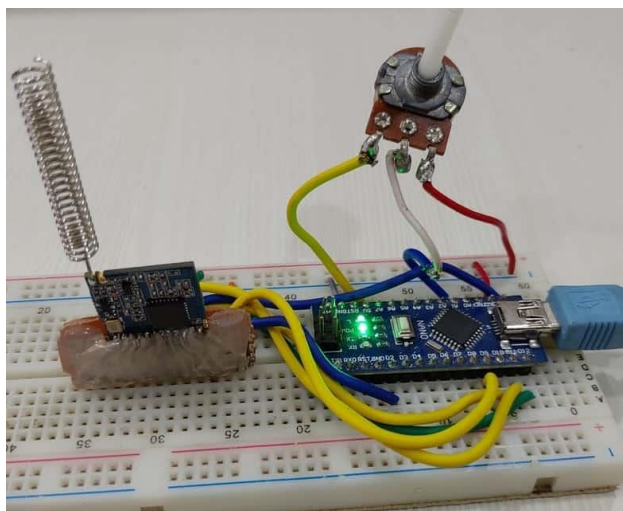
1.2.1. Eksperymenty dla początkujących

W lepszym zrozumieniu działania stacji nadawczo-odbiorczych systemu *LoRa* mogą pomóc zamieszczone poniżej programy dydaktyczne dla *Arduino UNO* połączonego z modułem Ra-02 (SX1278) na pasmo 433 MHz. Zdobyte w ten sposób doświadczenia można wykorzystać też we własnych projektach i zastosowaniach domowych albo we własnej stacji amatorskiej.

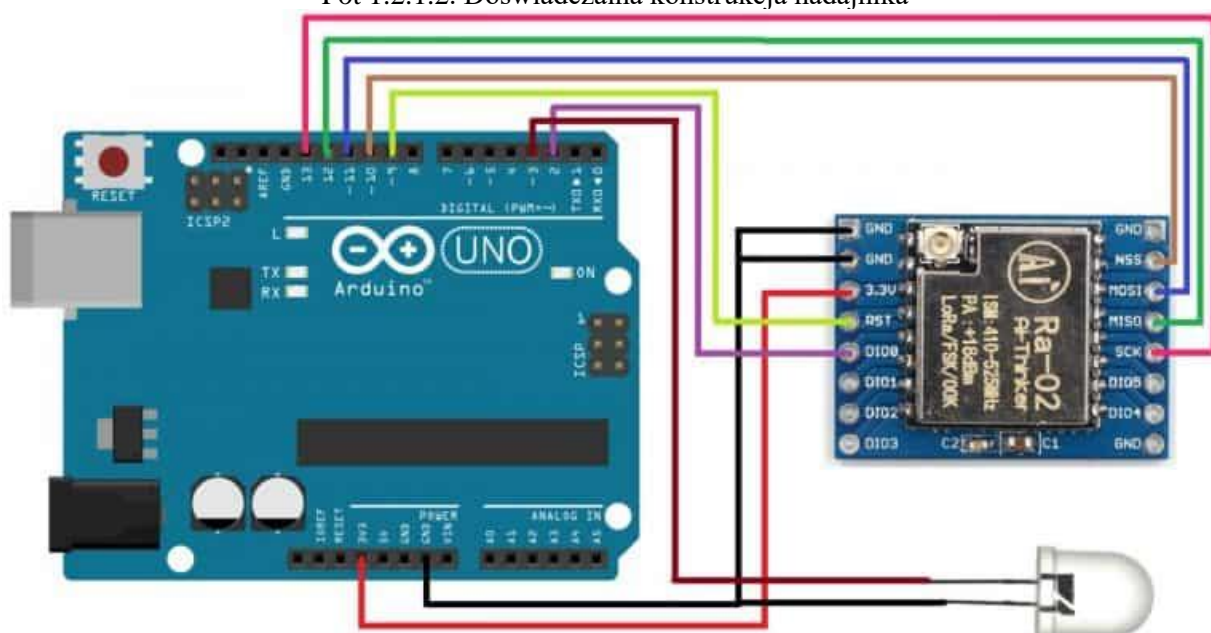
W poniższym przykładzie jasność diody świecącej po stronie odbiorczej jest regulowana za pomocą potencjometru 10 kΩ podłączonego do nadawczego mikrokomputera *Arduino*. Użyta w programie biblioteka obsługuje wszystkie moduły *LoRa* oparte na modemach SX1272, 1273, 1276, 1277, 1278, 1279 albo RFM95, 96, 97 i 98. Mogą one być zamontowane na płytkach rozszerzeń *Arduino* np. *Dragino* itp. Czulości odbiorników modemów SX dochodzą nawet do -148 dBm, a moce wyjściowe do 20 dBm (100 mW). Połączenie elementów nadajnika przedstawia schemat z rysunku 1.2.1.1. Moduł nadawczo-odbiorczy Ra-02 wymaga podłączenia anteny do widocznego na płytce w pobliżu górnej krawędzi gniazdka koncentrycznego. Brak anteny powoduje uszkodzenie nadajnika. Moduł SX1278 wymaga zasilania napięciem 3,3 V. Zasilanie napięciem 5 V spowoduje jego zniszczenie. Do komunikacji między *Arduino* i Ra-02 służy złącze SPI.



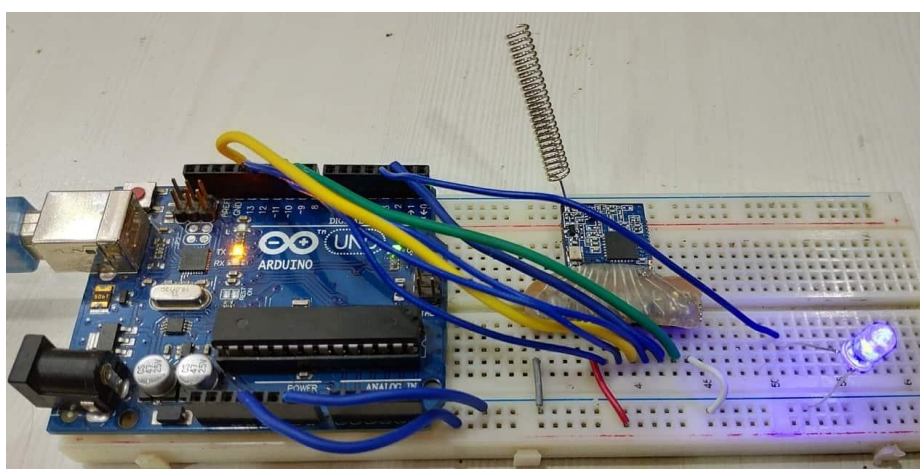
Rys. 1.2.1.1. Połączenie elementów stacji nadawczej



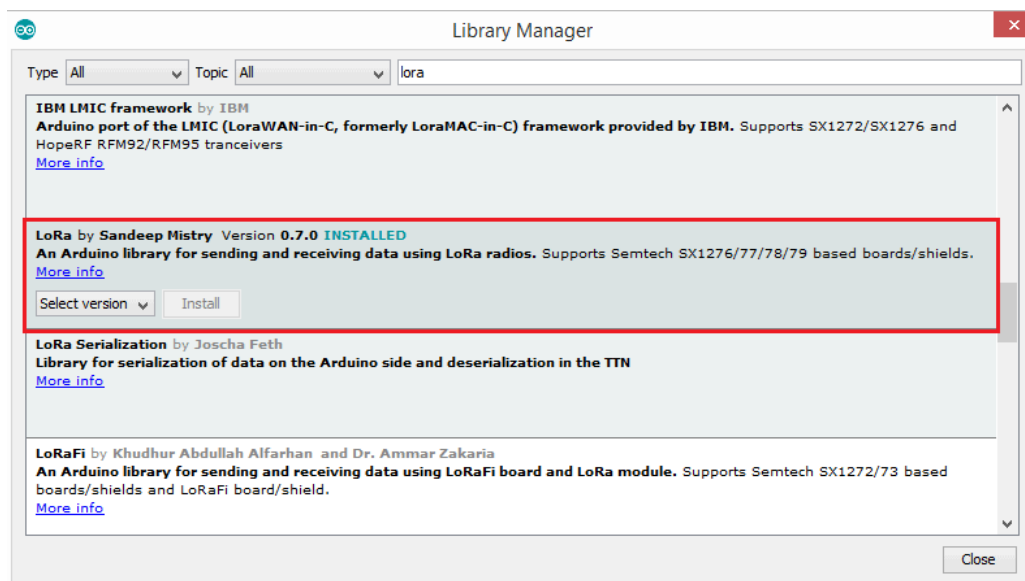
Fot 1.2.1.2. Doświadczalna konstrukcja nadajnika



Rys. 1.2.1.3. Połączenia elementów stacji odbiorczej. Dioda świecąca dowolnego typu jest połączona z wyjściem D3 Arduino



Fot. 1.2.1.4. Doświadczalna konstrukcja odbiornika



Rys. 1.2.1.5. Instalacja biblioteki za pomocą menadżera bibliotek w środowisku Arduino IDE

Program nadawczy

```
#include <SPI.h>
#include <LoRa.h>
int pot = A0;

void setup() {
  Serial.begin(9600);
  pinMode(pot,INPUT);

  while (!Serial);
  Serial.println("LoRa Sender");
  if (!LoRa.begin(433E6)) { // lub 915E6, pasmo pracy modułu w MHz
    Serial.println("Starting LoRa failed!");
    while (1);
  }
}

void loop() {
  int val = map(analogRead(pot),0,1024,0,255);
  LoRa.beginPacket();
  LoRa.print(val);
  LoRa.endPacket();
  delay(50);
}
```

Program odbiorczy

```
#include <SPI.h>
#include <LoRa.h>
int LED = 3;
String inString = ""; // zmienna na dane wejściowe
int val = 0;

void setup() {
```



```

Serial.begin(9600);
pinMode(LED,OUTPUT);

while (!Serial);
Serial.println("LoRa Receiver");
if (!LoRa.begin(433E6)) { // lub 915E6
  Serial.println("Starting LoRa failed!");
  while (1);
}
}

void loop() {

  // składanie pakietu
  int packetSize = LoRa.parsePacket();
  if (packetSize) {
    // odczyt pakietu
    while (LoRa.available())
    {
      int inChar = LoRa.read();
      inString += (char)inChar;
      val = inString.toInt();
    }
    inString = "";
    LoRa.packetRssi();
  }

  Serial.println(val);
  analogWrite(LED, val);
}

```

Po załadowaniu programów obracanie ośki potencjometru powinno powodować zmianę jasności diody. Eksperymenty są przeznaczone dla osób mających już pewną wiedzę na temat programowania *Arduino*.

1.2.2. Dydaktyczny program nadajnika APRS w systemie „Lora”

Program poniższy został opracowany przez OE1KEB, a dodatki zapewniające odczyt i transmisję ciśnienia atmosferycznego pochodzą od OE1KDA. Program jest przeznaczony dla Arduino z przedstawioną powyżej płytką rozszerzeń *LoRa* Dragino. Jest on przytoczony w całości jako program dydaktyczny mający ułatwić poznanie od kuchni działania nadajników danych pozycyjnych i telemetrycznych w systemie *LoRa*. Dane pozycyjne są na stałe wpisane do programu dzięki czemu niepotrzebny jest odbiornik GPS. Zamierzeniem autora nie jest opracowywanie rozwiązań konkurencyjnych w stosunku do rozwiązań powszechnie stosowanych, ale nadajnik z tym oprogramowaniem dobrze się spisuje od lat, jest odbierany przez austriackie przekaźniki APRS, a dane są przekazywane do *aprs.fi*. W programie zademonstrowano możliwość dowolnego dopasowania parametrów transmisji na poziomie podstawowym i wybór czujników (w poniższej konfiguracji jest to BPP180. Przykładowe parametry transmisji odpowiadają stosowanym w Austrii ale ich zmiana nie powinna przysporzyć trudności trochę bardziej doświadczonym programistom. W zależności od sytawianych sobie celów można całkowicie opuścić odczyt danych z czujnika i poprzestać na nadawaniu wyłącznie współrzędnych geograficznych albo podłączyć do Arduino czujnik innego typu.

```

// LoRa 9x_TX
#include <SPI.h>
#include <RH_RF95.h>
// do termometru analogowego nr 10

```

```
#include <math.h>
// do termometru nr 32 - DS18x20
#include <OneWire.h>
// do płytki LoRa
#define RFM95_CS 10
#define RFM95_RST 9
#define RFM95_INT 2
#define LED_BUILTIN 13

#define URUCHAMIANIE 1
//#undef URUCHAMIANIE

//wybor czujnika, wybrac jeden
#undef TEMP32
#undef TEMP10
#define BMP180

// formatowanie komunikatu
#define POCZT 36
#define POCZP 45
//-----
#ifdef TEMP32
// DS18S20 termometr
OneWire ds(6); // połączony z wyprowadzeniem 6

// zmienne do temperatury 32
byte i;
byte present = 0;
byte data[12];
byte addr[8];
// do konwersji
int HighByte, LowByte, TReading, SignBit, Tc_100, Whole, Fract;
#endif
//-----
#ifdef BMP180
#include "Barometer.h"
#include <Wire.h>
float temperature;
float temperatura_srednia;
float pressure;
float cisnienie_srednie;
float atm;
float altitude;
Barometer myBarometer;
#endif
//-----
// Parametry łącza radiowego
#define BW125 112
#define BW250 128
#define BW500 144
#define SF12 12
#define CR45 2
#define CR48 8
#define ARWT 7
#define LDOPTON 8
```

```

// Zmiana na 433.775 lub inna częstotliwość, musi być zgodzić się z odbiorcą!
#define RF95_FREQ 433.775

// sterownik radiowy
RH_RF95 rf95(RFM95_CS, RFM95_INT);

void setup() {
  pinMode(RFM95_RST, OUTPUT);
  digitalWrite(RFM95_RST, HIGH);
  pinMode(LED_BUILTIN, OUTPUT);
  while (!Serial);
  Serial.begin(9600);
  delay(100);
#ifdef URUCHAMIANIE
  Serial.println("Arduino LoRa TX Test!");
#endif
  // manual reset
  digitalWrite(RFM95_RST, LOW);
  delay(10);
  digitalWrite(RFM95_RST, HIGH);
  delay(10);

  while (!rf95.init()) {
    Serial.println("LoRa radio init failed");
    while (1);
  }

  // Ustawienia domyślne 434.000 MHz, 13 dBm, Bw = 125 kHz, Cr = 4/5, Sf = 128 odcinków/symbol,
  CRC włączona
  // Konieczna zmiana na 433,775 MHz, 23 dBm, Bw = 125 kHz, Cr = 4/5, Sf = 4096 bit/symbol, CRC
  włączona
  // Domyślna moc wyjściowa 13dBm, korzystanie z PA_BOOST.
  // przy użyciu modułów RFM95/96/97/98 posiadających wyprowadzenie PA_BOOST
  // moc nadajnika w granicach 5 - 23 dBm:
#ifdef URUCHAMIANIE
  // Rejestry konfiguracyjne po inicjalizacji (pierwsza konfiguracja z tabeli, indeks 0)
  Serial.print ("Rejestr 1d = ");
  Serial.println (rf95.spiRead(RH_RF95_REG_1D_MODEM_CONFIG1));
  Serial.print ("Rejestr 1e = ");
  Serial.println (rf95.spiRead(RH_RF95_REG_1E_MODEM_CONFIG2));
  Serial.print ("Rejestr 26 = ");
  Serial.println (rf95.spiRead(RH_RF95_REG_26_MODEM_CONFIG3));
  Serial.print ("Rejestr 01 (OP_MODE) = ");
  Serial.println (rf95.spiRead(RH_RF95_REG_01_OP_MODE));
#endif
  // Pasma i współczynnik korekcji,
  // Header_Explicit = 0, CRC_off = 0
  rf95.spiWrite(RH_RF95_REG_1D_MODEM_CONFIG1, BW125 + CR45 + 0);
  // Zmiana współczynnika rozpraszania na SF12 - 0xc4
  // ARW + Timeout = 7
  rf95.spiWrite(RH_RF95_REG_1E_MODEM_CONFIG2, SF12 * 16 + ARWT);
  // LowDoptON = 8
  rf95.spiWrite(RH_RF95_REG_26_MODEM_CONFIG3, LDOPTON);
#ifdef URUCHAMIANIE

```

```

// sprawdzenie
Serial.print ("Rejestr 1d = ");
Serial.println (rf95.spiRead(RH_RF95_REG_1D_MODEM_CONFIG1));
Serial.print ("Rejestr 1e = ");
Serial.println (rf95.spiRead(RH_RF95_REG_1E_MODEM_CONFIG2));
Serial.print ("Rejestr 26 = ");
Serial.println (rf95.spiRead(RH_RF95_REG_26_MODEM_CONFIG3));

Serial.println ("Czestotliwosc po starcie ");
Serial.println (rf95.spiRead(RH_RF95_REG_06_FRF_MSB));
Serial.println (rf95.spiRead(RH_RF95_REG_07_FRF_MID));
Serial.println (rf95.spiRead(RH_RF95_REG_08_FRF_LSB));
delay (5000);
#endif
if (!rf95.setFrequency(433.775)) {
#ifdef URUCHAMIANIE
  Serial.println("Set Frequency failed");
#endif
  delay(10);
}
// Częstotliwość: 433775 / 61,03515625 -> 7106969,6 ->
// 7106969 -> 6C 71 47 -> 108 113 47
rf95.spiWrite(RH_RF95_REG_06_FRF_MSB, 108);
rf95.spiWrite(RH_RF95_REG_07_FRF_MID, 113);
rf95.spiWrite(RH_RF95_REG_08_FRF_LSB, 71);
#ifdef URUCHAMIANIE
  // sprawdzenie
  Serial.println ("Czestotliwosc ustawiona ");
  Serial.println (rf95.spiRead(RH_RF95_REG_06_FRF_MSB));
  Serial.println (rf95.spiRead(RH_RF95_REG_07_FRF_MID));
  Serial.println (rf95.spiRead(RH_RF95_REG_08_FRF_LSB));
  Serial.println ("Moc po starcie ");
  Serial.println (rf95.spiRead(RH_RF95_REG_09_PA_CONFIG));
  Serial.println (rf95.spiRead(RH_RF95_REG_4D_PA_DAC));
#endif
rf95.setTxPower(23, false);
#ifdef URUCHAMIANIE
  //Sprawdzenie
  Serial.println ("Moc ustawiona ");
  Serial.println (rf95.spiRead(RH_RF95_REG_09_PA_CONFIG));
  Serial.println (rf95.spiRead(RH_RF95_REG_4D_PA_DAC));
#endif
#ifdef BMP180
  myBarometer.init();
  temperatura_srednia = 0;
  cisnienie_srednie = 0;
#endif
}
// podobno maksymalna długość nadawanych pakietow wynosi 40-45 bajtow
//char meteopacket[] = "OE1KDA-11>APRS:!4808.22N/01620.86E_c...s...g...t-20b10250h..";
char radiopacket[] = "OE1KDA-11>APRS:!4808.22N/01620.86E-";
String temp_string = "";
String cisn_string = "";
String wilg_string = "";

```

```

String nap_string = "";
uint8_t buf[RH_RF95_MAX_MESSAGE_LEN];
uint8_t len = sizeof(buf);

void loop() {
  Serial.println("Sending to rf95_server");
  // Send a message to rf95_server
#ifdef TEMP10
  #ifdef URUCHAMIANIE
    Serial.print(temperatura10(analogRead(0)));
    Serial.println("C");
  #endif
  temp_string = String(temperatura10(analogRead(0)));
  radiopacket[POCZT] = 'T';
  radiopacket[POCZT + 1] = '=';
  for (int i = 0; i <= 3; i++)
    { radiopacket[POCZT + 2 + i] = temp_string[i];
    }
  radiopacket[POCZT + 6] = ' ';
  radiopacket[POCZT + 7] = 'C';
#endif
#ifdef TEMP32
  temperatura32();
  radiopacket[POCZT] = 'T';
  radiopacket[POCZT + 1] = '=';
  for (int i = 0; i <= 3; i++)
    { radiopacket[POCZT + 2 + i] = temp_string[i];
    }
  radiopacket[POCZT + 6] = ' ';
  radiopacket[POCZT + 7] = 'C';
#endif
#ifdef BMP180
  barometrBMP180();
  if (temperatura_srednia == 0)
    { temperatura_srednia = temperature;
    }
  else
    { temperatura_srednia = temperatura_srednia - temperatura_srednia / 8;
      temperatura_srednia = temperatura_srednia + temperature/8;
    }
  temp_string = String(temperatura_srednia);
  if (cisnienie_srednie == 0)
    { cisnienie_srednie = pressure;
    }
  else
    { cisnienie_srednie = cisnienie_srednie - cisnienie_srednie / 4;
      cisnienie_srednie = cisnienie_srednie + pressure / 4;
    }
  cisl_string = String((cisnienie_srednie * 1.021)/100, 1);
  radiopacket[POCZT] = 'T';
  radiopacket[POCZT + 1] = '=';
  for (int i = 0; i <= 3; i++)
    { radiopacket[POCZT + 2 + i] = temp_string[i];
    }
  radiopacket[POCZT + 6] = ' ';

```

```

radiopacket[POCZT + 7] = 'C';
radiopacket[POCZP] = 'P';
radiopacket[POCZP + 1] = '=';
for (int i = 0; i <= 6; i++)
  { radiopacket[POCZP + 2 + i] = (cisl_string[i] != 0) ? cisl_string[i]: ' ';
  }
radiopacket[POCZP + 9] = ' ';
radiopacket[POCZP + 10] = 'h';
radiopacket[POCZP + 11] = 'P';
radiopacket[POCZP + 12] = 'a';
#endif
#ifdef URUCHAMIANIE
  Serial.print("Sending "); Serial.println(radiopacket);
#endif
radiopacket[POCZP + 13] = 0;
#ifdef URUCHAMIANIE
  Serial.println("Sending...");delay(10);
#endif
rf95.send((uint8_t *)radiopacket, 59);
#ifdef URUCHAMIANIE
  Serial.println("Waiting for packet to complete...");delay(10);
#endif
rf95.waitPacketSent();
puls();
// Oczekiwanie na odpowiedź
#ifdef URUCHAMIANIE
  Serial.println("Waiting for reply..."); delay(10);
  if (rf95.waitAvailableTimeout(1000))
  {
    // Po otrzymaniu odpowiedzi
    if (rf95.recv(buf, &len))
    {
      Serial.print("Got reply: ");
      Serial.println((char*)buf);
      Serial.print("RSSI: ");
      Serial.println(rf95.lastRssi(), DEC);
    }
    else
    {
      Serial.println("Receive failed");
    }
  }
  else
  {
    Serial.println("No reply, is there a listener?");
  }
#endif
delay(1000);
}

#ifdef TEMP10
double temperatura10(int RawADC)
{
// odczyt temperatury z czujnika nr 10 podłączonego do A0
double Temp;

```

```

Temp = log(((10240000/RawADC) - 10000));
Temp = 1 / (0.001129148 + (0.000234125 + (0.0000000876741 * Temp * Temp))*
Temp );
Temp = Temp - 273.15;
return Temp;
}
#endif

#ifdef TEMP32
void temperatura32 (void)
{
  ds.reset_search();
  if ( !ds.search(addr) ) {
    Serial.print("No more addresses.\n");
    ds.reset_search();
    return;
  }

  Serial.print("R=");
  for( i = 0; i < 8; i++) {
    Serial.print(addr[i], HEX);
    Serial.print(" ");
  }
  if ( OneWire::crc8( addr, 7) != addr[7]) {
    Serial.print("CRC is not valid!\n");
    return;
  }

  if ( addr[0] == 0x10) {
    Serial.print("Device is a DS18S20 family device.\n");
  }
  else if ( addr[0] == 0x28) {
    Serial.print("Device is a DS18B20 family device.\n");
  }
  else {
    Serial.print("Device family is not recognized: 0x");
    Serial.println(addr[0],HEX);
    return;
  }

  ds.reset();
  ds.select(addr);
  ds.write(0x44,1);    // start konwersji, na końcu włączenie mocy

  delay(1000);    // 750 ms może wystarczyć lub nie
  // można tutaj wywołać ds.depower(), ale wykona to funkcja ds.reset().

  present = ds.reset();
  ds.select(addr);
  ds.write(0xBE);

  Serial.print("P=");
  Serial.print(present,HEX);
  Serial.print(" ");
  for ( i = 0; i < 9; i++) {    // 9 bajtów

```

```

    data[i] = ds.read();
    Serial.print(data[i], HEX);
    Serial.print(" ");
}
Serial.print(" CRC=");
Serial.print( OneWire::crc8( data, 8), HEX);
Serial.println();

// Konwersja
LowByte = data[0];
HighByte = data[1];
TReading = (HighByte << 8) + LowByte;
SignBit = TReading & 0x8000; // sprawdzenie najstarszego bitu
if (SignBit) // wartość ujemna
{
    TReading = (TReading ^ 0xffff) + 1; // uzupełnienie dwójkowe
}
Tc_100 = (6 * TReading) + TReading / 4; // mnożenie przez (100 * 0.0625) czyli 6.25

Whole = Tc_100 / 100; // rozdzielanie części całkowitej i części ułamkowej
Fract = Tc_100 % 100;
if (SignBit) // gdy wartość ujemna
{
    Serial.print("-");
}
Serial.print(Whole);
Serial.print(".");
if (Fract < 10)
{
    Serial.print("0");
}
Serial.print(Fract);

Serial.print("\n");
temp_string = (SignBit)? "-" + String(Whole) : String(Whole);
}
#endif
#ifdef BMP180
void barometrBMP180(void)
{
    temperature = myBarometer.bmp085GetTemperature(myBarometer.bmp085ReadUT()); // odczyt
    temperature, bmp085ReadUT należy wywołać najpierw
    pressure = myBarometer.bmp085GetPressure(myBarometer.bmp085ReadUP()); // odczyt ciśnienia
    altitude = myBarometer.calcAltitude(pressure); // wysokość na podstawie wartości nie
    skompensowanej
    atm = pressure / 101325;
    Serial.print("Temperature: ");
    Serial.print(temperature, 2); // wyświetlenie dwóch miejsc po przecinku
    Serial.println(" deg C");

    Serial.print("Pressure: ");
    Serial.print(pressure, 0); // tylko liczba całkowita.
    Serial.println(" Pa");

    Serial.print("Ralated Atmosphere: ");

```



```

Serial.println(atm, 4); //wyświetlenie 4 miejsc po przecinku

Serial.print("Altitude: ");
Serial.print(altitude, 2); //wyświetlenie dwóch miejsc po przecinku
Serial.println(" m");

Serial.println();

delay(1000); //odczekanie sekundy

}
#endif
void puls(void) {
  char i;
#ifdef URUCHAMIANIE
  digitalWrite (LED_BUILTIN, HIGH);
  delay (5000);
  digitalWrite(LED_BUILTIN, LOW);
  delay (5000);
#else
  for(i = 0;i<=60; i++)
  {
    digitalWrite (LED_BUILTIN, HIGH);
    delay (5000);
    digitalWrite(LED_BUILTIN, LOW);
    delay (5000);
  }
#endif
}

```

W przypadku korzystania z czujnika temperatury i wilgotności względnej DHT11 konieczne jest dodanie pliku nagłówkowego i podanych niżej deklaracji

```

#include "DHT.h"
#define DHTPIN A0 // wejście analogowe dla
#define DHTTYPE DHT11 // DHT 11
DHT dht(DHTPIN, DHTTYPE);
Oraz w void setup()
inicjalizacji
dht.begin();

```

Do odczytu danych z DHT11 służą polecenia gdzie temp jest zmienną typu float, a hum – typu int.

```

temp = dht.readTemperature();
hum = dht.readHumidity();

```

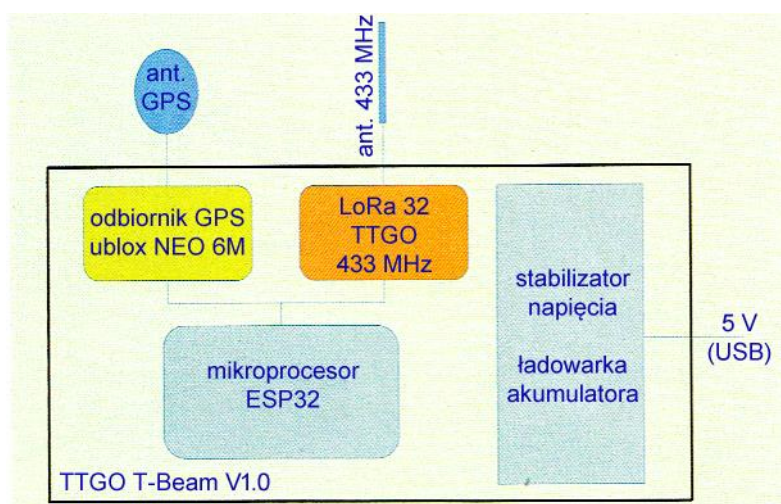
1.3. LoRa APRS

Od początku istnienia systemu APRS komunikaty są transmitowane na kilku ogólnie znanych częstotliwościach w pasmach 2 m i 70 cm emisją AX.25 AFSK – a więc przy użyciu dowolnych radiostacji FM. Szybkość transmisji wynosi standardowo 1200 bit/s. Na częstotliwościach krótkofalowych stosowana jest szybkość transmisji 300 bit/s. W transmisji LoRa APRS używane są częstotliwości pracy w podzakresach 433 i 434 MHz.

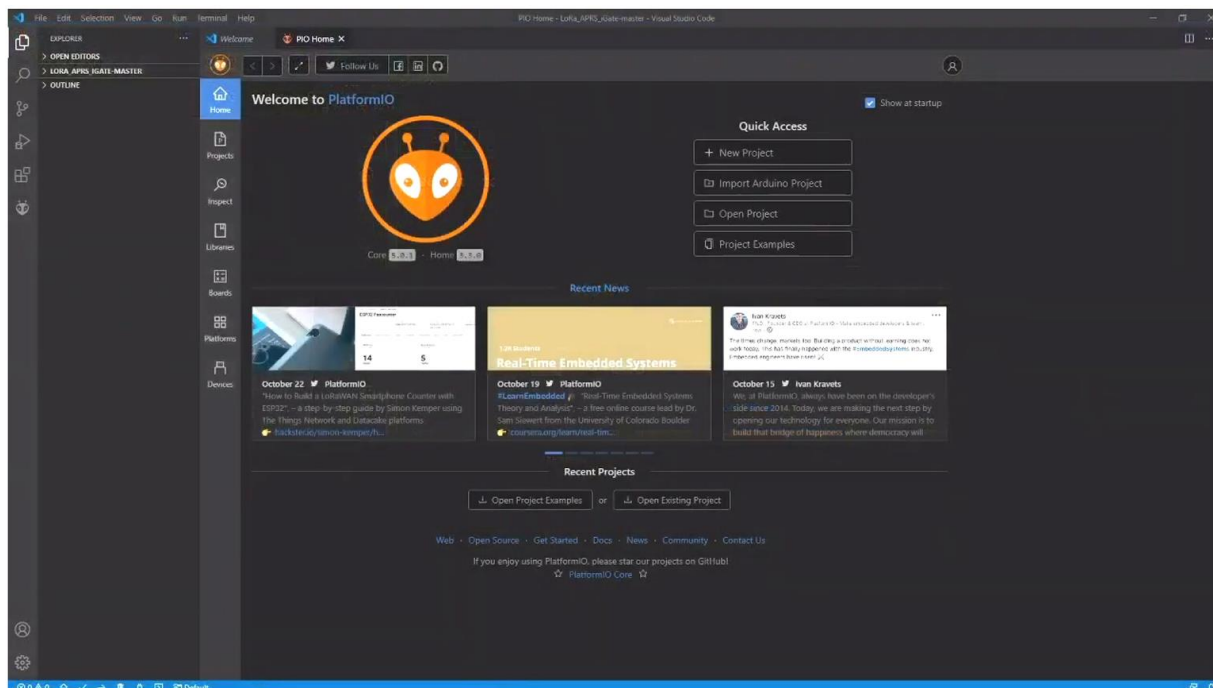
W klasycznym systemie komunikaty APRS nadawane przez stacje indywidualne są retransmitowane przez cyfrowe stacje przekaźnikowe (ang. *digipeater*) i dzięki temu są odbieralne na pewnym obszarze. W szczęśliwej sytuacji docierają one do bramek radiowo-internetowych skąd trafiają do serwerów APRS-IS i są widoczne na tle map w Internecie. Najpopularniejszą z takich witryn jest *aprs.fi*. Każda ze stacji indywidualnych może też służyć jako (uzupełniająca) stacja przekaźnikowa retransmitująca komunikaty do publicznych stacji przekaźnikowych i bramek internetowych.

Stopniowo konkurentem tego rozwiązania staje się transmisja komunikatów w systemie *LoRa* – pracującym z rozpraszaniem widma sygnału. W zastosowaniach krótkofalarskich system pracuje w paśmie 70 cm z mocami nadajników nie przekraczającymi przeważnie 100 mW. Dzięki rozpraszaniu widma sygnału uzyskiwane są zasięgi od kilku do kilkunastu i więcej kilometrów. Szybkości transmisji są naogół niskie i takie są też ilości przekazywanych danych. Łączność przebiega więc według zasady „powoli ale skutecznie”. Poprzez bramki radiowo-internetowe komunikaty docierają, tak jak w pierwszym przypadku, do serwerów APRS-IS i są wyświetlane na mapach. W odróżnieniu od rozwiązania klasycznego radiowa retransmisja komunikatów przez bramki internetowe i stacje indywidualne nie jest zalecana gdyż zwiększa tylko niebezpieczeństwo kolizji ograniczając w ten sposób skuteczną przepustowość kanału.

Do najważniejszych elementów sieci APRS-LoRa należą nadajniki komunikatów pozycyjnych (ang. *tracker*) i bramki radiowo-internetowe (ang. *iGate*). Stosunkowo niewielkim kosztem i przy małym zapotrzebowaniu na energię – pozwalającym na zasilanie z akumulatorów – uzyskuje się sieci przydatne także w ramach akcji ratunkowych i łączności kryzysowych. Bramki internetowe pracują wyłącznie odbiorczo i mogą być uruchamiane również przez nasłuchowców i inne osoby zainteresowane, ale nie posiadające (jeszcze) licencji. Stacje meldujące pozycję pracują nadawczo i do ich uruchomienia konieczna jest licencja amatorska. W ich konfiguracji konieczne jest podanie znaku wywoławczego operatora.



Rys. 1.3.1. Schemat blokowy płytki TTGO T-Beam (źródło: [1.3.7])

Rys. 1.3.2. Okno główne środowiska *PlatformIO*

W obu rodzajach stacji stosowane są moduły mikrokomputerów z procesorami ESP32 i nadawczo-odbiorcze moduły LoRa (modemy) SX1278. Mogą być to przykładowo modele *TTGO T-Beam V1.1* ESP32, *TTGO LoRa32*, *Helltec WiFi LoRa 32* itd. W bramkach internetowych znajdują zastosowanie moduły bez odbiorników GPS – wystarczy wprowadzenie stałych współrzędnych w konfiguracji, natomiast nadajniki komunikatów pozycyjnych powinny być wyposażone w odbiorniki satelitów GPS. Do połączenia z Internetem bramki korzystają ze złącza WiFi. Pierwszy z wymienionych typów posiada odbiornik GPS i znajduje zastosowanie w stacjach meldujących położenie, a dwa dalsze jako nie wyposażone w GPS nadają się głównie do pracy w bramkach. Nic nie stoi jednak na przeszkodzie wykorzystaniu pierwszego modelu do pracy w bramkach, o ile zakup większej liczby identycznych urządzeń okaże się korzystniejszy finansowo. Modele *T-Beam* posiadają pojemniki dla akumulatorów typu 18650 i mogą dzięki temu być zainstalowane w miejscach, w których niemożliwe jest zasilanie z sieci. Jako obudowę można wykorzystać wprawdzie dowolne pudełko o pasujących wymiarach, ale w Internecie oferowane są obudowy dopasowane do wymiarów różnych typów płytek. Możliwe jest również ich samodzielne wydrukowanie na drukarce trójwymiarowej (3-D) w oparciu o dostępne w Internecie pliki konstrukcyjne.

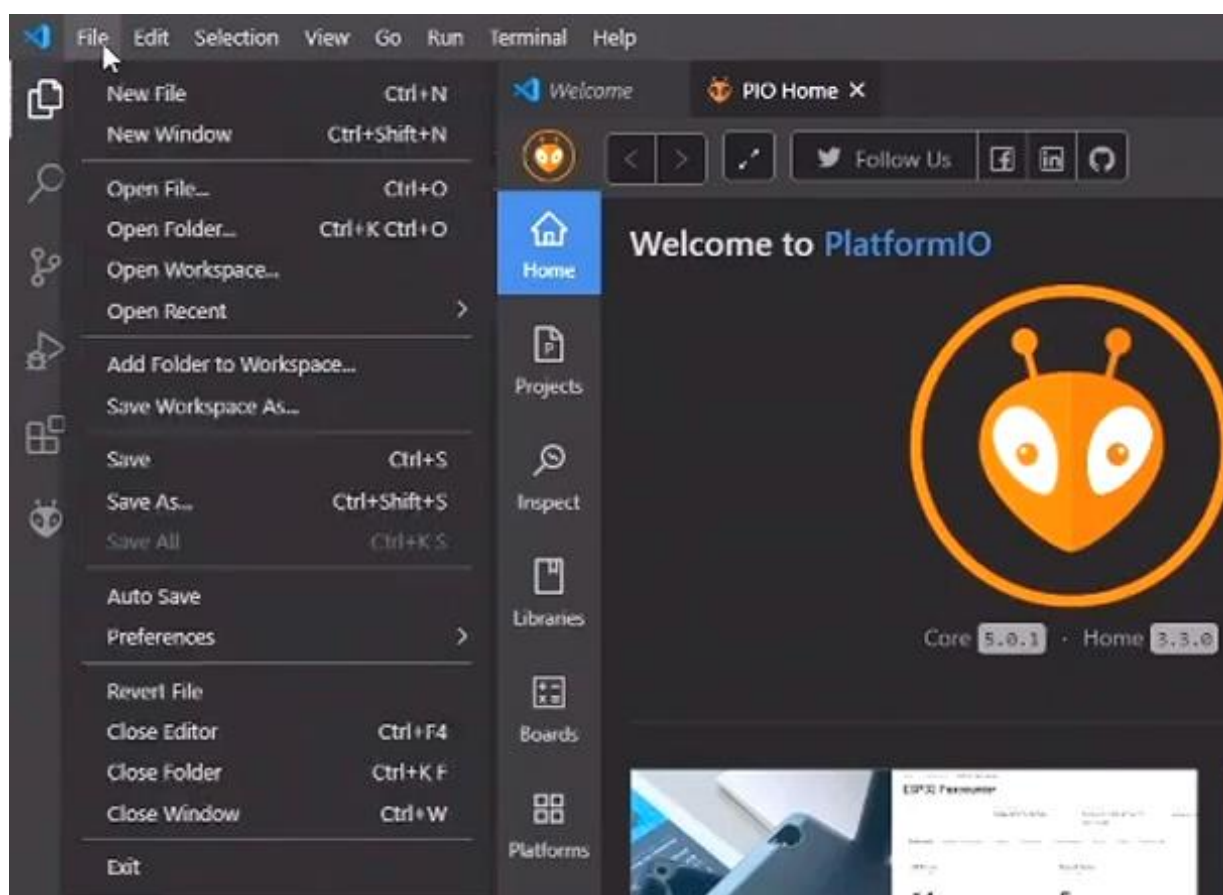
Praktyczne jest, gdy posiadają one wyświetlacz – nawet tylko o rozdzielczości 128 x 64 punkty. Część modułów jest fabrycznie wyposażona w wyświetlacze, ale ewentualne podłączenie go we własnym zakresie nie jest sprawą skomplikowaną. Konieczne jest połączenie masy, zasilania oraz sygnałów SDA i SCL złącza I2C – czyli w sumie czterech przewodów.

Oferowane w zestawach anteny bywają niedostrojone dokładnie do częstotliwości pracy, korzystne może okazać się zastąpienie ich przez skuteczniejsze. Dla upewnienia się czy warto je zamienić można zmierzyć rezonans i przebieg WFS w zakresie pasma 70 cm korzystając z analizatora antenowego.

Komunikaty APRS można uzupełnić o dane telemetryczne takie jak ciśnienie atmosferyczne, temperaturę i wilgotność względną po dodaniu czujnika BME280 lub podobnego, o inne wielkości pomiarowe zależne od typu czujnika (np. również napięcie zasilania) albo o dowolne teksty. Niektóre wersje oprogramowania oferują standardowo takie możliwości. Dla części z nich dostępne są także kody źródłowe co pozwala bardziej doświadczonym programistom na dodanie dowolnej pożądanej funkcjonalności.

Oprogramowanie stacji meldującej pozycję i bramki autorstwa OE5BPA jest dostępne odpowiednio pod adresami [1.3.2] i [1.3.3]. Spakowane archiwum programu jest pobierane po naciśnięciu w oknie *githuba* zielonego przycisku „Code” i wybraniu w menu punktu „Download ZIP”. Archiwum należy następnie rozpakować do dowolnego katalogu na PC. Oprogramowanie nie posiada serwera HTTP i dlatego konieczne jest wprowadzenie parametrów konfiguracyjnych w kodzie źródłowym.

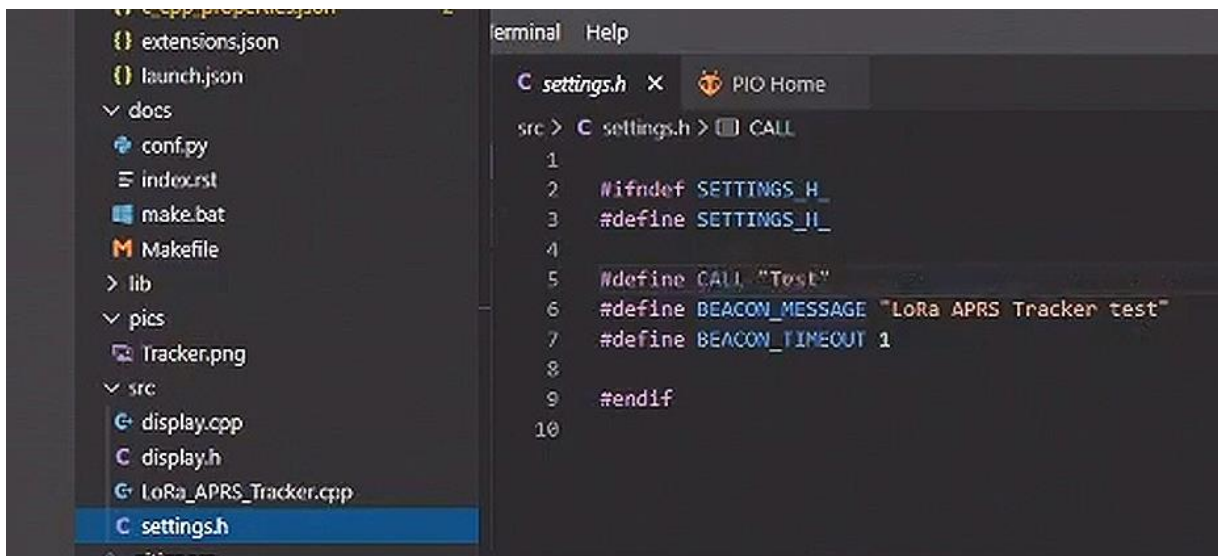
Wpisanie oprogramowania do mikrokomputera wymaga uprzedniego zainstalowania na PC najpierw środowiska programistycznego *Visual Studio Code* (VSCode), a następnie – *PlatformIO* IDE [1.3.4]. Konieczne może być też zainstalowanie sterowników dla złącza USB: TTGO POE Board CH340, Heltec CP210X itp. Podane w objaśnieniach na stronach [1.3.2] lub [1.3.3] dodatkowe biblioteki (j.np. APRS-Decoder-Lib) są pobierane automatycznie przez *PlatformIO* w trakcie kompilacji. Przeważnie znajdują się one na Githubie. Oprócz wersji dla systemu Windows istnieje również wersja dla Linuksa. Po otwarciu *PlatformIO* (rys. 1.3.2) należy przez menu „File | Open folder” („Plik | Otwórz katalog”) wybrać katalog z rozpakowanym programem (nadajnika lub bramki w zależności od realizowanej konstrukcji) – patrz rys. 1.3.3. Program w nim zawarty jest wówczas ładowany do środowiska programistycznego. Po załadowaniu w lewym oknie eksploratora wyświetlany jest spis plików programu (struktura projektu). Z dzungli plików należy wybrać jedynie plik nagłówkowy *settings.h*. Jego treść zostaje wyświetlona w prawym oknie. Plik zawiera kilka poleceń *#define*, w których należy podać własny znak wywoławczy, część tekstową komunikatu APRS i odstęp czasu między transmisjami (rys. 1.3.4).



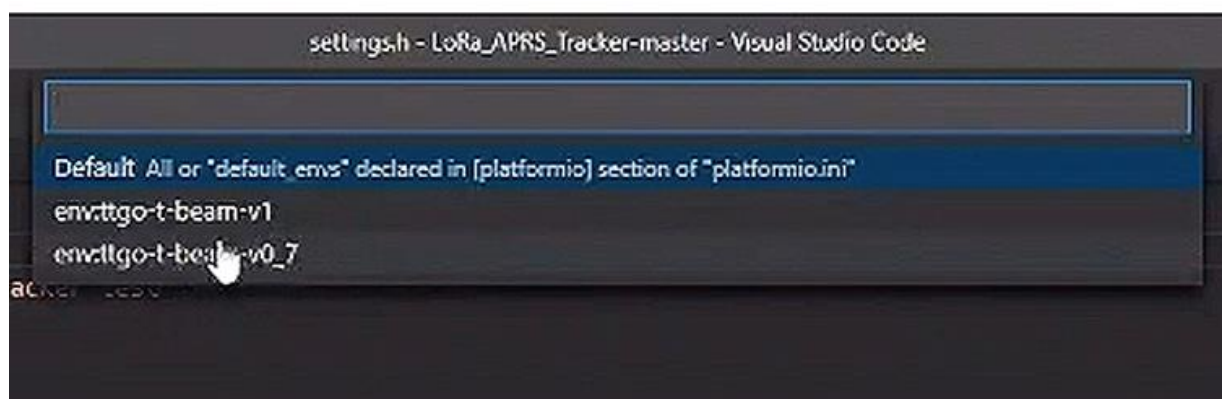
Rys. 1.3.3. Otwieranie katalogu programu

Kolejnym krokiem jest wybór typu mikrokomputera. Po naciśnięciu napisu „Default” w niebieskiej linii informacyjnej u dołu okna otwierane jest u góry okienko wyboru wersji (rys. 1.3.5). Program nadawczy może pracować zarówno na TTGO V0.7 jak i V1. Należy wybrać wersję zgodną z posiadaną płytką. W celu skompilowania i załadowania programu do pamięci mikrokomputera należy nacisnąć w dolnej niebieskiej linii strzałkę skierowaną w prawo (rys. 1.3.6). Płytkę mikroprocesora powinna zostać podłączona do PC przez złącze USB zaraz po otwarciu programu *PlatformIO*. Ładowanie programu trwa dobrą chwilę. Po jego zakończeniu można odłączyć mikroprocesor od PC i jest on już gotowy do użycia. Po pierwszym włączeniu albo po dłuższym czasie nieużywania proces synchronizacji odbiornika GPS może trwać 10 – 15 minut. Transmisja komunikatów APRS rozpoczyna się dopiero po odebraniu kompletnych danych pozycyjnych z satelitów GPS. W przypadku wystąpienia błędów w trakcie ładowania najlepiej jest skasować zawartość pamięci mikrokomputera (za pomocą funkcji „Erase Flash” środowiska programistycznego) i załadować program ponownie.

Każda późniejsza zmiana parametrów konfiguracyjnych wymaga ponownej kompilacji i załadowania programu do pamięci mikrokomputera.



Rys. 1.3.4. Plik *settings.h*



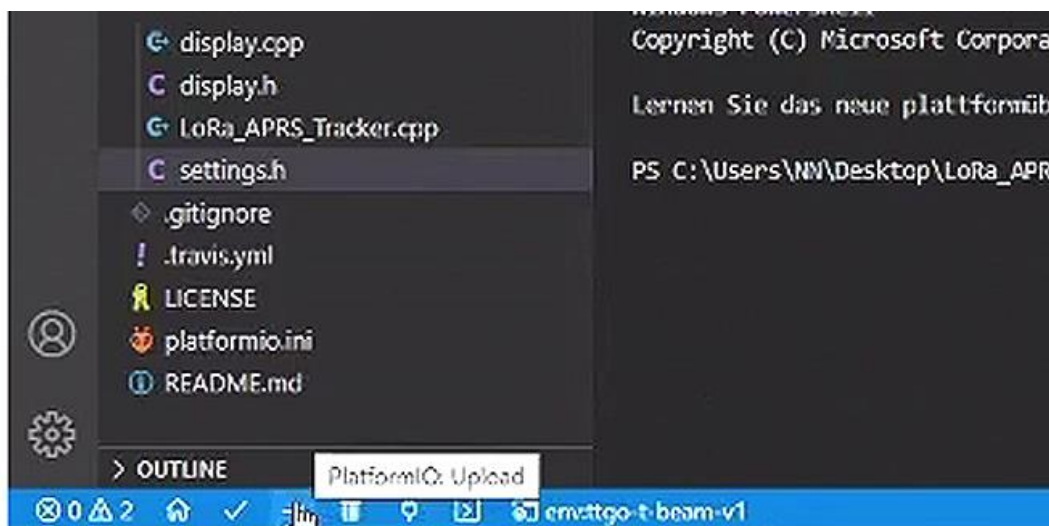
Rys. 1.3.5. Wybór wersji mikrokomputera

Analogicznie przebiega ładowanie oprogramowania bramki internetowej. Wybór modułów mikroprocesorów jest szerszy, ponieważ nie muszą one posiadać odbiornika GPS i pojemnika dla akumulatora (są przeważnie instalowane w miejscach, gdzie nie ma problemów z zasilaniem z sieci). Współrzędne geograficzne stacji są podawane w konfiguracji oprogramowania. Dane konfiguracyjne są zawarte w pliku *is-cfg.json*. Należą do nich znak wywoławczy lub nazwa stacji (w przypadku osób nielicencjonowanych), dane dostępowe do lokalnej sieci WLAN, komunikat APRS, współrzędne geograficzne stacji, hasło dostępu do sieci APRS-IS, adres serwera sieci i numer kanału logicznego. Jako serwer dostępowy APRS-IS można pozostawić *euro.aprs2.net*. Generator hasła dostępu w oparciu o znak lub nazwę stacji znajduje się w Internecie pod adresem [1.3.5] (rys. 1.3.8). Po załadowaniu programu identycznie jak w pierwszym przypadku należy dodatkowo załadować plik konfiguracyjny *is-cfg.json*. W tym celu po naciśnięciu symbolu główki zagadkowego stworzenia konieczne jest wybranie w lewym oknie eksploratora używanej płytki mikrokomputera (np. *envttgo-lora32-v2*) w przykładzie z fot. 1.3.2, następnie należy przejść do punktu „Platform” poniżej i wybrać „Upload Filesystem Image” (rys. 1.3.7). W przypadku pojawienia się na ekranie meldunku błędu najczęściej pomaga ponowne załadowanie pliku konfiguracyjnego. Po wyświetleniu meldunku o braku jakiejś biblioteki trzeba dopisać jej nazwę w pliku *platformio.ini*.

O ile w klasycznym systemie APRS bramki radiowo-internetowe były instalowane w miejscach dogodnych i zapewniających możliwie duże zasięgi, o tyle w sieci APRS-LoRa korzystniejsza jest instalacja możliwie dużej liczby bramek o niedużych zasięgach – nawet przy oknie mieszkania lub na balkonie. Zmniejsza się w ten sposób niebezpieczeństwo kolizji pakietów danych. Stacje stałe powinny z tego

samego względu nadawać komunikaty nawet co 20 minut lub rzadziej. Stacje pracujące z samochodu nie powinny nadawać częściej niż co minutę – najlepiej, aby odstęp czasu zależał od szybkości ruchu. Nie powinny one retransmitować radiowo odebranych komunikatów aby nie zwiększać obciążenia kanałów radiowych. Bramki uruchamiane przez osoby nie posiadające licencji tym bardziej nie powinny pracować nadawczo. Korzystne jest, aby bramki pracowały na okrągło bez przerw. O ile w przypadku stacji nadawczych meldujących radiowo swoje położenia praca bezobsługowa (bez obecności operatora w pobliżu) wymaga uzyskania specjalnej licencji, o tyle bramki czysto odbiorcze mogą pracować bez nadzoru.

W Austrii i w Niemczech w sieci APRS LoRy stosowana jest częstotliwość 433,775 MHz, współczynnik rozpraszania (SF w terminologii *LoRy*) 12, szerokość pasma 125 kHz, szybkość transmisji 183 bit/s i stopa danych korekcyjnych CR 4/5 co oznacza położenie nacisku na większe zasięgi i niezawodność transmisji. Często stosowanym rozszerzeniem dla stacji indywidualnych jest -7, a dla bramek internetowych -10. Standardowym symbolem dla stacji indywidualnych jest symbol pieszego. Można go oczywiście zmienić na dowolny inny pasujący do sytuacji. W Polsce stosowana jest (w chwili powstania niniejszego skryptu) częstotliwość 434,855 MHz, szerokość pasma 125 kHz, stopa danych korekcyjnych CR 4/7 i współczynnik rozpraszania SF równy 9 co zapewnia szybkość transmisji netto zbliżoną do 1200 bit/s (jak w klasycznym wydaniu APRS). Szczegółowej analizie doboru parametrów transmisji i uzyskiwanych wyników poświęcony jest punkt 1.1. Ponieważ krótkofalarstwo jest służbą eksperymentalną zachęcamy użytkowników do wypróbowania innych zestawów parametrów.



Rys. 1.3.6. Ładowanie programu do pamięci mikrokomputera

Oprogramowanie OE5BPA jest jednym z możliwych programów dla stacji APRS pracujących w systemie LoRa. Jest ono znane i rozpowszechnione nie tylko w Austrii ale i w innych krajach. W Polsce znane jest oprogramowanie SQ9MDD oparte na opracowaniu austriackich krótkofalowców OE1ACM i OE3CJB. Sprawa będzie się z pewnością dalej rozwijać dynamicznie i należy spodziewać się powstawania nowych programów i udoskonalonych wersji obecnych. Postaramy się uwzględniać je w przyszłych publikacjach.

Oprogramowanie SQ9MDD pracuje na modułach *TTGO T-beam* v.0.7 i v.1.0 oraz Lora32. Oferuje ono również funkcjonalność TNC w trybie KISS (rys. 1.3.9).

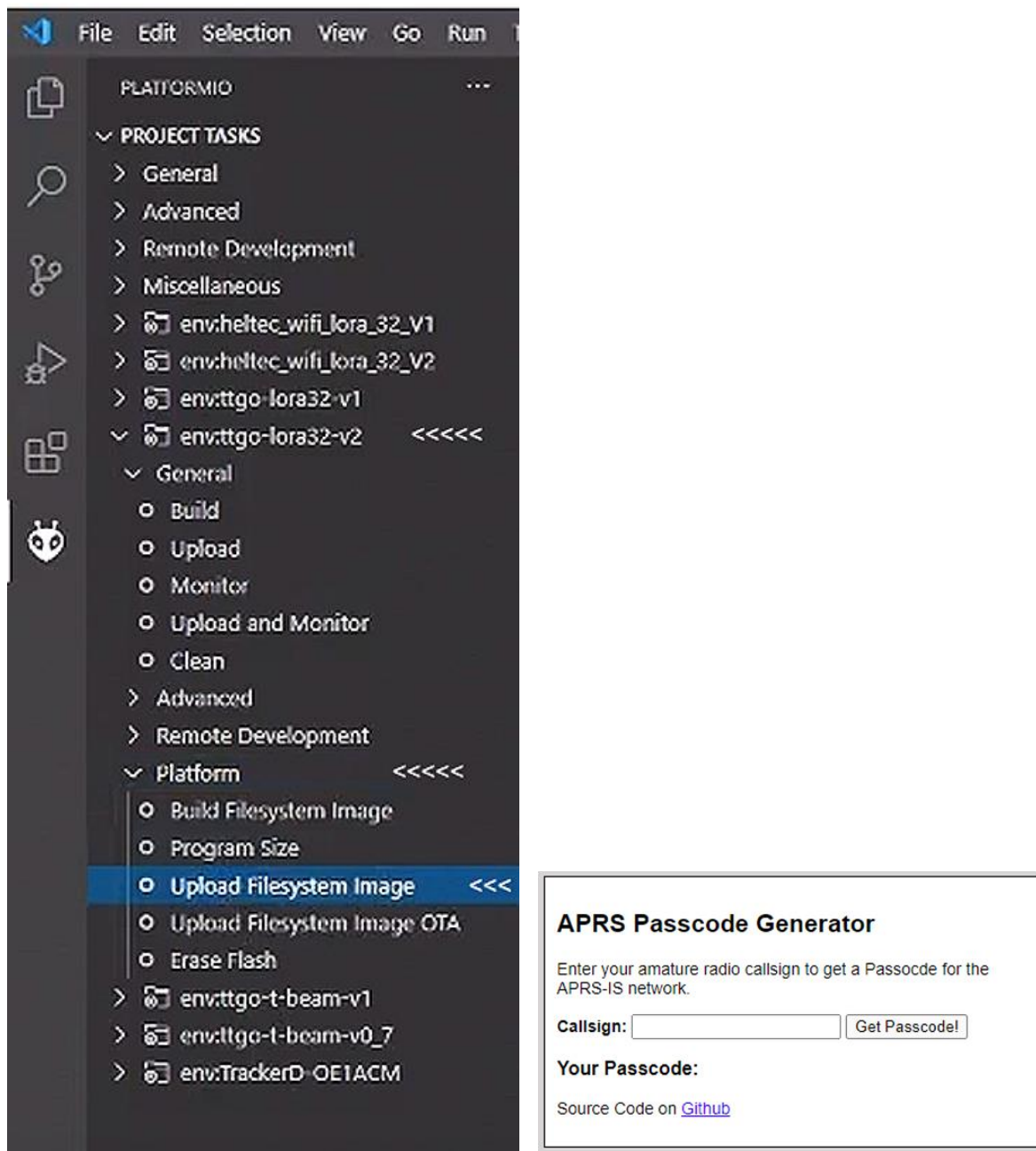
Uruchomienie programu na modułach LoRy wymaga pobrania go z githuba ze strony:

<https://github.com/SQ9MDD/TTGO-T-Beam-LoRa-APRS/releases> a następnie zainstalowania programu ładującego esp-home ze strony <https://github.com/esphome/esphome-flasher/releases>.

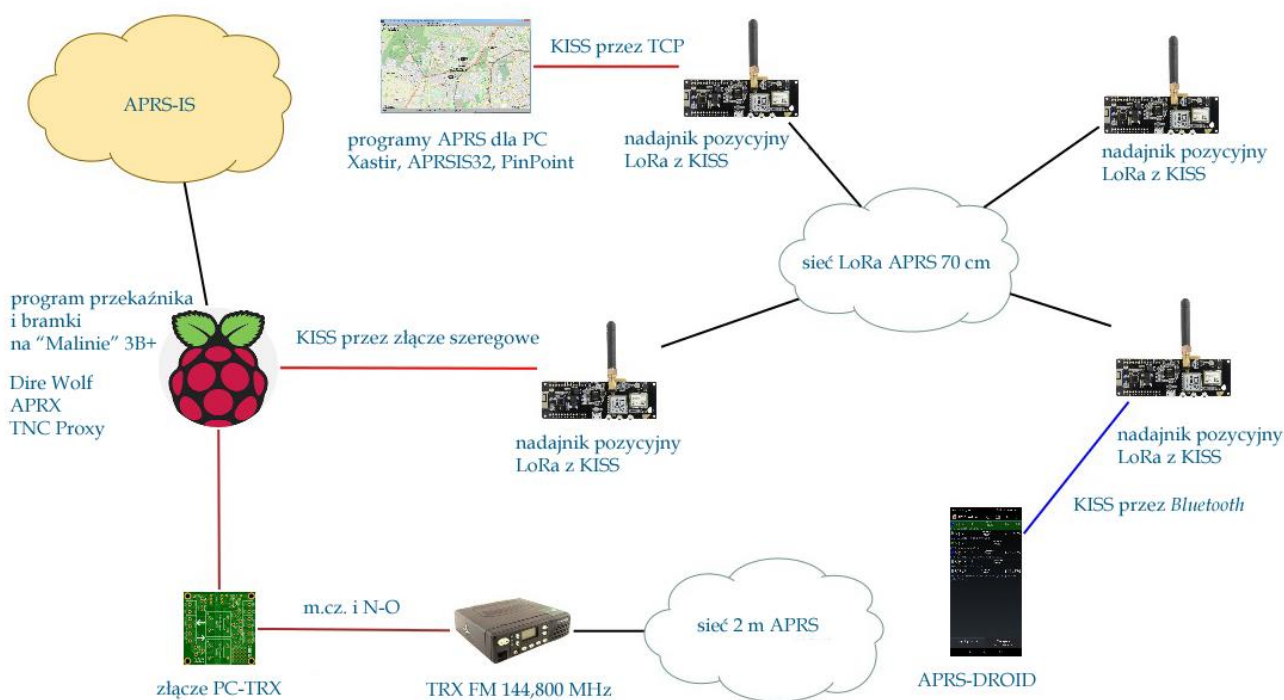
Następnie należy podłączyć moduł LoRy do złącza USB i wywołać program ładujący. Dla zapewnienia komunikacji z mikroprocesorem konieczne może być też zainstalowanie sterownika CP210X. W programie ładującym należy wybrać złącze szeregowo, do którego jest podłączony moduł i plik oprogramowania pobrany z githuba, a na koniec nacisnąć przycisk ekranowy „Flash ESP”.

Do zainstalowania oprogramowania z plików źródłowych konieczne jest, jak w przypadku programu OE5BPA, użycie środowiska *Visual Studio Code* i uzupełnienia *PlatformIO*.

Po otwarciu *PlatformIO* należy wczytać kod źródłowy przy użyciu punktu „Import Arduino Project”, potem nacisnąć na symbol mrówki i na punkt ładowania „Upload”. Uruchomienie programu wymaga połączenia się z modułem i skonfigurowania dostępu do lokalnej sieci bezprzewodowej WiFi przez podanie nazwy sieci i hasła dostępu. Konfiguracja parametrów APRS wymaga podania częstotliwości nadawania i odbioru, szybkości transmisji (domyślnie jest to 300 bodów), wprowadzenia znaku wywoławczego stacji z rozszerzeniem, wybrania symbolu stacji oraz wyboru treści komunikatu i wprowadzenia jego tekstu. Konieczne jest też ustalenie odstępów czasowych między kolejnymi transmisjami.



Rys. 1.3.7. Dodatkowy krok przy ładowaniu konfiguracji bramki radiowo-internetowej (po lewej)
 Rys. 1.3.8. Okno generatora hasła dostępu do APRS-IS. Należy wpisać w nim znak wywoławczy lub przyjętą nazwę bramki i nacisnąć przycisk „Get Passcode!” (po prawej). Niektóre ze spotykanych w Internecie generatorów haseł dostępu generują dla znaków stacji z rozszerzeniem inne hasła niż dla znaku podstawowego. Hasła te nie są przyjmowane przez sieć APRS-IS



Rys. 1.3.9. Możliwe sposoby wykorzystania oprogramowania SQ9MDD

1.4. Wyposażenie stacji APRS w systemie „LoRa”

Stacja *APRScube* konstrukcji DL3DCW (<http://aprscube.de>) jest urządzeniem składającym się z gotowych modułów opartych na modułach *m2stack*: podstawowego z mikrokomputerem ESP32, złączem WLAN, kieszenią dla pamięci microSD, 16 MB pamięci programu i dwucalowym wyświetlaczem, modułu radiowego Lora o mocy wyjściowej 60 mW (18 dBm), odbiornika GPS, modułu QRO – wzmacniacza mocy 1 W (30 dBm), bateryjnego, prototypowego i innych. Minimalny zestaw modułów dla APRS składa się z modułu podstawowego, modułu LoRa i modułu GPS.

APRScube jest przewidziana do pracy w warunkach domowych albo przewoźnie w samochodzie. Możliwe jest podłączenie kombinowanego czujnika temperatury, wilgotności i ciśnienia atmosferycznego (ENV II). Jako uzupełnienie dostępny jest statyw z czujnikiem temperatury i wilgotności powietrza. Oba czujniki mają przypisany ten sam adres i dlatego nie mogą być używane równolegle. Do zasilania napięciem 5 V służy gniazdko USB typu C. Nowszy moduł ENV III zawiera czujniki SHT30 i QMP6988.

Tabela 1.4.1

Parametry czujników zawartych w modułach ENV II i ENV III (dane w nawiasach)

Czujnik SHT30		Czujnik BMP280 (QMP6988)	
Temperatura	-40 – 120°C	Ciśnienie	300 – 1100 hPa
Uchyb temperatury	± 0,2°K (dla 0 – 60°C)	Uchyb ciśnienia	±1 hPa (±3,9 hPa)
Wilgotność wzgl.	10 – 90%	Złącze I2C, wtyk PH2.0 czterokontaktowy	
Uchyb wilgotności	2%	Wymiary 24 x 32 x 8 mm (24,2 x 32,2 x 8,1 mm)	

Stacja może pracować jako urządzenie meldujące współrzędne odczytane z odbiornika GPS (ang. *tracker*) i jako bramka internetowa przekazująca do serwera odebrane dane – *iGate*. Przy pracy w domu dane pozycyjne mogą być przekazywane bezpośrednio do serwera APRS-IS przez sieć lokalną WLAN, a przy pracy ruchomej poza domem nadawane radiowo w standardzie *LoRa*. Uruchomienie, konfiguracja i obsługa stacji *APRScube* są nieskomplikowane.

W witrynie <http://aprscube.de> dostępne są aktualne wersje oprogramowania, program ładujący ESP32 *Flash_Download_Tool* i sterownik *CP210X*.

Zmiana konfiguracji stacji w pliku *APRScube.ini* nie wymaga ponownego ładowania całości oprogramowania z PC. Przy użyciu programu *WinSCP* możliwy jest dostęp FTP do *APRScube*, pobranie pliku i ponowne wpisanie go po dokonaniu modyfikacji.



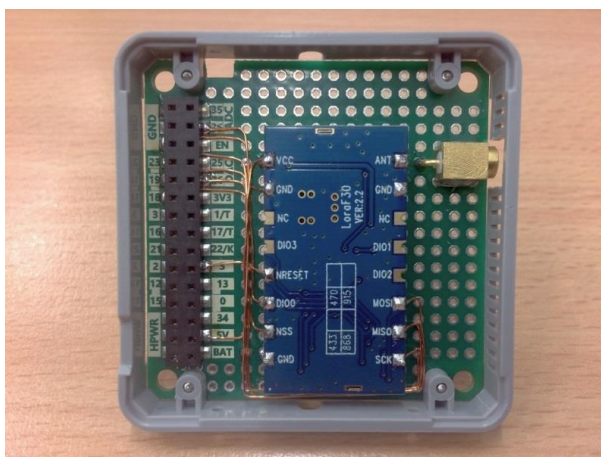
Fot. 1.4.1. APRScube



Fot. 1.4.2. Moduł GPS



Fot. 1.4.3. Kombinowany czujnik ENV II



Fot 1.4.4. Moduł wzmacniacza mocy 1 W

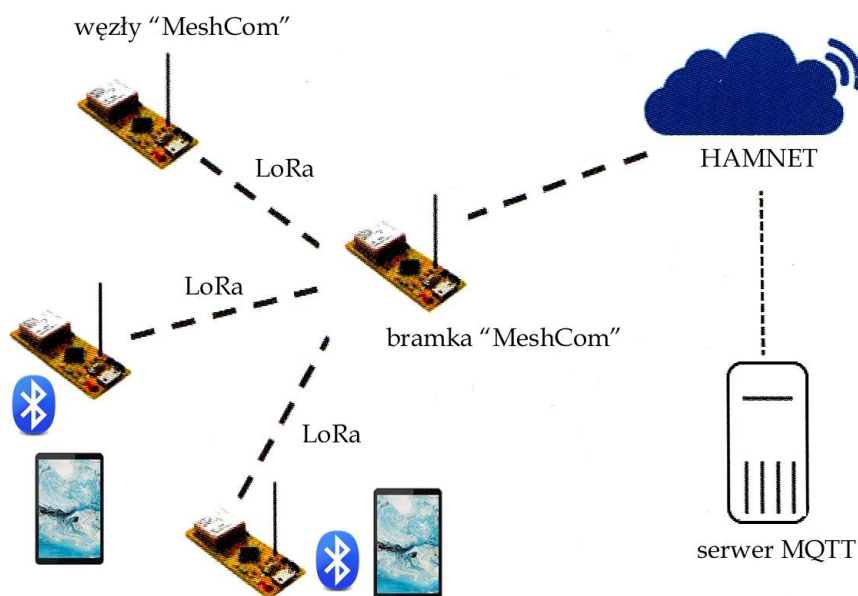
1.5. Sieć systemu „LoRa MeshCom”

Krótkofalarska sieć „LoRa MeshCom” jest opracowywanym przez krótkofalowców austriackich rozwiązaniem pozwalającym na przesyłanie meldunków tekstowych, danych telemetrycznych, pozycyjnych APRS i rozkazów zdalnego sterowania na znaczne odległości mimo małych mocy nadawania. Jej zaletą jest także niskie zapotrzebowanie na energię, dzięki czemu stacje sieci mogą być zasilane bateryjnie. Sieć może okazać się przydatna również w łącznościach kryzysowych i ratunkowych. Sieć *Meshcom* ma topologię siatki, w której poszczególne węzły pośredniczą w komunikacji między dowolnymi innymi węzłami i bramkami łączącymi ją z serwerami MQTT poprzez Hamnet. Różni się więc ona od sieci LoRaWAN mającej topologię gwiazdową, której centralnymi punktami są koncentratory (bramki) i serwery internetowe. W odróżnieniu od sieci LoRaWAN – TNN, Helium, ChirpStack itp. – pracujących głównie w paśmie 868 MHz sieć *MeshCom* korzysta z pasma 433 MHz na zasadach obowiązujących krótkofalowców.

Zasięgi modułów *LoRa* mogą w zależności od zakresu częstotliwości i stosowanych anten przekraczać 10-20 km w terenach wiejskich i 1-5 km w terenach miejskich. W szczególnie korzystnych warunkach

sygnały nadawane z mocą 300 mW były odbierane nawet w odległościach 80 – 100 km. Moc wyjściowa modułów nie przekracza jednak przeważnie 100 mW, a ze względu na ich nagrzewanie zalecane jest korzystanie z niższych mocy. Zastosowane liniowe rozpraszanie widma sygnału (ang. *chirp*) znacząco zwiększa odporność na zakłócenia i osiągnięte zasięgi [1.5.3].

Krótkofalowcy korzystają z modułów na częstotliwość 433 MHz (EU433). W rozwiązaniu austriackim wykorzystano popularne i niedrogie moduły *LoRa* typów *Lilygo-TTGO-T-Beam* (z odbiornikiem GPS, fot. 1.5.1), *Lilygo TTGO Lora* (bez GPS), firm Heltec, DollaTec, Wisblock i podobne. Jako oprogramowanie wewnętrzne modułów wykorzystano oprogramowanie firmy *Meshtastic* [1.5.5], ale z konfiguracją parametrów i częstotliwości znormalizowaną dla sieci krótkofalarskiej. Jest to więc zasadniczo własna wersja oprogramowania (obecnie *MeshCom 1.2.65* dla mikroprocesorów ESP32 dostępna w witrynie OEVSV) gwarantująca kompatybilność z resztą sieci. Obecnie trwają prace nad przygotowaniem wersji *MeshCom 4.0*, która nie będzie się opierać na oprogramowaniu *Meshtastica*.



Rys. 1.5.1. Zasada organizacji sieci. Każdy z węzłów może pośredniczyć w komunikacji między pozostałymi elementami sieci

Stacje użytkowników – węzły (ang. *node*) – lub ich grupy (lokalne chmury radiowe) są połączone z innymi za pośrednictwem Hamnetu. Hamnet staje się więc siecią szkieletową dla stacji indywidualnych wyposażoną w bramki wejściowe LoRa (rys. 1.5.1). W razie potrzeby węzły użytkowników mogą pośredniczyć w przekazywaniu komunikatów do bramki i w przeciwną stronę. Sieć wokół bramki ma więc topologię mniej lub bardziej pełnej siatki (ang. *mesh*). Jej przydatność i niezawodność rosną w miarę wzrostu liczby czynnych stacji. Bramki różnią się od węzłów jedynie tym, że dysponują połączeniem z Internetem lub z Hamnetem i utrzymują w ten sposób połączenie z serwerem MQTT. Jest on w ten sposób połączony ze wszystkimi lokalnymi chmurami tworzącymi dzięki niemu wspólną większą. Zasadniczo pracuje on więc jak znany z innych sieci reflektor. Meldunki pozycyjne są przekazywane do *aprs.fi*.

W Austrii na trasach prowadzących do bramek hamnetowych używana jest częstotliwość 433,175 MHz (w Niemczech m.in. 439,700 MHz). Szerokość pasma sygnału wynosi obecnie 250 kHz (poprzednio było to 125 kHz), stopa korekcji 4/6, a współczynnik rozpraszania 11 (w Niemczech również 10). Szyfrowanie treści (PSK) jest oczywiście wyłączone.

Moduły wyposażone w odbiornik GPS pozwalają na przesyłanie bieżących danych pozycyjnych, natomiast w modułach nie wyposażonych w odbiorniki można wprowadzić stałą pozycję w konfiguracji. Dostępny bezpłatnie (w *Githubie*) program *Meshtastic* (obecnie w wersji 1.2.67) dla Androida pozwala na połączenie telefonu lub komputera androidowego z modułem przez złącze *Bluetooth* i zobrazowanie odebranych pozycji na mapie albo na wyświetlanie i nadawanie komunikatów tekstowych.

W sieci *MeshCom* stosowane są dwa mechanizmy zabezpieczające przed nieskończoną retransmisją komunikatów w pętłach. Jednym z nich jest analiza numerycznych identyfikatorów komunikatu. Komunikaty o znanym identyfikatorze (a więc już wcześniej retransmitowane przez dany węzeł) nie są ponownie przekazywane dalej. Drugim z nich jest ograniczenie liczby skoków czyli liczby retransmisji między węzłami podobnie jak dla sieci APRS. W oprogramowaniu *Meshtastica* ich maksymalna liczba jest ograniczona do siedmiu, ale standardowo krótkofalowcy korzystają z ograniczenia do pięciu, z tym, że każdy z nadawców może wybrać dowolną wartość w podanym zakresie.

Najpopularniejsze obecnie moduły TTGO są wyposażone albo w procesory ESP32 albo nRF52. Do ładowania oprogramowania wewnętrznego [1.5.9] można skorzystać z programu *ESPTool* dla Windows albo z programu *esptool* napisanego w *Pythonie*. Po załadowaniu programu następuje jego wywołanie w oknie terminalowym. Jednym z takich programów jest *GitBash* dla Windows [1.5.6]. W wywołaniu podawane są parametry konfiguracyjne:

```
meshtastic --set-owner OE1XXX --set-region EU433 --ch-index 0 --ch-psk none
```

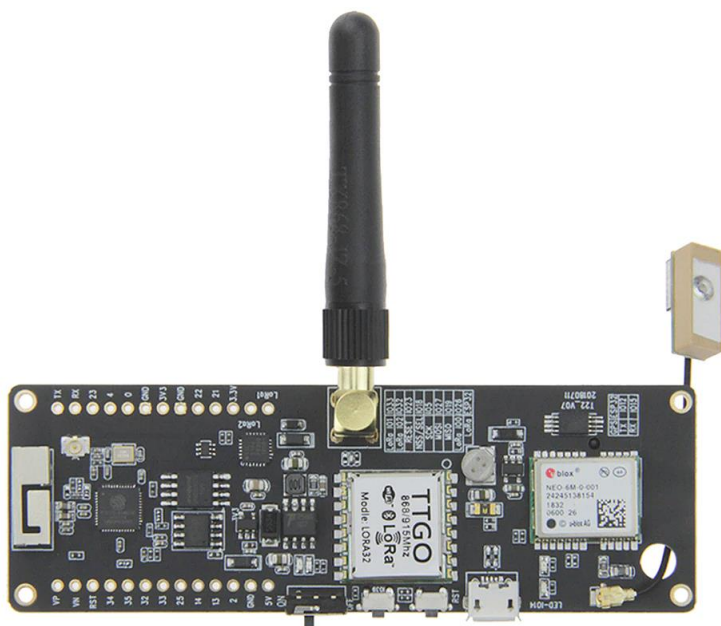
W wersji krótkofalarskiej parametr „region EU433” zawiera całą konfigurację częstotliwości i parametrów transmisji. Zamiast programu terminalowego *GitBash* można użyć *PowerShella* dla Windows lub każdego innego o podobnej funkcjonalności.

Parametry mogą być podawane oddzielnie w kolejnych poleceniach *meshtastic*.... Dla modułów nie posiadających odbiornika GPS można w ramach argumentów wprowadzić też współrzędne geograficzne i wysokość n.p.m., np.: `--setlat 44.33 --setlon 16.03 --setalt 200`. Nadawany stały tekst podaje się za pomocą argumentu `--settext '...'`, dla tekstów zmiennych wygodniej jest korzystać z programu androidalowego. Program dla iPhone'a znajduje się dopiero w fazie próbnej i nie jest powszechnie dostępny.

Polecenie *meshtastic --info* służy do wywołania informacji o parametrach konfiguracyjnych stacji i jest bardzo przydatne w diagnostyce w przypadku wystąpienia problemów.

Bramki wejściowe Hamnetu wymagają dodatkowo podania w ramach argumentów danych dostępowych dla WiFi (identyfikatora, hasła) i adresu IP serwera MQTT (internetowego albo hamnetowego). Są to odpowiednio polecenia *meshtastic --set wifi_ap_mode false*; *meshtastic --set wifi_ssid 'nazwa sieci'*; *meshtastic set wifi_password 'hasło dostępu'*; *meshtastic --set mqtt_server 'adres ip serwera w hamnecie'*; *meshtastic --ch-index 0 --ch-set uplink_enabled true*; *meshtastic --ch-index 0 --ch-set downlink_enabled true*.

Bramki korzystają z tego samego oprogramowania, co węzły i różnią się tylko podaniem tych dodatkowych parametrów.



Fot. 1.5.2. Moduł z odbiornikiem GPS



Fot. 1.5.3. Moduł *DollaTek Lora 433* bez odbiornika GPS

Serwer MQTT (*Message Queing Telemetry Transport*) służy do rozprowadzania komunikatów w sieciach TC/IP przy użyciu protokołu MQTT (opracowanego w 1999 roku na potrzeby łączności satelitarnej i od 2013 roku stosowanego w sieci *Internetu Przedmiotów* – IoT). Pośredniczy on więc w komunikacji między bramkami *LoRa* w Hamniecie. Dodatkowo zapewnia także połączenia z systemami APRS, DAPNET, Messengera w Hamniecie, austriackiego serwera Tetra (służącego do wymiany komunikatów w tej sieci), itd. dla komunikatów zawierających dane odpowiedniego rodzaju. Rozwiązanie austriackie korzysta z własnego oprogramowania serwera *ÖVSV-MeshCom-Server* zamiast powszechnie stosowanego oprogramowania *Mosquitto*. Pozwala to krótkofalowcom na pełniejszy wpływ na jego funkcjonalność i konfigurację niż w przypadku rozwiązania komercyjnego.

Tabela 1.5.1

Wybrane moduły radiowe LoRa 433 MHz

Producent	Typ	Wyposażenie
Lilygo	TTGO T-Beam (V1.1)	Łącze LoRa SX127x, WiFi, Bluetooth, ekran OLED 0,96 cala I2C, odbiornik GPS typu NEO-6M, gniazdko antenowe SMA, pojemnik na akumulator 18650
Lilygo	TTGO Lora (V2.0, V2.1)	Łącze LoRa SX127x, WiFi, Bluetooth, ekran OLED 0,96 cala, procesor ESP32, brak GPS, kieszeń microSD, gniazdko antenowe μ .FL lub SMA, gniazdko mikroUSB (zas. 5 V), gniazdko zasilania z akumulatora litowego 2,7 – 3,6 V
Heltec Automation	WiFi LoRa 32 (V2)	Łącze LoRa SX1276/1278, WiFi, Bluetooth, ekran OLED 0,96 cala 128 x 64 pkt., procesor ESP32, gniazdko mikro-USB (zas. 5 V), kabel do zasilania (bat. 3,7 V), brak GPS, gniazdko antenowe μ .FL
Lilygo	TTGO T-Echo	Łącze LoRa SX1262, Bluetooth, ekran rInk 1,54 cala, gniazdko antenowe μ .FL, procesor nRF5284, odbiornik GPS typu L76K

1.5.1. Przykładowe zastosowania sieci

Proponowane są następujące możliwości wykorzystania sieci:

- komunikacja między dowolnymi dwoma stacjami w sieci (lokalnej chmurze radiowej) – w stylu *WhatsApp* przez program androidowy,
- komunikacja z grupą,
- nadawanie tekstów z komputerów androidowych (wiadomości do 228 znaków),
- nadawanie informacji pozycyjnych,
- połączenie z funkcją SMS sieci DMR IPSC2,
- komunikacja z modułami telegraficznymi Morserino,
- łączności satelitarne przez transponder QO-100 WB w kanale 2409,25 MHz,
- zdalnie połączenie dostępnych publicznie przycisków alarmowych (policja, pogotowie, straż), normalnie w sieciach 868 MHz, ale w sytuacjach alarmowych przełączane do sieci amatorskiej i amatorskiego serwera MQTT,
- połączenia ze stacjami pracującymi emisją JS8Call itp.,
- transmisja różnych danych telemetrycznych, meteorologicznych, informacji o burzach i wyładowaniach elektrycznych,
- ostrzeżenia o niebezpiecznych zjawiskach (podtopienia, silne wiatry, silne opady, burze itd.),
- zdalne sterowanie przemiennikami amatorskimi lub ich włączanie i wyłączanie,
- zawiadamianie i przywoływanie jak w amatorskiej sieci DAPNET (POCSAG),
- lokalny dostęp do Hamnetu (wolna transmisja danych),
- i wiele innych.

Oczywiście są to tylko propozycje i zakres zastosowań można dowolnie uzupełniać. Każdy z użytkowników sieci może dowolnie wybierać interesujące go możliwości i aspekty.

1.5.2. Instalacja i uruchomienie oprogramowania

1. Instalacja *Pythona* dla Windows (w aktualnych wersjach PIP jest instalowany automatycznie z *Pythonem*, wobec tego punkt 3 staje się zbędny):

Z adresu <https://meshtastic.discourse.group/t/tutorial-setting-up-a-windows-machine-for-use-with-meshtastic-python/2872>

Należy koniecznie zaznaczyć pole „Add Python ... to PATH”.

2. Instalacja *GitBash* dla Windows (nie jest ona niezbędna, można skorzystać z *PowerShella* dla Windows, który jest standardowo zainstalowany z systemem lub dowolnego innego interpretatora *Shella*)

Z adresu <https://gitforwindows.org/>.

3. Pobranie *PIP* dla Windows (obecnie jest on instalowany automatycznie z *Pythonem*, patrz p. 1)

Z adresu <https://phoenixnap.com/kb/install-pip-windows>

4. Uruchomienie *GitBash* lub *PowerShella* i sprawdzenie wersji *Pythona* przez podanie w jego oknie polecenia

```
py --version
```

5. Aktualizacja *PIP* i sprawdzenie jego wersji przez podanie w oknie *shella* poniższych poleceń

```
py -m pip install --upgrade pip
```

```
pip --version
```

6. Instalacja i aktualizacja programu narzędziowego *ESPTOOL* i oprogramowania *Meshtastic* komunikującego się z modulem *LoRy* przez złącze COM

```
pip install --upgrade esptool
```

```
pip install --upgrade pytap2
```

```
pip install --upgrade meshtastic==1.2.95
```

7. Sprawdzenie instalacji

```
meshtastic -h
```

Meldunek „*bash ... command not found*“ oznacza, że *Python* nie został dopisany do zmiennej systemowej *PATH* (patrz punkt 1).

W przypadku meldunku błędu należy sprawdzić czy w zmiennej znajdują się wpisy

```
C:\Users\xxxxxx\AppData\Local\Programs\Python\Python310\Scripts\
```

```
C:\Users\xxxxxx\AppData\Local\Programs\Python\Python310
```

Gdzie *xxxxxx* oznacza nazwę użytkownika pod Windows.

Zmienna środowiskowa *DOMAIN* musi zawierać wpisy

```
C:\Users\xxxxxx.DDDDD\AppData\Roaming\Python\Python310
```

```
C:\Users\xxxxxx.DDDDD\AppData\Roaming\Python\Python310\Scripts\
```

Gdzie *xxxxxx* oznacza nazwę użytkownika jak wyżej, a *DDDDDD* – domenę.

8. Instalacja programu zapisującego w pamięci procesora *ESPHome-Flasher* służącego do wpisania oprogramowania do modułu *LoRa*. Konieczne jest podanie numeru złącza COM używanego przez ten moduł i pliku oprogramowania modułu. Numer jest widoczny w *Menadźerze Urządzeń* po podłączeniu modułu do złącza USB komputera. Jeżeli po podłączeniu modułu *LoRy* w *menadźerze* nie pojawia się dodatkowe złącze COM konieczne jest zainstalowanie sterownika *CP210x* lub *CP910x* – zależnie od modelu.

Z adresu <https://github.com/esphome/esphome-flasher/releases/tag/1.4.0>.

Plik *ESPHome-Flasher-1.4.0-Windows-x64.exe*.

9. Skopiowanie oprogramowania do procesora

Połączyć moduł *LoRy* z komputerem za pomocą kabla USB,

Wywołać *ESPHome-Flasher-1.4.0-Windows-x64.exe*

Wybrać w *menadźerze urządzeń* odpowiadające złącze COM,

Wybrać plik z oprogramowaniem *Meshtastic* właściwym dla modelu mikrokomputera. Wybranie niepasującej wersji programu spowoduje, że część jego funkcji nie będzie działać prawidłowo lub w ogóle będzie niedostępna.

Załadować i powtórnie wystartować procesor.

10. Konfiguracja przy użyciu następujących (opisanych poprzednio) poleceń podawanych w oknie *PowerShella* lub *GitBasha*

```
meshtastic --set-owner SPIXXX (zalecane jest podanie znaku z rozszerzeniem jak dla APRS)
```

```
meshtastic --set region EU433
```

```
meshtastic --ch-index 0 --ch-set psk none
meshtastic --set is_always_powered true (jeżeli požądane stałe połączenie z telefonem lub komputerem)
meshtastic --pos-fields POS_BATTERY POS_ALTITUDE POS_ALT_MSL (jeżeli požądane nadawanie informacji o wysokości i stanie baterii (akumulatora); lub za pomocą połączonego polecenia
meshtastic --set-owner SPIXXX --set region EU433 --ch-index 0 --ch-set psk none --pos-fields POS_BATTERY POS_ALTITUDE POS_ALT_MSL --set is_always_powered true
```

Zasadniczo dopuszczalne są tutaj rozszerzenia znaków od -1 do -99, ale w komunikacji z urządzeniami APRS należy ograniczyć się do zakresu -1 do -15 aby zachować kompatybilność. Rozszerzenie -12 jest zarezerwowane dla węzłów połączonych z serwerem MQTT. Do serwera *aprs.fi* są transmitowane dane pozycyjne tylko dla stacji korzystających z rozszerzenia znaku wywoławczego.

Do sprawdzenia prawidłowości konfiguracji służy polecenie *meshtastic -info*. Wśród wyświetlanych danych znajduje się tam numer wersji oprogramowania, „region: EU433” i wiele innych. Definicja EU433 zawiera takie ustawienia jak pasmo 250 kHz i częstotliwość 433,175 MHz. W polu „modemConfig” podana jest szerokość pasma (poniżej w informacji 250), stopa korekcji (tu 4/6) współczynnik rozpraszania (Sf2048 oznacza 11). W stosunku do poprzednich ustawień 125 kHz, 12, 4/8 uzyskuje się znaczny wzrost szybkości transmisji przy około 4-krotnym obniżeniu czułości.

Przykładowa informacja:

```
TTGO T-Beam Info von <tu snak stacji>:
My info: { "myNodeNum": 3213632128, "numBands": 8, "firmwareVersion": "1.2.65",
"rebootCount": 26, "messageTimeoutMsec": 300000, "minAppVersion": 20200,
"maxChannels": 8 }
Preferences: { "positionBroadcastSecs": 375, "sendOwnerInterval": 9, "phoneTimeoutSecs": 900,
"lsSecs": 300, "region": "EU433", "locationShare": "LocEnabled", "gpsUpdateInterval": 375,
"fixedPosition": true }
Channels: PRIMARY psk=unencrypted { "modemConfig": "Bw250Cr46Sf2048", "psk": "AA=="
}
```

Primary channel URL: <https://www.meshtastic.org/d/#CgUYAyIBAA>

11. Konfiguracja bramki wymaga podania dodatkowo poleceń

```
meshtastic --set wifi_ap_mode false
meshtastic --set wifi_ssid 'SSID sieci'
meshtastic --set wifi_password 'hasło dostępu'
meshtastic --set mqtt_server 44.143.8.143 (przy dostępie przez HAMNET; serwer austriacki, w Polsce na pewno będzie to inny adres)
meshtastic --set mqtt_server 89.185.97.38 (przy dostępie przez Internet, uwaga jak wyżej)
meshtastic --ch-index 0 --ch-set uplink_enabled true
meshtastic --ch-index 0 --ch-set downlink_enabled true
meshtastic --ch-index 0 --ch-set psk none
meshtastic --set is_always_powered true
meshtastic --setalt mmm --setlat 99.99999 --setlon 999.9999 (ręczne wprowadzenie pozycji gdy bramka nie ma odbiornika GPS)
meshtastic --pos-fields POS_BATTERY POS_ALTITUDE POS_ALT_MSL
```

12. Instalacja programu androidowego (jeśli użytkownik chce wymieniać w ten sposób wiadomości, jeśli nie to instalacja jest zbędna). Komunikacja komputera z modułem „MeshCom” odbywa się przez złącze *Bluetooth*. Koniecznie musi być to kompatybilna z „MeshComem” wersja 1.2.67.

Pobranie i instalacja na komputerze albo telefonie androidowym z adresu <https://github.com/meshtastic/Meshtastic-Android/releases/tag/1.2.67>

Jeśli uprzednio była zainstalowana starsza wersja programu należy ją najpierw usunąć. Przed skorzystaniem z programu konieczne jest bluetoothowe sparowanie komputera androidowego z modułem *LoRy*. W obecnych wersjach oprogramowania kodem dostępowym BT jest 000000.

Punkty 1 – 8 są konieczne tylko za pierwszym razem i mają na celu zainstalowanie oprogramowania *meshtastic* na PC. Instalacja oprogramowania dla dalszych modułów *LoRy*, ich konfiguracja i sprawdzenie poprawności konfiguracji odbywają się w sposób podany w punktach 9 – 11.

```

Windows PowerShell
Copyright (C) Microsoft Corporation. Alle Rechte vorbehalten.

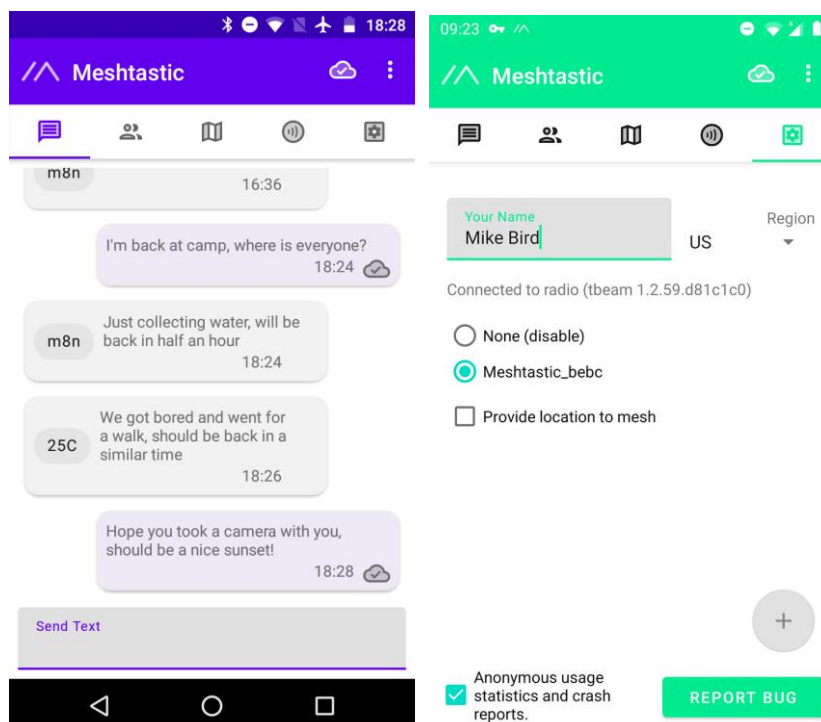
Lernen Sie das neue plattformübergreifende PowerShell kennen - https://aka.ms/pscore6

PS C:\Users\Krzysztof> meshtastic -h
usage: meshtastic [-h] [--configure CONFIGURE] [--export-config] [--port PORT] [--host HOST] [--seriallog SERIALLOG]
  [--info] [--get-canned-message] [--nodes] [--qr] [--get GET] [--set SET SET] [--seturl SETURL]
  [--ch-index CH_INDEX] [--ch-add CH_ADD] [--ch-del] [--ch-enable] [--ch-disable]
  [--ch-set CH_SET CH_SET] [--ch-longslow] [--ch-longfast] [--ch-mediumslow] [--ch-mediumfast]
  [--ch-shortslow] [--ch-shortfast] [--set-owner SET_OWNER] [--set-canned-message SET_CANNED_MESSAGE]
  [--set-owner-short SET_OWNER_SHORT] [--set-team SET_TEAM] [--set-ham SET_HAM] [--dest DEST]
  [--sendtext SENDTEXT] [--sendping] [--reboot] [--shutdown] [--reply] [--gpio-wrb GPIO_WRB GPIO_WRB]
  [--gpio-rd GPIO_RD] [--gpio-watch GPIO_WATCH] [--no-time] [--setalt SETALT] [--setlat SETLAT]
  [--setlon SETLON] [--pos-fields [POS_FIELDS ...]] [--debug] [--test] [--ble BLE] [--noproto]
  [--setchan DEPRECATED DEPRECATED] [--set-router] [--unset-router] [--version] [--support]

options:
-h, --help                show this help message and exit
--configure CONFIGURE    Specify a path to a yaml(.yaml) file containing the desired settings for the connected device.
--export-config           Export the configuration in yaml(.yaml) format.
--port PORT              The port the Meshtastic device is connected to, i.e. /dev/ttyUSB0. If unspecified, we'll try
                        to find it.
--host HOST              The hostname/ipaddr of the device to connect to (over TCP)
--seriallog SERIALLOG   Log device serial output to either 'stdout', 'none' or a filename to append to.
--info                  Read and display the radio config information
--get-canned-message     Show the canned message plugin message
--nodes                 Print Node List in a pretty formatted table
--qr                    Display the QR code that corresponds to the current channel
--get GET               Get a preferences field. Use an invalid field such as '0' to get a list of all fields. Can use
                        either snake_case or camelCase format. (ex: 'ls_secs' or 'lsSecs')
--set SET SET           Set a preferences field. Can use either snake_case or camelCase format. (ex: 'ls_secs' or
                        'lsSecs')
--seturl SETURL        Set a channel URL
--ch-index CH_INDEX    Set the specified channel index. Channels start at 0 (0 is the PRIMARY channel).
--ch-add CH_ADD        Add a secondary channel, you must specify a channel name
--ch-del               Delete the ch-index channel
--ch-enable            Enable the specified channel
--ch-disable           Disable the specified channel
--ch-set CH_SET CH_SET Set a channel parameter. To see channel settings available: '--ch-set all all --ch-index 0'.
                        Can set the 'psk' using this command. To disable encryption on primary channel: '--ch-set psk
                        none --ch-index 0'. To set encryption with a new random key on second channel: '--ch-set psk
                        random --ch-index 1'. To set encryption back to the default: '--ch-set default --ch-index 0'.
                        To set encryption with your own key: '--ch-set psk
                        0x1a1a1a1a2b2b2b2b1a1a1a1a2b2b2b2b1a1a1a1a2b2b2b2b1a1a1a1a2b2b2b2b --ch-index 0'.
--ch-longslow          Change to the long-range and slow channel
--ch-longfast          Change to the long-range and fast channel

```

Rys. 1.5.2.1. Okno PowerShella z poleceniem „meshtastic -h”



Rys. 1.5.2.2. Okna androidowego programu Meshtastic: komunikacji i połączeń BT

W górnej części okien widoczny jest pasek menu służący do wywołania kolejno okien komunikacji, sąsiednich odbieranych stacji, mapy z wyświetlaną pozycją stacji (jeżeli jej położenie jest odczytywane z odbiornika GPS albo zostało wprowadzone w konfiguracji), wyboru kanału (w łącznościach krótkofalarskich obecnie używany jest kanał 0) i konfiguracji. Za pierwszym razem w oknie łączności bluetoothowych należy nacisnąć przycisk z plusem i po znalezieniu węzła połączyć się z nim podając hasło dostępu 000000.

1.5.3. Wymiana meldunków z innymi systemami

1. Meldunki (komunikaty) klientów APRS do węzłów *MeshCom*

Dowolny klient APRS może przysyłać meldunki do węzłów „MeshCom”, ale pod warunkiem, że adresat jest w tym czasie czynny w sieci MeshCom. Długość wiadomości jest ograniczona do 180 znaków alfanumerycznych. Reszta jest obcinana. Znak adresata musi zawierać rozszerzenie, a więc dopuszczalny jest znak OE1KDA-1, ale nie OE1KDA. Przesyłanie danych telemetrycznych będzie możliwe dopiero w wersji *MeshCom 4.0*.

2. Meldunki (komunikaty) od węzłów *MeshCom* do klientów APRS mają format

:<adres docelowy>:<tekst>

Przykładowo

:OE1KDA-7:Pozdrowienia

Należy zwrócić uwagę na obowiązkowe dwukropki.

3. Meldunki (komunikaty) do APRS2SOTA wymagają zarejestrowania znaku nadawcy u G0LGS (<https://www.sotaspots.co.uk/>).

Zarejestrowani nadawcy mogą wówczas nadawać komunikaty adresowane do APRS2SOTA mające następujący format:

:APRS2SOTA:<numer rej. szczytu> <częstotliwość> <emisja> [znak] [uwagi]

Np.

:APRS2SOTA:OE/OO-303 145.500 FM OE5JFE/P Spot via Meshcom

4. Poczta elektroniczna do Internetu. Wiadomości mają następujący format:

:EMAIL:<adres odbiorcy> <tekst>

Należy zwrócić uwagę na znak odstępu między adresem odbiorcy i tekstem wiadomości.

1.6. Odbiór satelitów „LoRa”

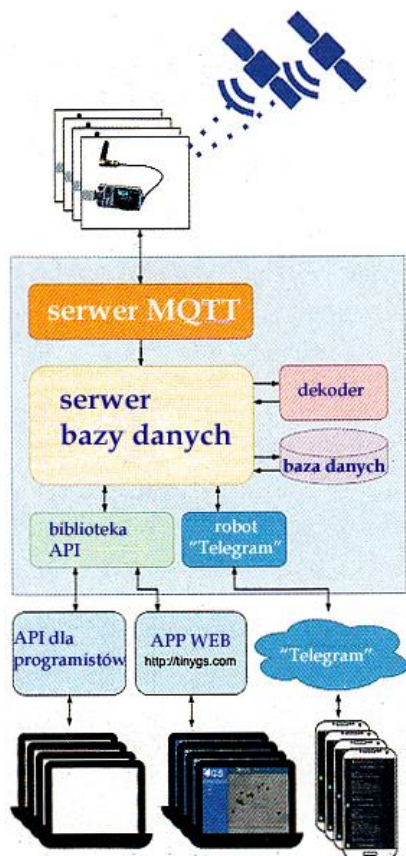
Rys. 1.6.1. Witryna *tinygs.com*: informacje o jednym z satelitów

Witryna [1.6.1] prezentuje światową sieć stacji naziemnych *TinyGS* odbierających satelity nadające dane telemetryczne w systemie LoRa, sondy meteorologiczne i nadajniki zainstalowane na pokładach innych obiektów latających. Sieć ta zapewnia ciągły odbiór danych z satelitów niezależnie od ich położenia na orbicie i nie narażając organizatorów eksperymentów na koszt utrzymywania profesjonalnych stacji naziemnych.

Tanie i nieskomplikowane wyposażenie stacji opiera się na modułach ESP32 LoRa (433 MHz), takich jak Heltec WiFi LoRa32 V1 i V2, TTGO Lora32 V1 i V2, T-BEAM z wyświetlaczami OLED 0,96 cala i inne ESP32 z wyświetlaczami wyposażone w nadawczo-odbiorcze moduły SX126X lub SX127X, oraz w łączce WLAN umożliwiające połączenie internetowe z siecią *TinyGS*.

Oprogramowanie stacji odbiorczej i program ładujący z PC do mikroprocesora przez złącze USB są dostępne w witrynie [1.6.2]. W celu skonfigurowania go należy za pomocą przeglądarki internetowej połączyć się z serwerem HTTP modułu pod adresem 192.168.4.1. Oprogramowanie stacji można załadować także korzystając ze środowiska *PlatformIO*. Po pierwszym załadowaniu i skonfigurowaniu programu możliwe jest włączenie funkcji automatycznej aktualizacji („AutoUpdate”). Możliwa jest też instalacja przy wykorzystaniu środowiska programistycznego „Arduino IDE”. Po zainstalowaniu i skonfigurowaniu programu konieczne jest zameldowanie stacji w sieci.

Załączone do modułów krótkie pionowe antenki są oczywiście niewystarczające. Konieczne jest użycie zewnętrznej anteny dipolowej, pionowej w rodzaju GP [1.6.3], kilkuelementowej anteny Yagi, anteny dwuzwojowej albo dowolnej innej anteny satelitarnej na pasmo decymetrowe. Najlepiej gdy znajduje się ona na dachu, ale dobre rezultaty (dzięki odbiciom od ścian budynków) dają też anteny balkonowe. Odebrane dane telemetryczne są przekazywane internetowo do serwera *MQTT*. Są one dostępne w Internecie pod adresem [1.6.1].



Rys. 1.6.2. Struktura sieci *TinyGS* (źródło <https://tinysgs.com>)

Fot. 1.63. TTGO w obudowie

Obecnie możliwy jest m.in. odbiór satelitów FEES, FEES2, FossaSat 2E1 – 2E6, Gossamer, Norbi i SATLLA-2A/B. Częstotliwości pracy różnią się dla poszczególnych satelitów, ale m.in. są to 400,45 MHz, 401,7 MHz, 436,703 MHz, 436,9 MHz i 437,6 MHz. Niektóre z satelitów pracują w paśmie 915 MHz. Możliwe jest przekazywanie do sieci danych pochodzących z dowolnych satelitów LoRa i od innych obiektów latających. Dane te mogą być nadawane zresztą nie tylko w systemie *LoRa* ale

i dowolnymi innymi rodzajami modulacji odbieralnymi przez moduły *LoRa*, a więc FSK, GFSK, MSK, GMSK i OOK.

Oprogramowane jest wyposażone w tryb testowy. W celu sprawdzenia pracy odbiornika konieczne jest użycie drugiego modułu *LoRa* jako źródła sygnału. Nadawane przez to źródło próbne pakiety danych nie są przekazywane do sieci.

Architekturę sieci przedstawiono na rysunku 1.6.2. Dane z satelitów są odbierane przez dużą liczbę stacji naziemnych i następnie przekazywane przez WLAN do internetowego serwera MQTT (ang. *message queing telemetry transport*). Protokół ten jest używany w transmisji danych telemetrycznych między różnymi urządzeniami także w przypadku sieci o ograniczonych możliwościach i większych opóźnieniach w transmisji. Serwer MQTT otrzymuje oprócz danych telemetrycznych informacje o satelitach, takie jak ich nazwa, częstotliwość pracy i rodzaj modulacji. Jest on połączony z głównym serwerem sieci, na którym następuje dekodowanie i zapis (archiwizacja) danych.

Dane odebrane przez własną stację i przez wszystkie inne zameldowane w sieci są udostępniane w witrynie *TinyGS*. Użytkownicy mogą selekcjonować dane według stacji odbiorczych, satelitów i wybierać pakiety danych. Dane te mogą pochodzić również od innych obiektów np. balonów i zawierać informacje o ich lokalizacji.

Użytkownicy mogą także wywoływać informacje o aktualnie odbieralnych satelitach i o stacjach odbiorczych wraz z ich współrzędnymi geograficznymi, statystykami odbioru itp. Oprócz tego można wyświetlać położenie odbieralnych satelitów na tle mapy.

1.7. Próbnik połączeń „LoRa”

Skonstruowane z myślą o zastosowaniach krótkofalarskich urządzenie, opisane w „Funkamateurze” 3/2023, jest oparte na module *ESP32 LoRa V2* firmy „Heltec” i służy do sprawdzania działania i jakości połączeń radiowych w systemie „LoRa” w paśmie 433 MHz. Może ono przydać się konstruktorom urządzeń łącznościowych i programistom opracowującym ich oprogramowanie albo do badań zasięgu sieci.

Przewidziano także możliwość wykorzystania układu do nauki telegrafii. Konieczne jest podłączenie do kontaktu GPIO25 wtórnika emiterowego, na dowolnym tranzystorze npn, sterującego słuchawki o oporności $2 \times 16 \Omega$ (rys.1.7.3). Generowany jest ton o częstotliwości 650 Hz.

Próbnik FTD może być sterowany przez łącza *Bluetooth* (BT) albo USB z komputera lub inteligentnego telefonu. Ustawione parametry i odbierane dane są wyświetlane na ekranie telefonu, komputera i na wyświetlaczu próbnika. Moduł jest wyposażony w wyświetlacz organiczny OLED o przekątnej 0,96 cala i złącze WiFi. Podobnie jak w module RA-02 występuje tutaj gniazdko antenowe $\mu\text{FL/IPEX}$. Oprócz tego jest on wyposażony w gniazdko JST-PH do podłączenia akumulatora, który można ładować przez gniazdko USB (rys. 1.7.1).

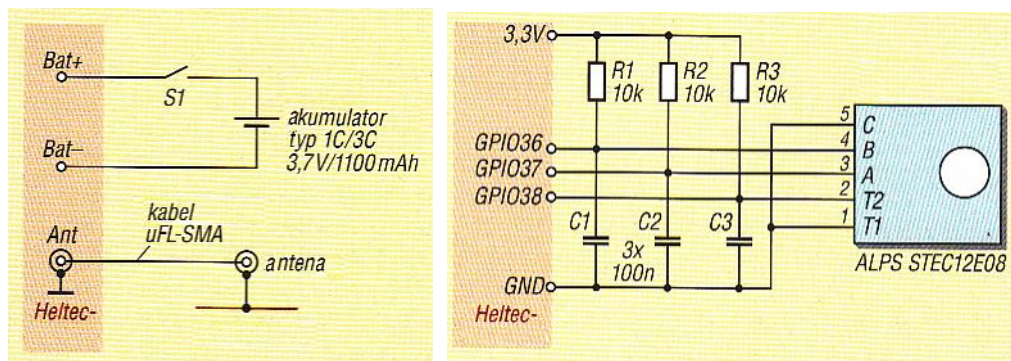
W konstrukcji przewidziano możliwość dołączenia do kontaktów GPIO36–38 obrotowego 24-pozycyjnego kodera obrotowego typu ALPS STEC 12E08 (rys. 1.7.2). Jego wykorzystanie jest przewidziane dopiero w przyszłych wersjach oprogramowania.

Oprogramowanie w postaci archiwum jest dostępne w Internecie w witrynie „Funkamateura” pod adresem https://www.funkamateur.de/tl_files/downloads/hefte/2023/dl2mee_field-test-device.zip.

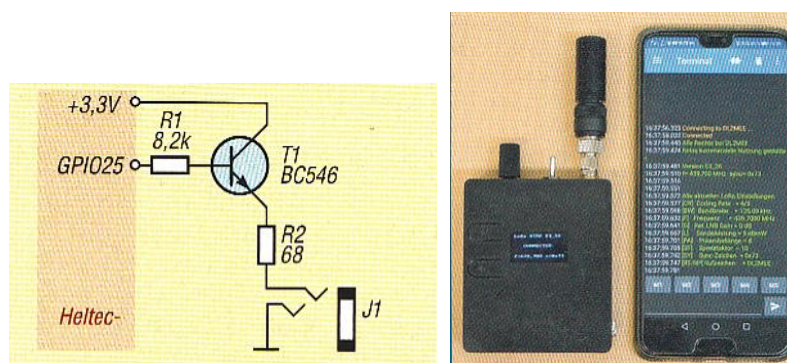
Do załadowania oprogramowania służy program narzędziowy *ESPTool.py* dla PC dostępny w witrynie <https://docs.espressif.com/projects/esptool/en/latest/esp32/esptool>.

Na komputerze ładującym konieczne jest zainstalowanie sterownika CP2102 firmy *Silicon Labs*. W większości przypadków jest on już zainstalowany w związku z wcześniejszymi potrzebami. Po załadowaniu do procesora ESP32 i wywołaniu oprogramowanie rozpakuje się samoczynnie i wykonuje niezbędne kroki instalacyjne. W przypadku wystąpienia błędów w czasie instalacji należy odłączyć zasilanie od modułu, odłączyć go od złącza USB i po upływie 20 sekund po podłączeniu przerwanych połączeń wystartować ESP32 ponownie. Po uruchomieniu oprogramowania warto wprowadzić w konfiguracji własny znak wywoławczy.

Program można obsługiwać przez złącze USB albo beprzewodowo przez złącze BT przy użyciu dowolnego programu terminalowego z komputera albo telefonu androidowego, np. bezpłatnego programu *Serial Bluetooth Terminal App*. Szybkość transmisji wynosi domyślnie 115200 bodów. Korzystanie ze złącza BT wymaga uprzedniego sparowania urządzeń.



Rys. 1.7.1. Podłączenie akumulatora i anteny (przeciwwaga anteny – pasek miedziany na płytce)
Rys. 1.7.2. Podłączenie kodera



Rys. 1.7.3. Podłączenie słuchawek przez wtórnik emiterowy Fot. 1.7.4. Widok ogólny

Użytkownik ma do dyspozycji następujące funkcje: wprowadzenie znaku wywoławczego (#RP *znak*), odbiór i nadawanie pakietów „LoRa” (polecenia #ME dla włączenia monitorowania i #PL dla zmiany parametrów), nadawanie tekstu radiolatarni (#BI odstęp czasu; #BT tekst komunikatu i #BAKE dla włączenia transmisji), funkcję stacji przekaźnikowej (#REP), monitorowania dwóch kanałów (#MO... dla wprowadzenia częstotliwości i nazw kanałów; #MONI dla włączenia), pracę w trybie APRS (#APRS...) i jako trener do nauki telegrafii (#CW...). Podanie poleceń bez parametrów powoduje wyświetlenie pomocy i informacji o niezbędnych parametrach.

Dla włączenia transmisji komunikatów APRS tak aby pozycja była widoczna na mapie w witrynie *aprs.fi* służy następująca (przykładowa) sekwencja poleceń:

#RP OE1KDA	- wprowadzenie znaku wywoławczego
#L 18	- moc nadawania 18 dBm
#APRS DEF	- parametry APRS
#APRS ANZ	- położenie stacji
APRS PUT	- nadanie pakietu.

Dane pozycyjne są zapisane na stałe w programie i należy je dopasować do stanu rzeczywistego.

Do nauki telegrafii służą (przykładowo) następujące polecenia:

#CW A 10	- nadanie 10 skrótów CW
#CW F 2	- nadawanie grup po 5 znaków przez 2 minuty
#CW M 4	- nadawanie grup mieszanych.

[1.1] www.arduino.cc – główna witryna projektu „arduino”

[1.2] sklep.avt.pl – sklep internetowy wydawnictwa AVT

[1.3] www.swiatradio.com.pl – witryna internetowa „Świata Radio”

[1.4] www.kg-gps.de

[1.5] www.dragino.com – witryna producenta nakładek „LoRa” dla „Arduino”

[1.6] aprs.fi – obserwacja położenia stacji APRS i ich komunikatów

[1.1.1] <https://unsigned.io> – kalkulator szybkości transmisji i czułości kanału „LoRa”

- [1.3.1] aprs-map.info – wyświetlanie lokalizacji stacji na mapie
- [1.3.2] https://github.com/lora-aprs/LoRa_APRS_Tracker – oprogramowanie nadajnika pozycji OE5BPA
- [1.3.3] https://github.com/lora-aprs/LoRa_APRS_iGate – oprogramowanie bramki internetowej OE5BPA
- [1.3.4] platformio.org – oprogramowanie *PlatformIO* IDE i *VSCode*
- [1.3.5] <https://apps.magicbug.co.uk/passcode/> – generator hasła dostępu do APRS-IS
- [1.3.5a] <https://aprs.do3sw.de/> – generator hasła dostępu do APRS-IS
- [1.3.6] „Stacja APRScube“, Krzysztof Dąbrowski, OE1KDA, „Świat Radio” 7-8/2022, str. 48
- [1.3.7] „iGate und Tracker bei DB0SL”, Kurt Moraw, DJ0ABR, CQDL 7/2022, str. 45
- [1.5.1] „LoRa MeshCom”, Kurt Baumann, OE1KBC, QSP 1/2022, str. 14
- [1.5.2] „LoRa MeshCom”, Kurt Baumann, OE1KBC, QSP 2/2022, str. 11
- [1.5.3] „Nadajnik APRS w systemie LoRa”, Krzysztof Dąbrowski, OE1KDA, Świat Radio 12/2019 str. 30
- [1.5.4] „Telemetry”, tom 33 z serii „Biblioteka polskiego krótkofalowca”
- [1.5.5] meshtastic.org – oprogramowanie wewnętrzne modułów
- [1.5.6] <https://gitforwindows.org>
- [1.5.7] <https://github.com/LilyGO/TTGO-T-Beam#start-of-content> – krótkofalarskie oprogramowanie dla modułów TTGO/T-Beam z GPS
- [1.5.8] <https://wiki.oevsv.at/wiki/MeshCom> – informacje w języku niemieckim
- [1.5.9] <https://wiki.oevsv.at/wiki/MeshCom/eshCom-Firmware> – oprogramowanie wewnętrzne modułów dopasowane do potrzeb krótkofalarskich
- [1.5.10] <https://de.wikipedia.org/wiki/MQTT> – opis protokołu MQTT
- [1.5.11] oe1kbc@oevsv.at – prowadzący projekt „LoRa MeshCom”, referat projektów narodowych i międzynarodowych OEVSV
- [1.5.12] „Amateurfunk 2023”, „Software für den Funkamateure 2023”, VTH-Verlag
- [1.5.13] „LoRa APRS – eine Einführung”, Karsten Heddenhausen, DC7OS, CQDL 4/2-23, str. 52
- [1.6.1] <https://tinygs.com> – witryna TinyGS
- [1.6.2] <https://github.com/G4lile0/tinyGS>
- [1.6.3] <https://github.com/G4lile0/tinyGS/wiki>
- [1.6.4] „TinyGS – winzige Bodenstationen für LoRA-Satelliten“, Andreas Bilsing, DM4TG, CQDL 4/2023, str. 44

2. Sondy meteorologiczne

Codziennie o ustalonych międzynarodowo godzinach wypuszczane są do atmosfery balony meteorologiczne. Umieszczone na ich pokładach sondy pozwalają na pomiar co najmniej ciśnienia atmosferycznego, temperatury, punktu rosy i wilgotności otoczenia. Dane te są przesyłane na ziemię drogą radiową. Po osiągnięciu wysokości 15 – 30 km balony pękają i transportowane przez nie sondy spadają na spadochronach na ziemię. Ich śledzenie i poszukiwanie stało się nową gałęzią krótkofalarskiego hobby. Balony meteorologiczne startują przeważnie o godzinach 12 i 00 czasu uniwersalnego, ale z niektórych ośrodków są one dodatkowo wysyłane o godzinach 6, 18 UTC albo jeszcze o innych porach. Przeważnie sondy są odbieralne przez około dwie godziny po starcie (maksymalną wysokość osiągają po około 1,5 – 2 godzinach). Początek i koniec okresu odbioru są zależne od odległości stacji odbiorczej od miejsca startu i ewentualnych przeszkód terenowych. Zasięg odbioru rośnie wraz z wysokością osiągniętą przez balon. Drugim rodzajem sond są sondy wyrzucane z samolotów i opadające na ziemię na spadochronikach.



Fot. 2.1. Sondy RS41 i RS92 Fot. 2.2. Odbiornik z modułem LoRa dostosowany do odbioru sond


Oprócz podstawowych pomiarów meteorologicznych czasami prowadzone są pomiary danych interesujących naukowców j.np. natężenia promieniowania radioaktywnego, stężenia ozonu.

Wyniki pomiarów są nadawane radiowo w paśmie 400 – 406 MHz. Moc nadajników jest przeważnie ograniczona do 100 mW dla wysokości poniżej 10 km i do 300 mW dla wysokości do 30 km. Stosowane jest kluczkowanie GFSK, a szerokość kanału w.cz. wynosi 5 – 10 kHz. Do odbioru sygnałów nadawanych przez sondy najprościej jest użyć dowolnego odbiornika amatorskiego (lub radiostacji amatorskiej) odbierającego wąskopasmową modulację FM (NFM – stosowana w pasmach amatorskich), pokrywającego zakres 400 – 406 MHz i połączonego z komputerem. Konieczne jest, aby miał on gniazdo danych – jak to ma miejsce w wielu modelach radiostacji amatorskich – ponieważ pasmo przenoszenia wzmacniaczy m.cz. jest za wąskie do odbioru modulacji cyfrowych.

Do dekodowania danych odebranych z sond firm Vaisala (RS41, RS92KL, RS92SGP, RS92AGP), Meteolabor (SRS400-C34) lub Modem (M2, M10 itd.) służy program *SondeMonitor* dla Windows [2.7]. Ostatnim rozwiązaniem jest system *MySondy* składający się z oprogramowania serwera *MySondy SERVER* i programu śledzącego *MySondy FINDER* dla Androida [2.3].

Możliwe jest także korzystanie z odbiorników programowalnych (SDR) w tym m.in. z *Fun Cube Dongle*. Możliwość odbioru emisji SSB jest przydatna do odbioru słabych sygnałów sond po ich wylądowaniu na ziemi. Stosunkowo niskie moce nadajników sond wymagają stosowania spolaryzowanych pionowo anten zewnętrznych. Mogą być to anteny Yagi 5-elementowe lub dłuższe albo anteny dwuzwojowe [2.4].

Większość balonów meteorologicznych jest obecnie wyposażona w sondy radiowe zawierające odbiorniki GPS (RS41, RS..., M2K2, M10). Dzięki temu możliwe jest dokładne śledzenie ich pozycji i wysokości, a przez to również szybkości i kierunków wiatrów. Bieżąca transmisja współrzędnych umożliwi śledzenie na żywo (w Internecie) trasy lotu i trajektorii opadania sondy. Część sond udaje się dzięki temu odnaleźć i spożytkować. Oprócz danych meteorologicznych, pozycyjnych i naukowych transmitowane są dane techniczne: napięcie zasilania i temperatura w pojemniku sondy. Sondy są umieszczane w pojemnikach styropianowych izolujących układ elektroniczny od niskich temperatur panujących na dużych wysokościach.



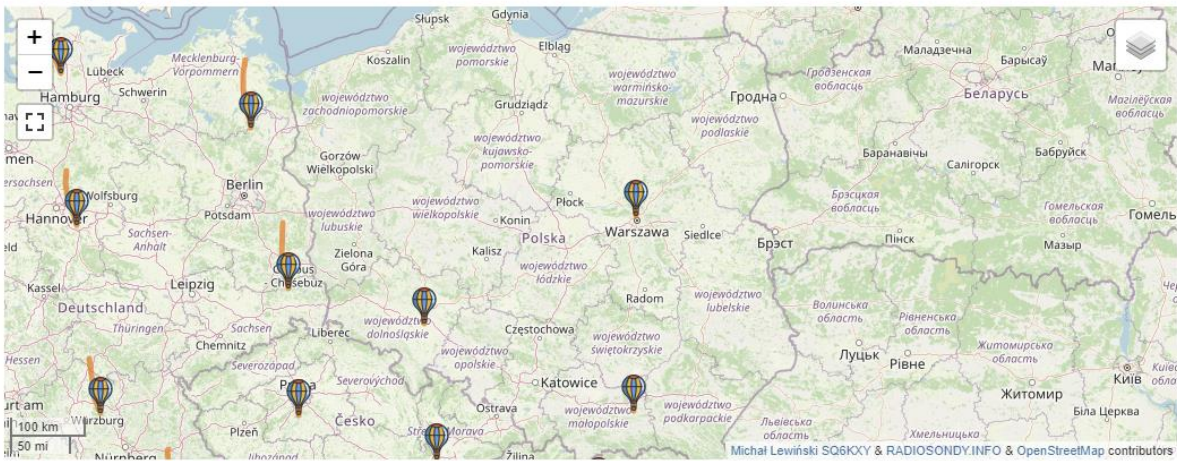
SQ6KXY Radiosonde Tracker

Meteorologische Sonden Datenbank

▶ **Gepante Suche von Sonden**

Gemeldet von	Gültig von [UTC]	Gültig bis [UTC]	Nummer	Start Ort	Beschreibung
--------------	------------------	------------------	--------	-----------	--------------

▶ **Derzeitig aktive Sonden**



Nummer	Typ	Letzter Frame [UTC]	φ	λ	Kurs	Geschwindigkeit	Höhe	Steigung	Start Ort	Frequenz	Entfernung vom QTH
U2510718	RS41	2023-02-04 11:30:29	44.9151	-123.008	6 °	43 km/h	502 m	6.3 m/s		401.40 MHz	0.00 km
T3110930	RS41	2023-02-04 11:30:31	40.2271	18.1511	159 °	33 km/h	892 m	4.9 m/s	Galatina (IT)	403.50 MHz	0.00 km
U2420423	RS41	2023-02-04 11:30:19	50.234	-5.23884	72 °	24 km/h	3656 m	4.0 m/s	Camborne (GB)	403.300 MHz	0.00 km
U4624050	RS41	2023-02-04 11:30:36	47.3455	19.1658	153 °	59 km/h	3823 m	5.1 m/s	Budapest (HU)	403.200 MHz	0.00 km
D22045549	DFM17	2023-02-04 11:30:23	37.1788	-80.3343	119 °	100 km/h	3832 m	4.8 m/s	Blacksburg [VA] (US)	404.010 MHz	0.00 km
S4220165	RS41	2023-02-04 11:30:32	43.3654	-8.46445	310 °	19 km/h	4000 m	3.7 m/s	A Coruña (ES)	404.002 MHz	0.00 km
ME9900022	M20	2023-02-04 11:30:31	43.7094	4.34326	156 °	50 km/h	4123 m	2.9 m/s	Nîmes (FR)	402.002 MHz	0.00 km
U1650309	RS41	2023-02-04 11:30:35	-37.5925	144.925	47 °	63 km/h	4281 m	3.6 m/s	Melbourne (AU)	401.500 MHz	0.00 km
D20068163	DFM09	2023-02-04 11:30:08	12.2119	-69.0053	314 °	20 km/h	4357 m	7.4 m/s	Willemstad (CW)	405.509 MHz	0.00 km
ME9A00726		2023-02-04 11:30:29	54.6196	17.5047	182 °	81 km/h	4387 m	6.0 m/s		MHz	0.00 km
ME27229D0	M10	2023-02-04 11:30:21	37.9639	-1.20146	246 °	31 km/h	4396 m	5.0 m/s	Murcia (ES)	403.402 MHz	0.00 km

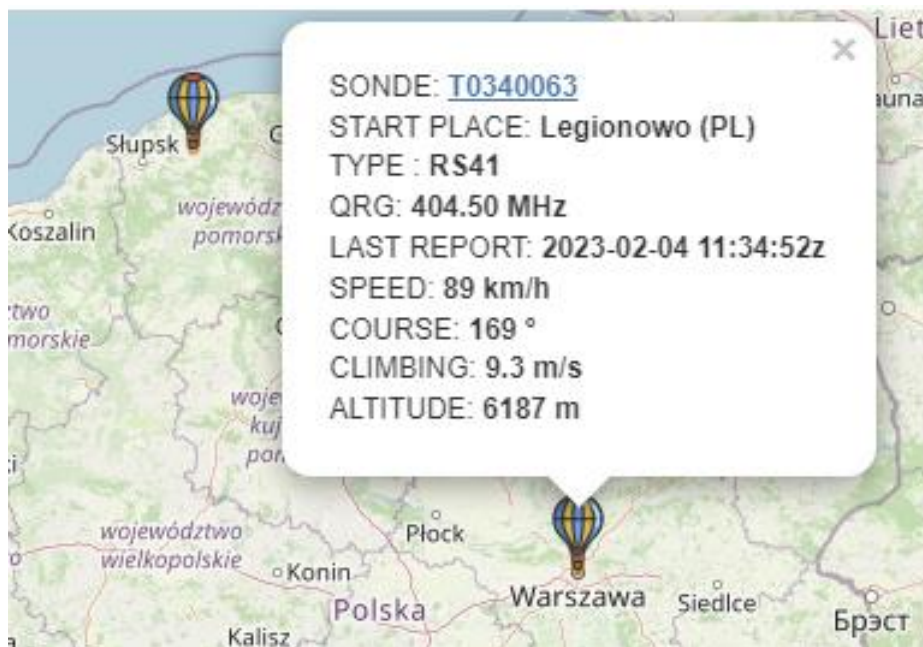
Rys. 2.1. Witryna SQ6KXY. Obserwacja sond z 4 lutego 2023

Michał Lewiński SQ6KXY opracował oprogramowanie *Radiosondy* służące do śledzenia sond i uruchomił witrynę [2.1] umożliwiającą wygodną obserwację przelotów w Internecie. W witrynie podawane są najważniejsze informacje o nich takie jak numer identyfikacyjny, szybkości lotu i wznoszenia się, współrzędne geograficzne, wysokość i częstotliwość nadawania. Dawniej dane te były również przyjmowane i wyświetlane przez *aprs.fi*.

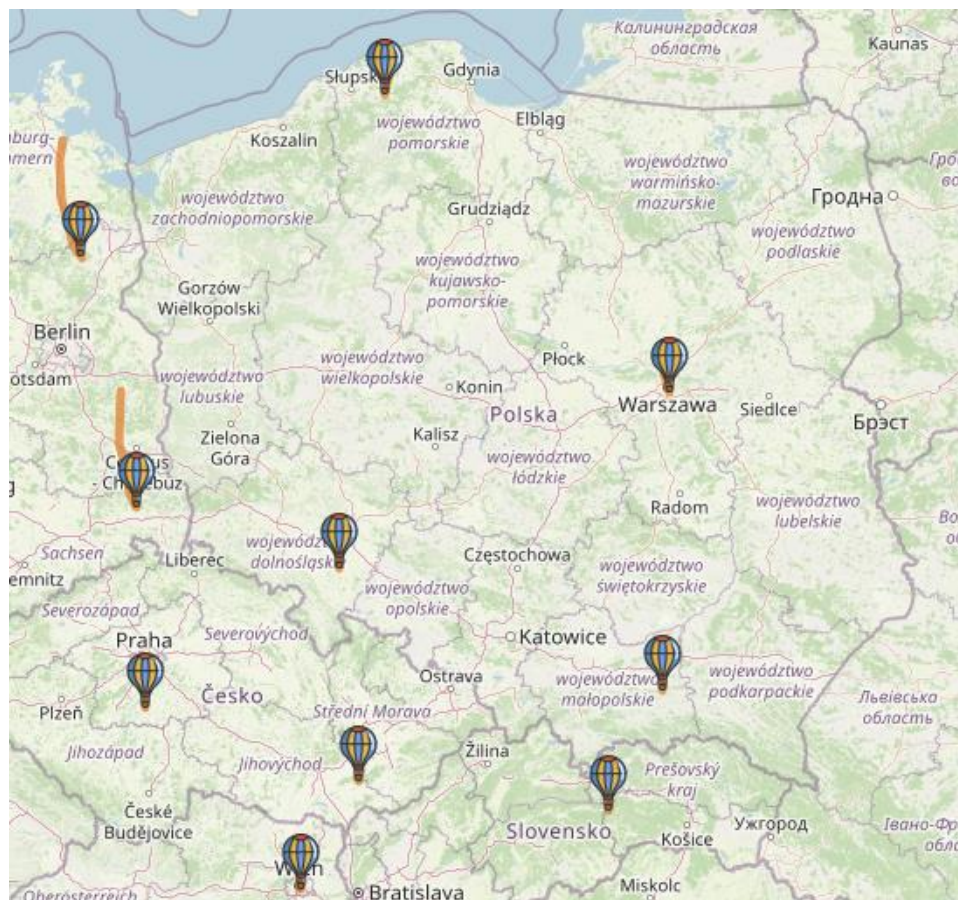
Po zarejestrowaniu się w witrynie SQ6KXY użytkownicy mogą pobrać odwzorowanie pamięci dla *Maliny*. Program służy do odbioru sygnałów sond za pomocą programowalnego odbiornika (ang. SDR) podłączonego do *Maliny*. Możliwe jest równoległe podłączenie trzech odbiorników, z których każdy pokrywa wycinek 2 MHz, co zapewnia pokrycie pełnego zakresu 400 – 406 MHz.

Odebrany z witryny obraz pamięci musi zostać zapisany w module pamięciowym SD za pomocą programu *Win-Disk-Imager* lub podobnego.

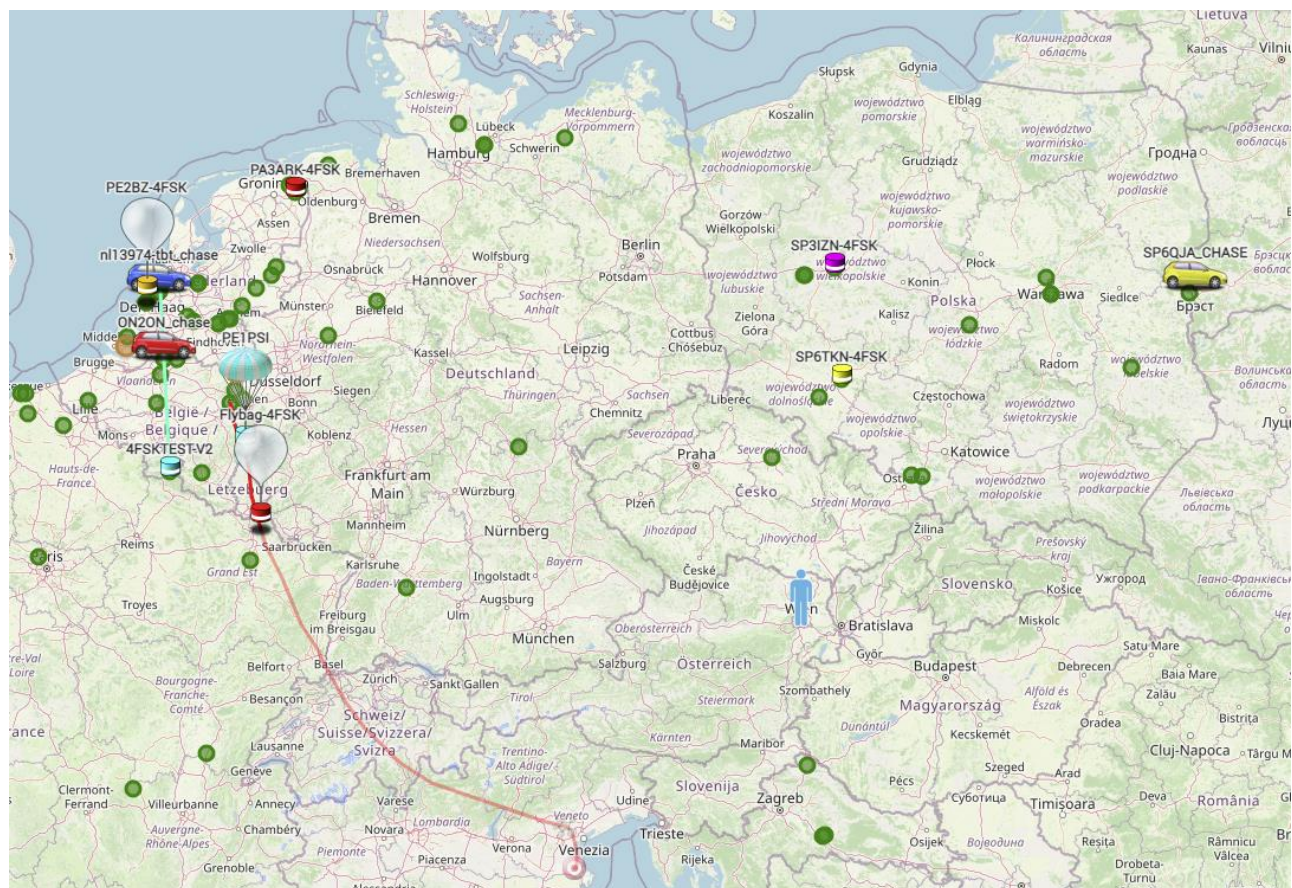
W zależności od ich rodzaju i życzenia właścicieli sondy powinno się do nich zwracać za niewielkim wynagrodzeniem albo wyrzucać zgodnie z lokalnymi zasadami segregacji odpadów: oddzielnie baterie i oddzielnie część elektroniczną. Mimo spadochronu część sond ulega uszkodzeniu przy lądowaniu, a ponowna kalibracja czujników pomiarowych nie zawsze się właścicielom opłaca.



Rys. 2.2. Po naciśnięciu symbolu balonu wyświetlane są informacje o nim i jego ruchu



Rys. 2.3. Miejsca startów balonów meteorologicznych w Polsce i w najbliższych okolicach

Rys. 2.4. Fragment mapy z witryny *amateur.sondehub.org*

Wśród zainteresowanych poszukiwaniem spadłych sond znajdują się oczywiście krótkofalowcy, którzy m.in. instalują w nich oprogramowanie nadające komunikaty pozycyjne w standardzie APRS (FM) albo komunikaty tekstowe emisjami RTTY (SSB) i CW. Tylko niektóre modele radiosond (np. *Vaisala RS41 SGP*) nadają się do takiego wykorzystania przez przeprogramowanie. Oprogramowanie radiolatarni dla zawartego w nich mikroprocesora STM32 autorstwa SQ5RWU zostało udoskonalone przez OM3BC [2.8]. Ze względu na niską moc nadajnika sondy – 60 mW – radiolatarnia nadaje się bardziej do zastosowań w lotach balonów krótkofalarskich aniżeli w stacjach naziemnych. Do zaprogramowania mikroprocesora konieczny jest programator ST-LINK V2 oraz program STM32 ST-LINK Utility ze sterownikiem. Pod adresem [2.9] znajduje się oprogramowanie sondy jako nadajnika do „Łowów na lisa” w paśmie 70 cm. Nadawane komunikaty zawierają znak wywoławczy i napięcie zasilania.

Lekkim i energooszczędnym rozwiązaniem jest wykorzystanie jako odbiorników modułów LoRa 433 MHz LILYGO, TTGO, TTGO T-Beam i Heltec TTGO LoRa. Wbudowany w nich układ nadawczo-odbiorczy LoRa SX127x nie jest ograniczony tylko do tej emisji. Pozwala on na pracę emisjami GFSK, FSK, MSK, GMSK iOOK (CW) i może znaleźć szersze zastosowanie w różnych projektach krótkofalarskich. Oprogramowanie pozwalające na odbiór sond typów RS41SG/SGP, M10, M20 i innych oraz instrukcja jego instalacji znajdują się w witrynie [2.10].

Użytkownicy modułów TTGO mają do dyspozycji androidowy program śledzący *MySondy GO* [2.3].

Tabela 2.1

Polskie ośrodki startowe balonów meteorologicznych i niektóre ośrodki w krajach sąsiadujących

Ośrodek	Częstotliwości [MHz]	Godziny startów [UTC]	Typ sondy	Osiągane wysokości [km]
Łeba	403,0 – 404,0	00, 12	M20, RS41SG	32
Wrocław	403,0 – 403,8	00, 12	M20, RS41SG	31
Tarnów	403,2	00, 12, (18)	M20	
Legionowo	404,5	00, 12, (18)	RS41SG	34
Greifswald	402,3 – 404,7	00, 12	RS41SGP	28
Lindenberg	405,1 – 404,5 – 405,8	00, 06, 12, 18	RS41SGP	32
Praga	401,1 – 404,2	00, 06, 12	RS41SG	31
Prostějów	402,1 – 404,1	00, 12	RS41SGP	30
Poprad-Ganowce	400,6 – 402,61	00, 12	RS41SG	32

[2.1] www.radiosondy.info

[2.2] radiosonde.eu – wyczerpujące informacje o balonach, sondach, ich odbiorze i poszukiwaniach. Języki francuski, włoski, hiszpański, częściowo niemiecki.

[2.3] <http://radiosonde.eu/RS06/RS06S01.html> – projekt MyServer

[2.4] <http://radiosonde.eu/RS00-D/RS08D02-D.htm> – anteny

[2.5] <https://www.coaa.co.uk> – SondeMonitor

[2.6] amateur.sondehub.org – śledzenie balonów amatorskich i poszukiwanie sond

[2.7] <https://www.dl1nux.de/umbau-einer-vaisala-rs41-wettersonde/> – instrukcja programowania i konfiguracji sondy

[2.8] www.om3bc.com/docs/rs41/rs41_en.html – oprogramowanie dla sond RS41

[2.9] <https://github.com/darksidelemm/RS41FOX> – oprogramowanie nadajnika do „Łowów na lisa”

[2.10] https://github.com/dl9rdz/rdz_ttgo_sonde – oprogramowanie odbiorcze dla modułów LoRa

3. Przenoszenie rozmów w sieci DMR

W sieci DMR możliwe jest przenoszenie rozmów (ang. *roaming*) z jednego z przemienników znajdujących się na trasie przebywanej przez operatora stacji na następny. Wchodzące w grę przemienniki muszą być zapisane w sporządzonym przez operatora spisie.

W trakcie przemieszczania się radiostacja może samoczynnie poszukiwać lepiej odbieralnych (dających silniejsze sygnały) stacji przemiennikowych i przełączać się na stację dającą najlepsze rezultaty – jest to więc coś w rodzaju inteligentnego przeszukiwania pasma (rys. 3.1). Operator nie musi dzięki temu ręcznie przełączać kanałów i poszukiwać najlepszego połączenia. Radiostacja poszukuje przemienników wśród wpisanych na jej liście „Roam List”. W grę wchodzi jedynie przemienniki połączone z tą samą siecią DMR (BM albo IPSC2) i umożliwiające korzystanie z tej samej grupy rozmówców, tak aby można było podtrzymywać QSO. Muszą one też nadawać komunikaty radiolatarni (jest to zależne od ustawień dokonanych przez ich operatorów). Kryterium oceny jest siła sygnałów (RSSI), która musi też przekraczać podaną w konfiguracji wartość progową. Wartość tą podaje się w programie konfiguracyjnym lub wybiera z rozwijanej listy na ekranie. Do typowych wartości progowych należą -105 – -108 dBm, ale można spróbować dobrać eksperymentalnie wartość korzystniejszą w danej okolicy. Poszukiwanie następnego przemiennika zapewniającego dobrą łączność rozpoczyna się po spadku siły sygnału poniżej zadanego progu.



Rys. 3.1. Przykładowe rozmieszczenie przemienników na trasie ruchu stacji. Ich zasięgi częściowo zachodzą na siebie dzięki czemu możliwe jest prowadzenie łączności bez przerw. Ich częstotliwości pracy i kody CC mogą się różnić, ale wszystkie muszą pozwalać na korzystanie z jednej i tej samej grupy

W przypadku nie znalezienia przemiennika dającego siłę odbioru przekraczającą wartość progową radiostacja kontynuuje poszukiwania, ale łączność jest prowadzone na dotychczas używanym kanale.

Korzystanie z funkcji przenoszenia (przekazywania) rozmów jest możliwe jeżeli w zasięgu odbioru znajdują się co najmniej dwa połączone z siecią przemienniki pozwalające na korzystanie z tej samej grupy rozmówców. W praktyce zasięgi tych przemienników muszą chociaż trochę zachodzić na siebie. Funkcją przenoszenia rozmów dysponują radiostacje Motoroli i Hytery chociaż nie wszystkie. Popularne chińskie modele Baofenga, Tytery, Retevisa i inne tej klasy cenowej niestety nie dają takich możliwości. Włączanie i wyłączanie funkcji wymaga przypisania jednemu z przycisków programowalnych funkcji „Roam” i naciśnięcia go w miarę potrzeby.

W niektórych modelach radiostacji Motoroli możliwe jest też ręczne wywoływanie funkcji „Manual Site-Roaming” za pomocą jednego z przycisków programowalnych. Naciśnięcie przycisku powoduje w tym przypadku przejście radiostacji na następny kanał w spisie i po krótkim zapytaniu („ping”) sprawdzenie siły odbioru.

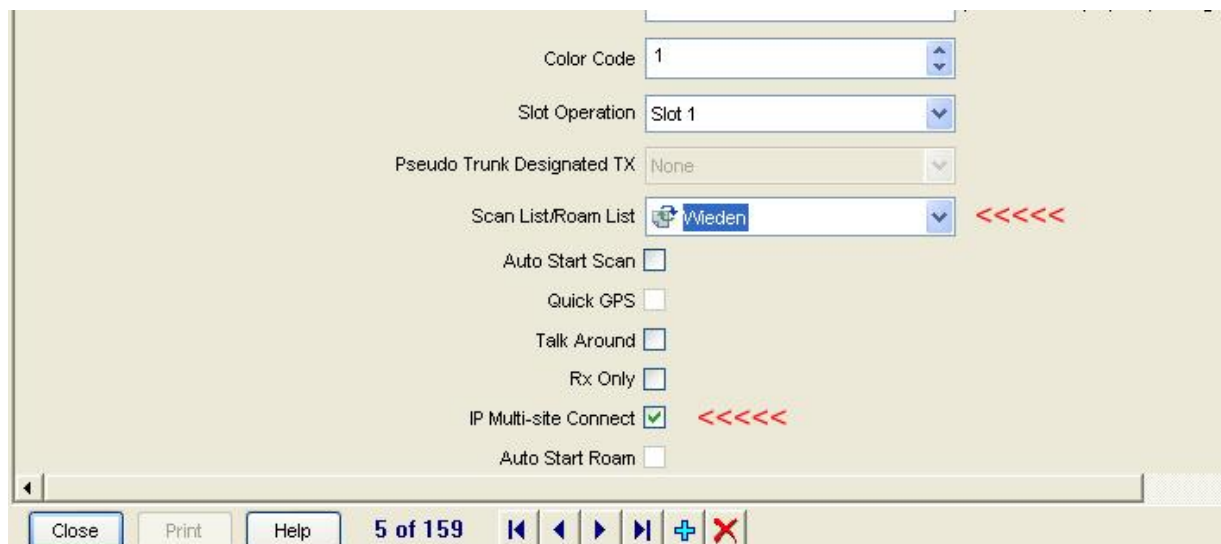


Rys. 3.2. Spisy kanałów dostępnych do wyboru i wybranych dla funkcji przenoszenia rozmów w programie CPS Hytery

Do sporządzenia spisu służy – przykładowo w oprogramowaniu CPS dla radiostacji Hytery PD785 – punkt „Roam”/„Roam List” (rys. 3.2). Po jego wywołaniu na ekranie komputera wyświetlane są dwa pola: po lewej stronie stacji dostępnych do wyboru („Available”), a po prawej – stacji wybranych przez operatora („Members”). Między nimi znajdują się przyciski służące do przesuwania wybranych stacji pomiędzy polami. Pod nimi znajduje się pole granicznej siły odbioru. W zakładkach (definicjach) kanałów DMR mających się pojawić w polu „Available” należy zaznaczyć kratkę „IP Multi-Site Connect” (w radiostacjach „Motoroli „IP Site Connect”) i w polu „Scan List/Roam List” („Motorola: „Scan/Roam List”) podać oznaczenie spisu. W zależności od potrzeb można założyć więcej takich spisów i w każdym z nich wybrać tylko stacje leżące w danej okolicy albo na części przebywanej trasie i używane w tych przypadkach grupy. Dla każdej grupy konieczna jest osobna lista, warto więc zastanowić się z góry, która grupa będzie naprawdę potrzebna. Przykładowo może być to grupa ogólnokrajowa albo któraś z grup regionalnych. Wygodnie jest nadać spisowi nazwę związaną z nim grupy. Dla grup aktywowanych na żądanie sprawa się komplikuje, ponieważ korzystanie z nich na kolejnym wybranym przemienniku wymagałoby transmisji użytkownika skierowanej do tej grupy na nowo wybranym przemienniku. Zasadniczo więc lepiej pozostać przy grupach dostępnych standardowo. Spisy przemienników wymagają oczywiście co pewien czas aktualizacji.

Zaznaczenie na zakładce kanału pola „Return to selected CH” powoduje powrót do pierwotnie używanego kanału po zakończeniu pracy w trybie przenoszenia QSO.

Przenoszenie rozmów funkcjonuje w przypadku korzystania z przemienników fabrycznych Motoroli lub Hytery, ale nie przy pracy przez przemienniki konstrukcji amatorskich lub mikroprzemienniki (ang. *hotspot*).



Rys. 3.3. Konfiguracja kanału. Konieczne jest zaznaczenia pola „IP Multi-Site Connect” i wybranie spisu w polu „Scan List/Roam List”



Rys. 3.4. Programowanie przycisku służącego do włączania funkcji przenoszenia

[3.1] „Amateurfunk Praxis. Digital-QTC“, JochenBernst, DL1YBL, Funkamateurl, 3/2020, str. 284

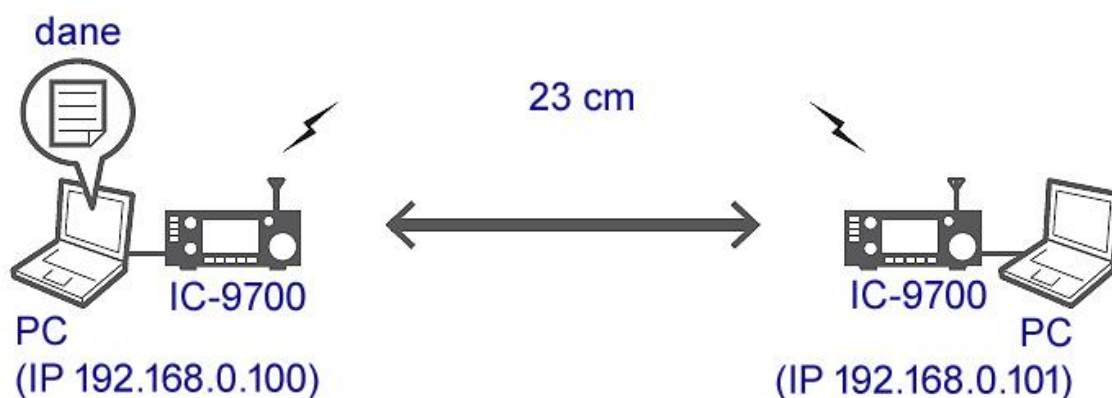
[3.2] <https://afundr.de>

[3.3] <https://n1atp.com>

4. Szybka transmisja danych w trybie DD

W paśmie 23 cm i w wyższych D-Star daje możliwość szybkiej transmisji danych z przepustowością 128 kb/s. Jest to szybkość maksymalna, a rzeczywista może się różnić od niej w zależności od bieżących warunków. Sygnał danych zajmuje pasmo w przybliżeniu 300 kHz. Transmitowane mogą być dowolne dane komputerowe, w tym także zakodowane cyfrowo sygnały głosu albo obrazy. Możliwa jest też wymiana plików komputerowych, zdalny dostęp do komputera korespondenta albo korzystanie z jego serwera http.

Podobnie jak w systemie transmisji głosu w pasmach 2 m i 70 cm do wyboru są łączności przez przemienniki albo bezpośrednio. Oczywiście muszą być to specjalne przemienniki dostosowane do transmisji danych z tą przepustowością – ID-RP1200VD, a nie przemienniki dla sygnałów cyfrowego głosu. Ze względu na to, że obecnie w Polsce nie ma żadnego przemiennika tego rodzaju ograniczymy się do łączności bezpośrednich. Z dostępnych na rynku modeli radiostacji do transmisji danych nadaje się jedynie IC-9700. Oprócz niego możliwością tą dysponuje od dawna już nie produkowany, ale ciągle używany model ID-1. Na transmisję danych w pasmach 23 cm i wyższych ma pozwalać zapowiadana radiostacja IC-905 pokrywająca pasma do 6 cm włącznie, a z dodatkowym modulem w.c.z. także pasmo 3 cm (10 GHz).



Rys. 4.1. Bezpośrednia transmisja danych. Radiostacja jest połączona z komputerem za pomocą kabla ethernetowego. Zalecane jest wyłączenie wszystkich pozostałych połączeń sieciowych komputera dla uniknięcia ewentualnych konfliktów

Dla łączności bezpośrednich w paśmie 23 cm przeznaczone są kanały 1255,000, 1255,500 i 1254,500 MHz. Zarówno łączności bezpośrednie jak i za pośrednictwem przemienników odbywają się w trybie simpleksowym. W łącznościach przemiennikowych w konfiguracji kanału należy wybrać transmisję simpleksową przez przemienniki – RPS, a w łącznościach bezpośrednich – zwykły simpleks.

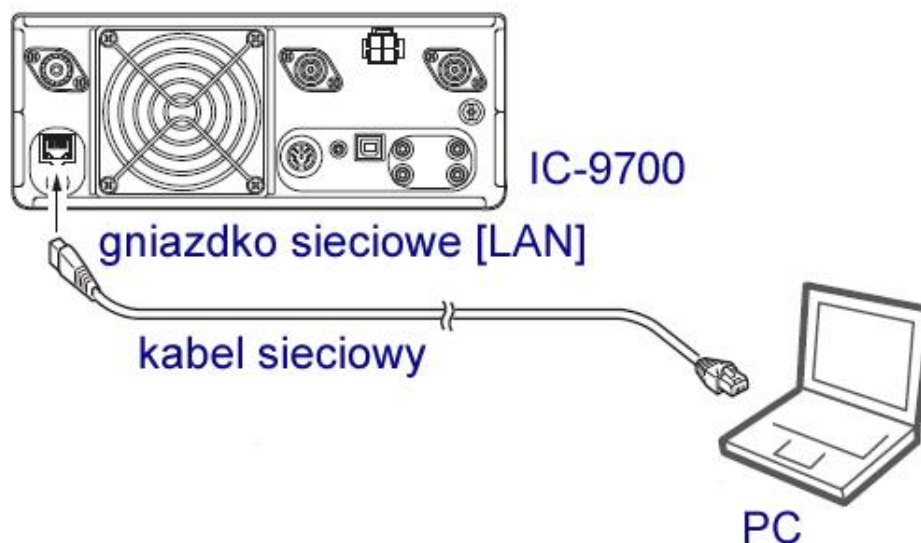
W łącznościach bezpośrednich operatorzy stacji muszą przypisać im stałe i jednoznaczne adresy IP. Zalecane są adresy z niepublicznych serii 192.168.0.100 i kolejnych, 192.168.100.100 i kolejnych albo 10.0.0.x. W początkowym okresie w związku z małą liczbą potencjalnie zainteresowanych stacji uzgodnienia w tej sprawie nie powinny przysparzać problemów. Adresy te są przyznawane komputerom połączonym z radiostacjami i dokomuje się w panelu sterowania w punkcie dotyczącym sieci. W przypadku łączności przemiennikowych adres z krótkofalarskiej serii 44 jest przyznawany przez DHCP przemiennika. Maska sieci jest równa 255.255.255.0.

Trochę myląca może być możliwość przyznania adresów IP radiostacjom, ale dotyczy to zdalnego sterowania przez lokalną sieć albo przez internet, a nie łączności DD.

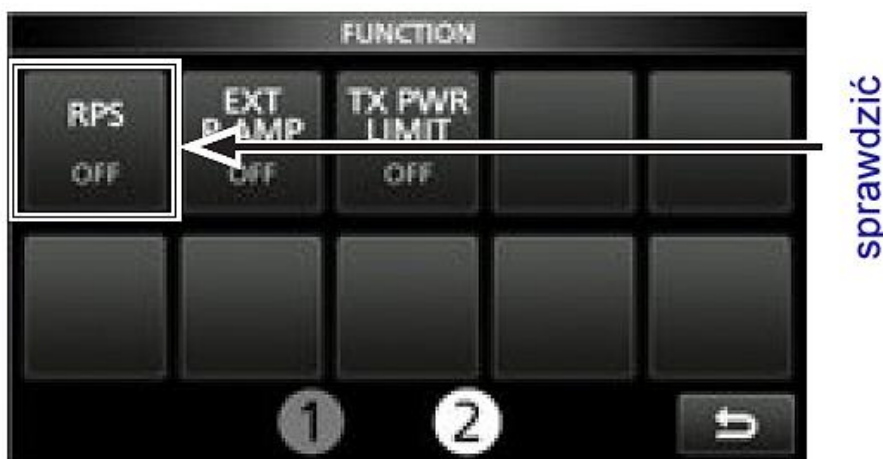
Po połączeniu radiostacji z komputerem rozpoczęcie transmisji wymaga wyłączenia blokady „TX INHIBIT”. Do włączania i wyłączania blokady służy klawisz „TRANSMIT” na przedniej ścianie radiostacji. Transmitowane dane nie są zaszyfrowane i są dostępne dla wszystkich. Należy wziąć to pod uwagę w trakcie łączności.

W punkcie SET > DV/DD Set > **DD Packet Output** należy nastawić transmisję wszystkich rodzajów pakietów („All”).

Konieczne jest podanie własnego znaku wywoławczego, który może różnić się od używanego w transmisji głosu. Znak podawany jest w menu SET > My Station > **My Call Sign (DD)**. Jako dodatek do znaku po ukośnej kresce można przykładowo podać DD, ale nie jest to konieczne.



Rys. 4.2. Połączenie komputera z radiostacją. Kabel sieciowy należy zakupić oddzielnie. Konieczne może być też wyposażenie komputera w złącze LAN jeśli nie posiada on go standardowo



Rys. 4.3. Ustawienie trybu simpleksowego w oknie konfiguracji IC-9700. Naciskanie pola RPS powoduje naprzemienne włączanie i wyłączenie trybu pracy przez przekaźnik simpleksowy

Po naciśnięciu przycisku „Function” i wybraniu drugiej strony na wskaźniku należy sprawdzić i ewentualnie wyłączyć tryb dla przekaźnika simpleksowego (RPS). Oczywiście należy ustawić pożądaną częstotliwość pracy w paśmie 23 cm.

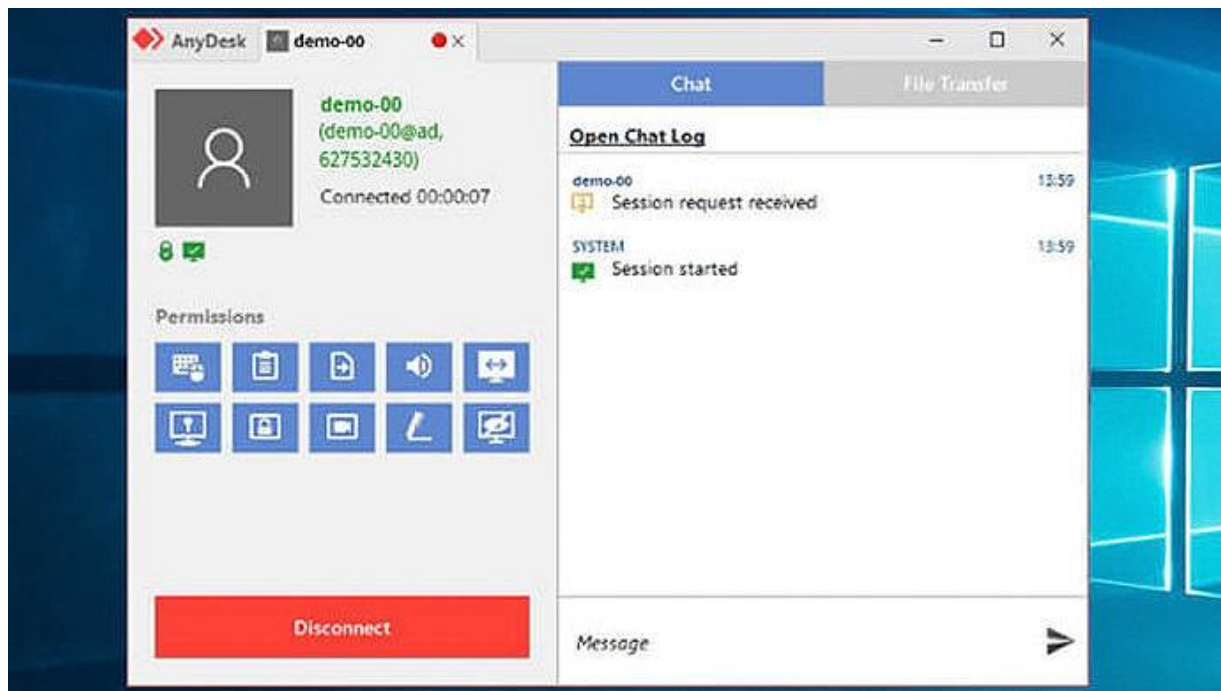
W oknie (2) > **CS** należy znak docelowy CQCQCQ w linii „UR:” zastąpić przez znak wywoławczy korespondenta. Na zakończenie konieczne jest wyłączenie blokady nadawania „TX INHIBIT” przez naciśnięcie klawisza „TRANSMIT”.

Użytkownik jest informowany o przebiegu transmisji za pomocą symbolu monitora ze strzałkami obok niego. Strzałka pomarańczowa oznacza odbiór danych i przekazywanie ich do komputera, a zielona – nadawanie danych do korespondenta.

W radiostacji ID-1 najwygodniej jest dokonać ustawień w programie sterującym ID-1. Krótki kabelek z wtyczką ethernetową, wystający z obudowy wymaga podłączenia przedłużacza.

Po zakończeniu konfiguracji dobrze jest sprawdzić połączenie za pomocą poleceń *ping* podawanego z okna wiersza poleceń.

Do zdalnego dostępu do komputera korespondenta można użyć programu „AnyDesk” w trybie połączenia indywidualnego P2P lub innego podobnego. Program jest dostępny bezpłatnie (dla celów prywatnych) w Internecie pod adresem *anydesk.com*. W oprogramowaniu zapory przeciwłamaniowej (ang. *firewall*) należy pozwolić na dostęp przez „AnyDesk” albo też wyłączyć zaporę całkowicie na czas łączności radiowej. Jest to jeszcze jeden powód dla wyłączenia wszystkich pozostałych sieciowych połączeń komputera ze światem zewnętrznym (LAN, WLAN). *AnyDesk* jest dostępny również dla innych systemów operacyjnych, a nie tylko dla Windows.



Rys. 4.4. Okno programu „AnyDesk”

5. Tłumienie wolnej przestrzeni

W literaturze poświęconej antenom i propagacji fal spotykamy się z terminem „tłumienie wolnej przestrzeni”. Wzór stosowany do jego obliczenia nie obejmuje strat energii w środowisku rozchodzenia się fali, w otoczeniu anten i samych antenach ani strat wywołanych przez odbicia od jonosfery czy powierzchni ziemi. Z jakimi zjawiskami mamy więc tutaj do czynienia i skąd biorą się wynikające z niego straty.

Wielkość zwaną potocznie tłumieniem wolnej przestrzeni, oznaczoną poniżej literą a , oblicza się z dostosowanych do praktycznego użytku, wzorów:

$$a \text{ [dB]} = 32,44 + 20 \log (f \text{ [MHz]}) + 20 \log (r \text{ [km]}) \quad (5.1)$$

lub

$$a \text{ [dB]} = 92,44 + 20 \log (f \text{ [GHz]}) + 20 \log (r \text{ [km]}) \quad (5.1a)$$

wywodzących się z podstawowego wzoru

$$a \text{ [dB]} = 20 \log (4\pi/c) + 20 \log (f \text{ [Hz]}) + 20 \log r \text{ [m]} \quad (5.1b)$$

gdzie f jest częstotliwością fali w podanych w nawiasie kwadratowym jednostkach, r – odległością anteny odbiorczej od nadawczej, a c – szybkością światła. Dla uproszczenia zrezygnowano z wyprowadzenia tego wzoru.

Wzór obowiązuje dla anten izotropowych (teoretycznych anten o kulistej charakterystyce promieniowania). Jak wynika z niego, nie chodzi tutaj o jakiegokolwiek efekty związane z tłumieniem fali w atmosferze, wpływy wilgoci, przewodności ziemi, strat przy odbiciach od jonosfery, strat w obiektach otaczających anteny, o wpływ rozpraszania lub ugięcia fali, nie uwzględnione są też straty w samych antenach i liniach zasilających itd.. Nie występują w nim żadne składowe związane z tego rodzaju zjawiskami i właściwościami fizycznymi jakichkolwiek obiektów. Wzór obowiązuje ogólnie – a więc nie tylko w pobliżu ziemi, ale również w przestrzeni kosmicznej gdzie straty energii w próżni są minimalne. O ile w przypadku strat nie uwzględnionych we wzorze chodzi o rzeczywiste straty energii polegające na zamianie jej w przeważającej części na ciepło, o tyle obliczana z powyższych wzorów wielkość a zależy monotonicznie tylko od częstotliwości i odległości anten nadawczej i odbiorczej.

Przykładowo dla fali o częstotliwości 7 MHz przy odległości 500 km otrzymujemy wartość 53,4 dB dodatkowo do tłumienia wnoszonego przez odbicia od jonosfery, które można orientacyjnie przyjąć na ok. 30 dB na skok, co daje prawie trzykrotną różnicę i to wyrażoną w dB. Dla częstotliwości 433 MHz i odległości 1 km tłumienie wolnej przestrzeni wynosi w przybliżeniu 85,2, a dla 868 MHz – 91,2 dB.

Przy uwzględnieniu wpływu atmosfery, zawartej w niej pary wodnej itd. otrzymujemy przykładowo dla częstotliwości 10 GHz tłumienie około 0,01 dB/km, co przy odległości 10 km daje tłumienie 0,1 dB, natomiast ze wzoru (1) na tłumienie wolnej przestrzeni otrzymuje się w przybliżeniu 128 dB, co jest różnicą gigantyczną. Porównując wyniki dla dowolnych odległości i częstotliwości otrzymujemy zawsze taką istotną różnicę. Wyobrażając sobie antenę w kształcie kuli otaczającą antenę nadawczą można oczekiwać, że odbierze ona prawie całą wypromieniowaną energię po odjęciu jedynie jej rzeczywistych strat na trasie i to niezależnie od częstotliwości i przy stosunkowo słabym wpływie odległości.

Wyjaśnienie zjawiska wymaga uwzględnienia dwóch czynników. Jednym z nich jest zmniejszanie się gęstości energii na jednostkę powierzchni w funkcji odległości (podział energii na zwiększającą się powierzchnię), a drugim wpływ powierzchni skutecznej anteny i jej zależności od częstotliwości.

Jeśli wyobrazimy sobie kwadrat o rogach wyznaczonych przez promienie mające swój początek w źródle promieniowania (rys. 5.1) – w tym przypadku w antenie – zauważamy, że jego boki rosną proporcjonalnie do odległości od źródła, co oznacza, że jego powierzchnia rośnie z kwadratem tej odległości. Zakładając pomijalne rzeczywiste straty energii na trasie oznacza to, że wypromieniowana energia rozdziela się na coraz to większą powierzchnię czyli jej gęstość (na jednostkę powierzchni) maleje z kwadratem odległości wg wzoru:

$$S = P/A = P/4\pi r^2$$

gdzie S jest gęstością energii, P – mocą promieniowaną, a r – odległością.

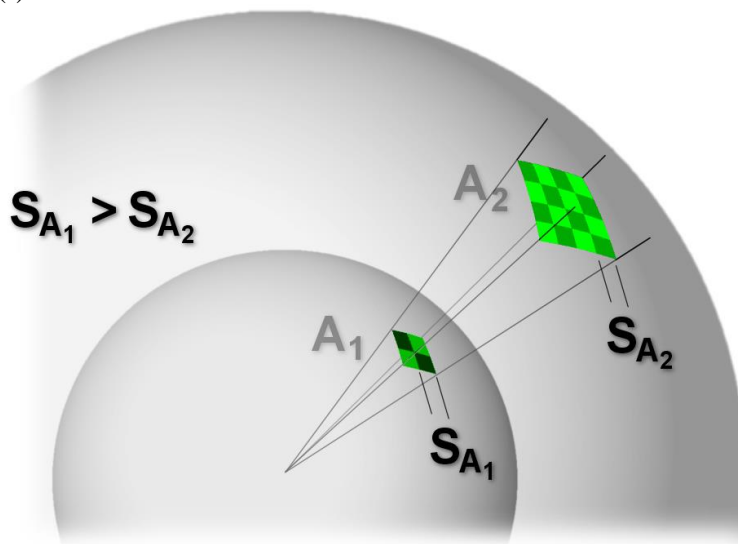
Przy zachowaniu stałych i ograniczonych wymiarów anteny odbiera ona coraz mniejszą część energii w miarę jej oddalania (w myśli) od anteny nadawczej – dla dwukrotnie większej odległości oznacza to spadek natężenia pola o 6 dB. Wyjaśnia to człon $20 \log (s)$ występujący we wzorze (1). Zjawisko to jest niezależne od częstotliwości.

Drugi człon wzoru wiąże się z powierzchnią skuteczną anteny. Oblicza się ją ze wzoru:

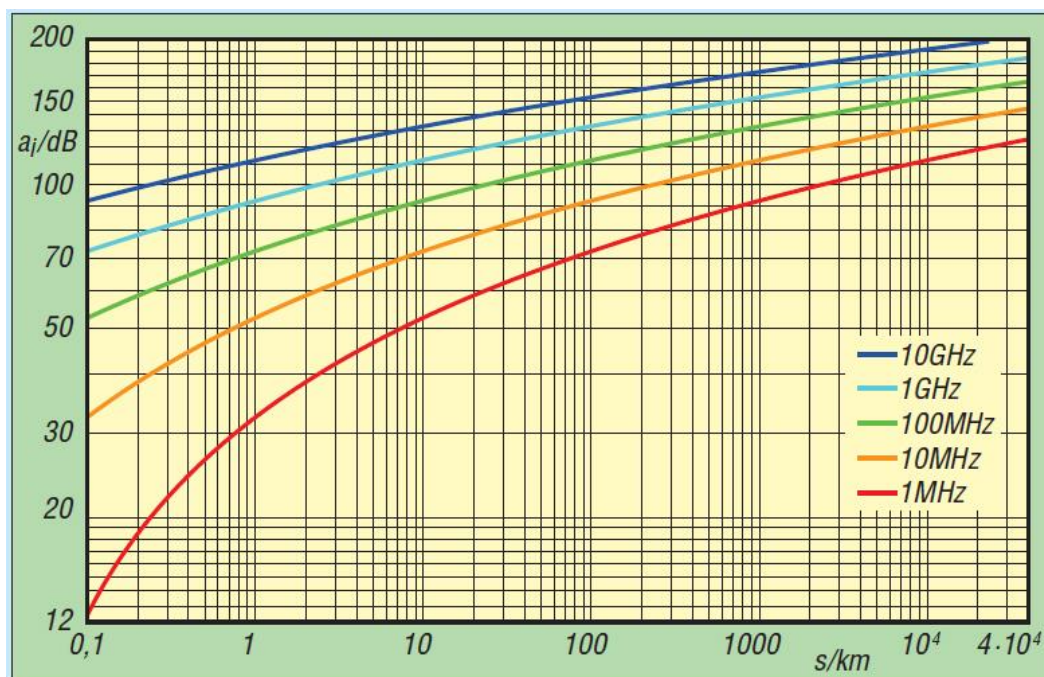
$$A = \frac{\lambda^2}{4\pi} G_i$$

gdzie λ jest długością fali, a G_i – zyskiem w stosunku do anteny izotropowej, dla dipola półfalowego jest to 2,14 dBi (1,637 raza).

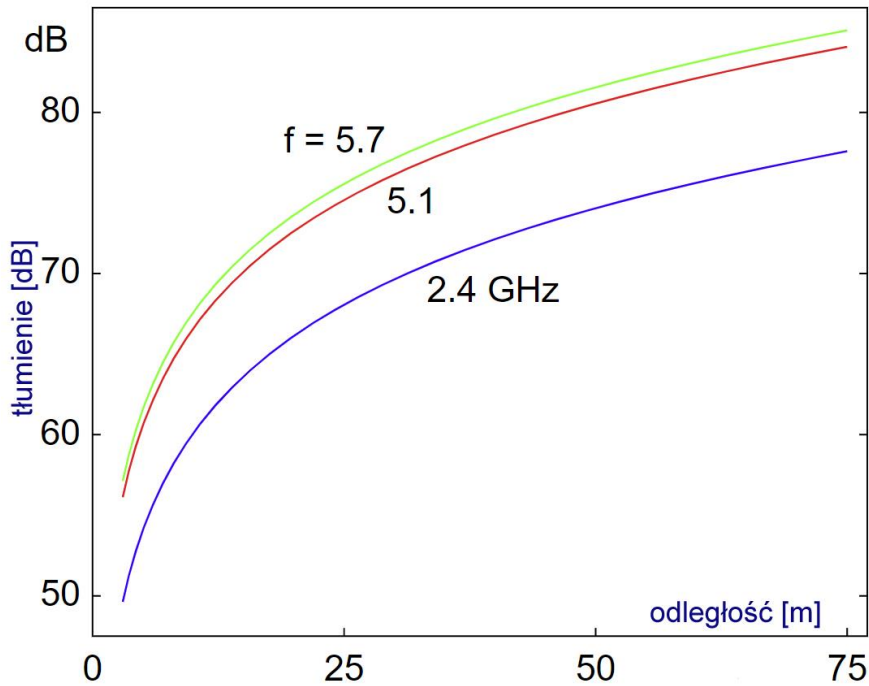
Powierzchnia skuteczna anteny jest dla wielu typów anten zależna od częstotliwości i przykładowo dla dipola półfalowego dla fali 2 m wynosi 0,52 m², a dla fali 20 m – 52 m². Wiąże się to oczywiście ze zmianą długości dipola w zależności od częstotliwości rezonansowej. Zmniejszanie się wymiarów anteny w funkcji częstotliwości powoduje, że odbiera ona coraz mniej energii. Wyjaśnia to pochodzenie człon zależnego od częstotliwości: $20 \log(f_2/f_1)$, a przy założeniu, że $f_1 = 1$ MHz lub GHz członu $20 \log(f_2) = 20 \log(f)$.



Rys. 5.1. Powierzchnia kwadratu A wyznaczona przez promienie rośnie z kwadratem jego odległości od źródła promieniowania. Źródło: [5.4]



Rys. 5.2. Wykres zależności ze wzorów (5.1) i (5.1a) dla różnych częstotliwości i odległości do 40000 km. Dla fal krótkich należy dodać tłumienie w wyniku odbić od jonosfery – po około 30 dB na skok fali. Różnica tłumienia pomiędzy poszczególnymi krzywymi wynosi dla każdej częstotliwości 20 dB



Rys. 5.3. Tłumienie wolnej przestrzeni dla częstotliwości sieci WLAN standardu IEEE 802.11



Rys. 5.4. Przykładowy bilans energetyczny w sieci LoRa

Tabela 5.1
Przykładowe wartości tłumienia wolnej przestrzeni

Częstotliwość	Uwagi	Tłumienie [dB]
27 MHz	300 m (zdalne sterowanie)	51
100 MHz	100 km (radiofonia)	112
13 GHz	30 km (radiolinie)	144
1575 MHz	25000 km (GPS L1)	184
15 GHz	38000 km (satelity geostacjonarne)	208
2,1 GHz	384000 km (ziemia – księżyc)	211
$5,56 \times 10^5$ GHz	$149,6 \times 10^6$ km (zielone światło, λ 540 nm, słońce – ziemia)	371*
4 GHz	$7,4 \times 10^{23}$ km (promieniowanie tła 3°K ze skraju kosmosu – 78 mld lat świetlnych)	582

Uwagi:

* Zysk kierunkowy oka ludzkiego wynosi około 76 dB, dzięki czemu bilans strat trasy spada do około 295 dB.

Dla pełnego skompensowania wpływu częstotliwości antena musiałaby mieć stałą i niezależną od niej powierzchnię skuteczną, co oznacza rosnący w funkcji częstotliwości zysk antenowy. Przykładem anten o takich właściwościach są anteny paraboliczne. Ich powierzchnia skuteczna w szerokim zakresie częstotliwości pozostaje niezmienna, a więc ich zysk rośnie w miarę zwiększania częstotliwości pracy.

Wpływ powierzchni skutecznej anteny jest z kolei niezależny od odległości między antenami nadawczą i odbiorczą.

Podsumowując należy stwierdzić, że zjawisko zwane tłumieniem wolnej przestrzeni oparte jest na zmniejszaniu się gęstości energii na jednostkę powierzchni, co przy niezmiennych i ograniczonych wymiarach anten skutkuje spadkiem napięcia na jej zaciskach i dodatkowo zmniejszaniem się powierzchni skutecznej anten w miarę wzrostu częstotliwości pracy dającym taki sam wynik. Nie jest to więc rzeczywiste tłumienie fali, a jedynie pogorszenie się efektywności wykorzystania promieniowanej energii przez antenę odbiorczą. Efekt ten musi być oczywiście uwzględniony przy obliczaniu bilansu energetycznego trasy połączenia, a przeciwdziała mu się przez zwiększanie zysku antenowego.

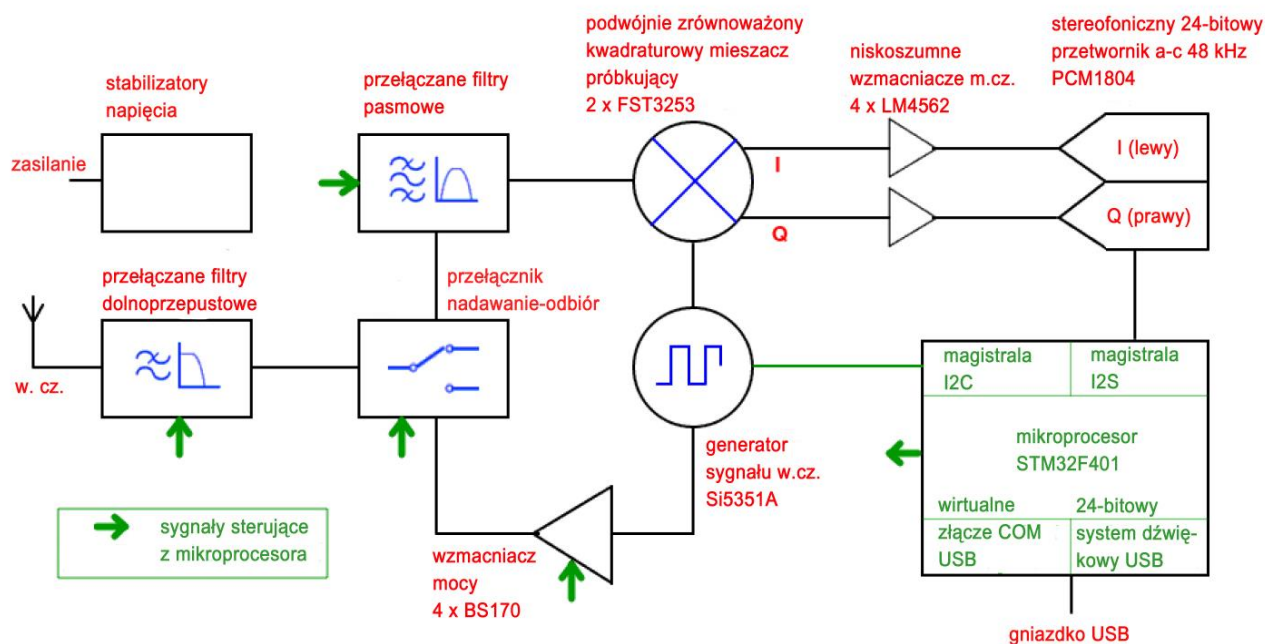
Co do terminologii to może lepiej byłoby zamiast o tłumieniu wolnej przestrzeni mówić o stratach na trasie propagacji lub o stratach w wolnej przestrzeni.

Na wyższych częstotliwościach, zwłaszcza mikrofalowych konieczne jest dokładne uwzględnienie tłumienia atmosferycznego, wpływu pary wodnej, transmisji wielodrożnej (odbić), wpływu ugięcia fali, pokrycia terenu roślinnością itp.

- [5.1] „Dämpfung“ der Funkstrecke“, Thomas Schiller, DC7GB, *Funkamateureur* 1/2012, str. 64 i *Funkamateureur* 2/2012, str. 171.
- [5.2] Dysk CD z rocznikiem 2012 miesięcznika *Funkamateureur*
- [5.3] „Praxis der Mikrowellenantennen“, Sepp Reithofer, *UKW-Berichte* 1987
- [5.4] Wikipedia
- [5.5] „Poradnik ultrakrótkofalowca“, Zdzisław Bieńkowski, WKŁ, Warszawa 1988

6. QDX – radiostacja z cyfrową obróbką sygnałów

Zestawy do własnej konstrukcji radiostacji QRP opierają się prawie zawsze na klasycznym rozwiązaniu nadajnika z filtrową metodą generacji sygnału SSB i odbiornika superheterodynowego. QDX posiada natomiast odbiornik z cyfrową obróbką sygnałów, a w nadajniku zastosowano uproszczoną cyfrową metodę generacji sygnału wyjściowego zamiast bardziej skomplikowanego nadajnika SSB. Radiostacja (zależnie od wersji) jest przeznaczona do pracy emisjami cyfrowymi w pasmach 80 – 20 m lub 20 – 10 m. Oprócz tego dostępna jest wersja jednopasmowa QDX-M dla pasma 10, 11 lub 12 m – do wyboru. W fabrycznym sprzęcie nadawczo-odbiorczym technika cyfrowej obróbki sygnałów już od dłuższego czasu stała się standardem.



Rys. 6.1. Schemat blokowy

Schemat blokowy radiostacji przedstawiono na rys. 6.1. Tor odbiorczy QDX zawiera kwadraturowy mieszacz próbkujący pracujący na scalonych przełącznikach CMOS typu FST3253 (lub odpowiednikach) i dostarczający sygnałów m.cz. synfazowego i kwadraturowego (I/Q) [6.1]. Sygnały heterodyny przesunięte w fazie o 90 stopni względem siebie pochodzą z tego samego układu syntezy co sygnał nadawany – układu scalonego Si5351A. Sygnały m.cz. po wzmocnieniu przez niskoszumne wzmacniacze operacyjne LM4562 i przetworzeniu na postać cyfrową w 24-bitowym przetworniku analogowo-cyfrowym (A-C) typu PCM1804 (pracującym z częstotliwością próbkowania 48 kHz) są podawane na 32-bitowy mikroprocesor STM32F401. 24-bitowa rozdzielczość zapewnia dynamikę przekraczającą 110 dB. W standardowym trybie pracy do PC przekazywany jest jeden wspólny strumień danych, ale do celów eksperymentalnych możliwe jest udostępnianie równoległe strumieni I i Q.

W torze nadawczym zastosowano uproszczoną metodę generacji sygnału w.cz. pozwalającą na nadawanie jedynie sygnałów o pojedynczej składowej modulującej. W odróżnieniu od typowych rozwiązań korzystających z nadajników SSB sygnał w.cz. jest generowany bezpośrednio na częstotliwości wyjściowej za pomocą scalonego syntezyera Si5351A – bez konieczności mieszania i filtracji sygnału jak w nadajniku SSB. Metoda ta jest ograniczona do sygnałów zawierających w każdym momencie tylko jedną składową – jeden ton – o stałej amplitudzie. Mogą to być przykładowo emisje FT8, FT4, WSPR i inne z rodziny WSJT, ale system nie nadaje się do nadawania sygnałów PSK31, Olivii, Contestii, MT63 i oczywiście dla fonii. Brak możliwości kształtowania zboczy sygnału uniemożliwia pracę telegraficzną. Zmiany częstotliwości w emisjach obrazowych SSTV, faksymike itp. są zbyt szybkie dla przyjętej metody pomiaru częstotliwości. Zaletą rozwiązania jest (z samej zasady pracy) brak sygnałów nośnej i niepożądanego wstęgi bocznej.

Częstotliwość tonu modulującego jest określana poprzez pomiar odstępu czasu między przejściami przez zero i w ten sposób jest obliczana poprawka dla częstotliwości nadawanej w stosunku do częstotliwości nominalnej nadajnika. Dokładność pomiaru zależy (dzięki uśrednianiu wyniku) od liczby cykli uwzględnianej w jego trakcie. W oparciu o zmierzona poprawkę następuje przestrojenie syntezer Si5351A. Dla emisji o szybkościach transmisji większych niż FT8 j.np. RTTY należy w konfiguracji programu podać mniejszą liczbę próbek, aby czas pomiaru był krótszy niż cykl zmian stanu transmitowanego sygnału. Domyślnie dla FT8 pomiar jest wykonywany dla 480 próbek, czyli co 0,01 sekundy. Dla uzyskania stabilności niezbędnej w wąskopasmowych emisjach cyfrowych syntezer jest sterowany z generatora wzorcowego TCXO o częstotliwości 25 MHz. Dwufazowy sygnał wyjściowy syntezer (0 i 180° z wyjść Clk0 i Clk1) steruje przeciwsobny wzmacniacz mocy pracujący w klasie D na czterech polowych tranzystorach przełącznikowych typu BS170 (połączonych parami równolegle). W zależności od wariantu wybranego w trakcie montażu może on dostarczać mocy 5 W przy napięciu zasilania 9 albo 12 V. Na wyższych pasmach moc wynosi około 3,5 – 4 W. W odróżnieniu od nadajników SSB wzmacniacz ten nie musi pracować liniowo i dzięki temu osiąga się wyższą sprawność. Ponieważ sygnał m.cz. nie moduluje bezpośrednio nadajnika, a jest używany jedynie do pomiaru jego częstotliwości nie ma niebezpieczeństwa przemodulowania i powodowania zakłóceń w sąsiednich kanałach. Regulator poziomu modulacji w WSJT-X należy ustawić na maksimum, gdyż zbyt niski poziom modulacji może ujemnie odbić się na dokładności pomiaru częstotliwości. Syntezer dostarcza sygnału o stałej amplitudzie, dlatego też nie ma możliwości regulacji mocy wyjściowej. Na niektórych pasmach moc wyjściowa może być niższa od 5 watów i wtedy można spróbować lekkiego przestrojenia odpowiedniego filtra dolnoprzepustowego przez przesuwanie jego uzwojeń na rdzeniu. Jednak przekroczenie mocy 5 W może spowodować uszkodzenie tranzystorów stopnia końcowego. Dla poprawienia ich chłodzenia można wstawić większy radiator własnego wykonania. Pomiar częstotliwości tonu modulującego przy użyciu szybkiej transformaty Fouriera wymagałby znacznie większej mocy przetwarzania procesora dlatego też konstruktorzy zdecydowali się na pomiar okresu drgań.



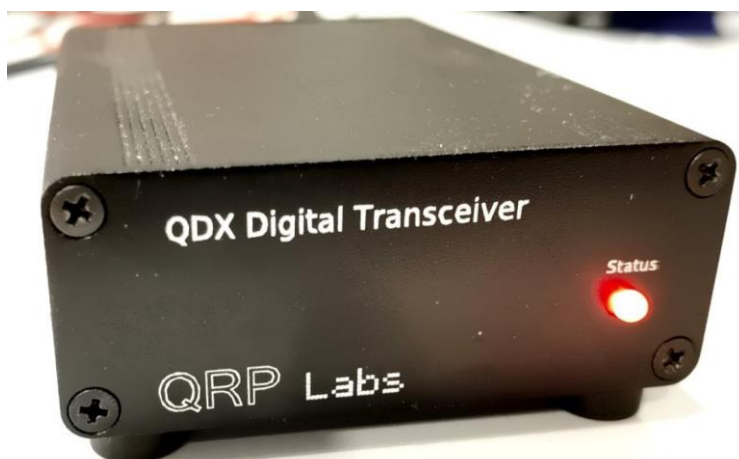
Fot. 6.2. Konstrukcja wewnętrzna

W transmisji danych dźwiękowych mikroprocesor zawarty w QDX symuluje podsystem dźwiękowy pracujący również z częstotliwością próbkowania 48 kHz i komunikujący się z PC przez złącze USB. Transmituje on do komputera strumień próbek pochodzących z przetwornika a-c i odbiera próbki sygnału modulującego. System ten należy wybrać w konfiguracjach WSJT-X, JS8Call, Fldigi lub innych stosowanych programów terminalowych dla emisji cyfrowych (jego nazwa zawiera oznaczenie

modelu QRP_Labs_QDX). Połączenie QDX z PC nie wymaga instalowania żadnych dodatkowych sterowników dla Windows ani dla Linuksa. Przez USB dostępne jest także złącze CAT używane do kluczowania nadajnika. Jako tranzystor wykonawczy służy przełącznikowy tranzystor MOSFET BSS123. Automatyczny sposób kluczowania – VOX – nie jest zalecany, gdyż włączenie nadajnika może nastąpić pod wpływem innych sygnałów pochodzących z komputera albo przypadkowych zakłóceń. Zestaw poleceń zdalnego sterowania jest podzbiorem używanego w TS-440 i TS-480. W przypadku wystąpienia jakichś problemów z kluczowaniem itp. można spróbować wybrać w ustawieniach WSJT-X drugi z modeli.

Pobór prądu przy odbiorze wynosi około 100 mA, a przy mocy nadawania 5 W i napięciu zasilania 9 V – około 1 A (0,7 A przy 12 V). Wymiary obudowy wynoszą 89 x 63 x 25 mm. Warto zamówić ją razem z zestawem gdyż płyta drukowana nie posiada otworów dla śrub i zamontowanie w innej obudowie może być trudniejsze.

Zestaw konstrukcyjny wymaga jedynie montażu elementów przewlekanych i nawinięcia kilku indukcyjności na rdzeniach pierścieniowych oraz dwóch transformatorów w.cz. Wszystkie elementy montowane powierzchniowo są już przylutowane fabrycznie. Uruchomienie radiostacji nie wymaga strojenia żadnych obwodów rezonansowych. Trzy filtry dolnoprzepustowe szóstego rzędu są automatycznie przełączane w zależności od wybranego pasma (80, 40 i 30/20 m), to samo następuje w wersji dla wyższych pasm. Instrukcja montażowa i aktualne wersje oprogramowania wewnętrznego są udostępniane w witrynie [6.1]. Wymiana oprogramowania jest nieskomplikowana dzięki temu, że pamięć QDX jest widoczna dla PC jako zewnętrzny nośnik USB.



Fot. 6.3. Na przedniej ścianie znajduje się tylko dioda sygnalizująca włączenie i (migając) nadawanie



Fot. 6.4. Gniazdko na tylnej ścianie

[6.1] <http://www.qrp-labs.com/qdx.html>

7. „SDR-Control” – program sterujący dla radiostacji Icoma

Wybór programów dla użytkowników komputerów firmy Apple jest znacznie mniejszy aniżeli dla korzystających z Windows. *SDR-Control* jest dostępny jedynie dla systemów iOS (komputerów iPad, telefonów iPhone) i macOS (od wersji 11 wzwyż). W warunkach domowych radiostacja jest połączona z komputerem w ramach lokalnej sieci WLAN, ale w warunkach polowych można też skonfigurować bezpośrednie połączenie między komputerem i IC-705 (w trybie „Access Point”).

Opracowany przez DL8MRE *SDR-Control* służy do sterowania przez sieć lokalną radiostacji IC-705, IC-7610 i IC-9700, odbiornika IC-R8600 oraz do sterowania przez złącze USB modelu IC-7300. W zależności od modelu urządzenia są połączone z siecią radiowo przez WiFi (IC-705) albo za pomocą kabla ethernetowego (IC-9700, 7610, R8600). DL8MRE jest również autorem programu *SmartSDR* przeznaczonego do sterowania radiostacjami *FlexRadio* i pracującego pod tymi samymi systemami operacyjnymi co *SDR-Control*.

Różnice między wersjami dla systemów iOS i macOS

W przeważającej części oba programy są identyczne, ale wersja dla systemu macOS oferuje dodatkowe możliwości. Pozwala ona na zdalne włączanie radiostacji, korzystanie ze zdalnej gałki RC-28, udostępnienia na ekranie więcej elementów obsługi, a poza tym posiada dekodery telegrafii.

Obie wersje programu są kompatybilne z najnowszymi modelami radiostacji pracującymi z cyfrową obróbką sygnałów. W starszych modelach brakuje wskaźników wodospadowych i innych funkcji wymaganych przez program.



Rys. 7.1. Okna główne programu dla iPada i MacOS

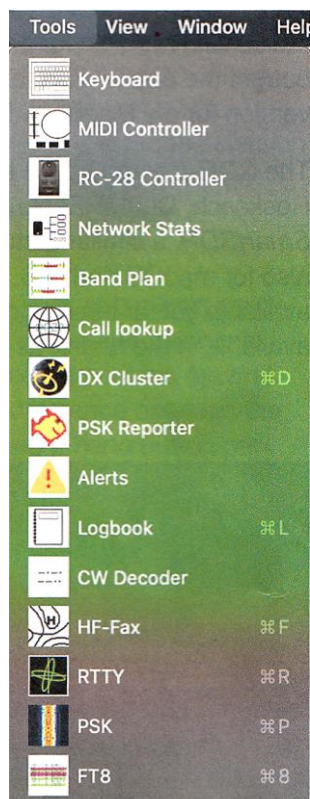
Konfiguracja połączenia radiostacji z siecią domową jest identyczna jak dla Icomowskiego programu RS-BA1 i jest ona opisana w instrukcjach obsługi. W przypadku gdy dostęp do domowej sieci został już skonfigurowany wystarczy tylko odczytać adres IP i kanały logiczne (ang. *port*) i wprowadzić je do konfiguracji *SDR-Control*. Wykorzystywane są trzy kanały logiczne UDP: do sterowania radiostacją, dla złącza szeregowego i dla transmisji dźwięku. Domyślnie w radiostacjach Icoma są to kanały o numerach 50001 – 50003. Program pozwala na skonfigurowanie dostępu do kilku radiostacji, ale

w danej chwili można korzystać tylko z jednej z nich. Instrukcje obsługi *SDR-Control* są dostępne w Internecie pod adresem [7.3], a jego konfiguracja nie jest trudna.

Operator może korzystać z analogowych emisji SSB i FM oraz z cyfrowego głosu w systemie D-STAR, a także z emisji FT8.



Rys. 7.2. Funkcje programu w wersji dla iPada



Rys. 7.3. Rozwijane menu na ekranie komputera MacBook Pro

Zdalne sterowanie przez Internet wymaga dodatkowo skonfigurowania przekierowywania adresów w modemie internetowym. W przypadku konfiguracji dla kilku TRX, dla każdego z nich należy podać

oddzielne numery kanałów logicznych i zaprogramować dla nich przekierowywanie danych. Możliwe jest także przypisanie radiostacjom publicznego adresu IP użytkownika i skorzystanie z usługi dyndns albo no-ip w celu uniezależnienia się od zmian adresu IP przez operatora internetowego.

Wariant dla systemu macOS pracuje pod jego wersją 11 lub nowszą. Działa on zarówno na procesorach Intele i Appla (M1, M2). Najnowsze wersje systemu operacyjnego pozwalają nawet na użycie *iPada* jako drugiego monitora dla komputera *MacBook Pro* itp.



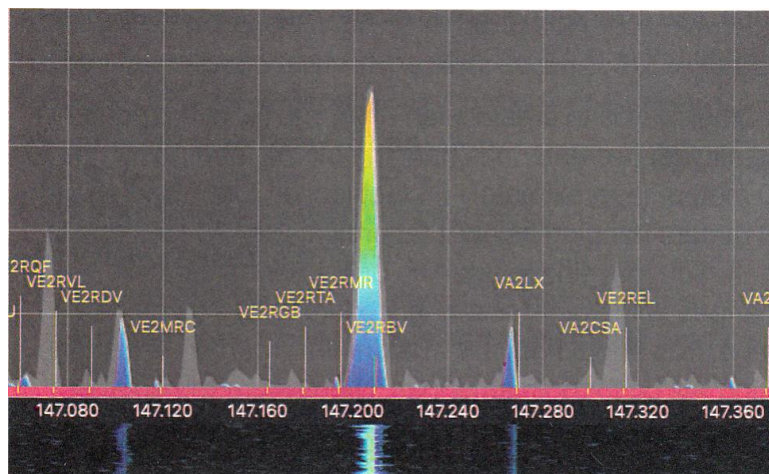
Rys. 7.4. Zdalne sterowanie przez MacBook Pro z gałką RC-28 i iPadem Pro jako drugim monitorem w trakcie łączności FT8

SDR-Control w eterze

Zdalna obsługa za pomocą obu wersji programu jest prosta i intuicyjna. Jakość dźwięku przekazywanego w obu kierunkach jest dobra – zarówno dla komputerów Mac jak i *iPada* – pod warunkiem korzystania z łącza internetowego dobrej jakości. Podłączenie zewnętrznego mikrofonu do *iPada* jest możliwe, ale nie jest to konieczne gdyż jakość dźwięku i jest tak dobra. W wersji dla *iPada* do kluczowania nadajnika służy przycisk ekranowy.

Przy gorszych połączeniach internetowych występują przerwy i opóźnienia w transmisji dźwięku i na wskaźniku wodospadowym, a nawet łączność całkiem się przerywa. Przy dobrym połączeniu internetowym wszystko funkcjonuje natomiast zgodnie z oczekiwaniami.

Pożyteczną funkcją jest wyświetlanie kanałów zapisanych w pamięci na tle wskaźnika wodospadowego. Pozwala to na wywołanie pamięci przez dotknięcie na ekranie (możliwe tylko dla odbiornika głównego).



Rys. 7.5. Znaki stacji na tle wodospadu IC-9700

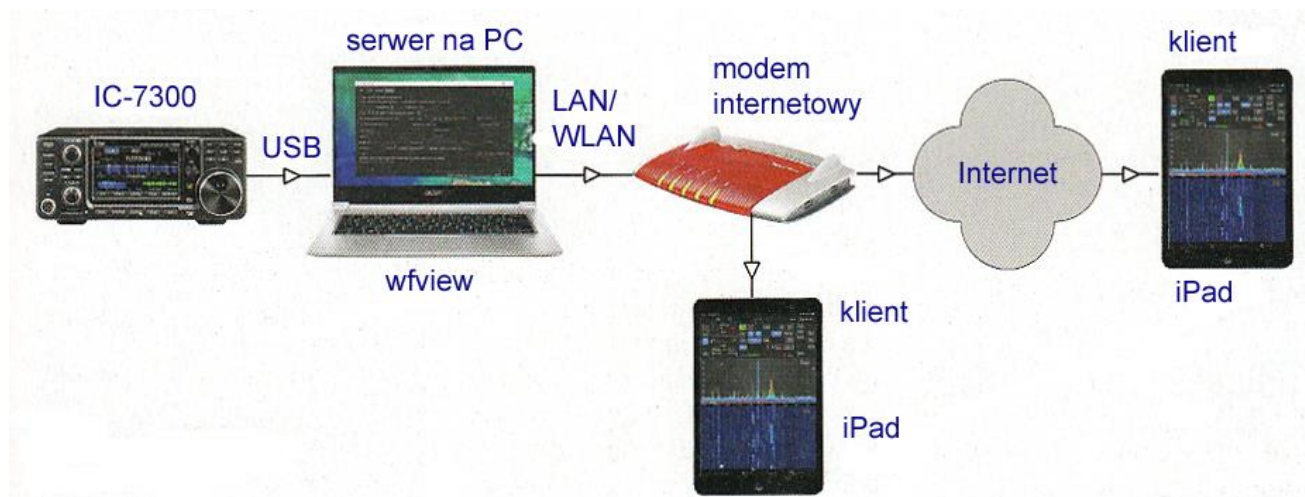
Oba warianty programu posiadają wbudowany terminal dla łączności FT8 i FT4, CW i RTTY, prowadzenie dziennika łączności, pozwalają na korzystanie ze skrzynek *DX-Cluster*, na meldowanie łączności w witrynie *PSKReporter* [7.5] i na dostęp do internetowych baz danych stacji amatorskich. Na tle wodospadu wyświetlane są zdekodowane znaki stacji, oprócz tego wyświetlane jest położenie stacji korespondenta na mapie i okna komunikatów FT8. W łącznościach telegraficznych konieczne jest korzystanie z przygotowanych wcześniej tekstów komunikatów.

Program pozwala na wygodne korzystanie z radiostacji Icoma z dowolnego miejsca w domu lub w ogrodzie oraz na zdalne sterowanie stacji przez Internet. Jest on dostępny (odpłatnie) jedynie w sklepie internetowym AppStore.

Zdalne sterowanie IC-7300

W odróżnieniu od poprzednio wymienionych modeli sprzętu IC-7300 nie posiada w oprogramowaniu serwera http, a jedynym złączem jest złącze USB. Połączenie radiostacji z siecią, tak aby mogła być zdalnie sterowana wymaga zastosowania pośredniczącego komputera PC z zainstalowanym programem *wfview*. Program ten zawiera serwer http, a sam komputer zapewnia połączenie z domową siecią przez złącze ethernetowe albo przez WiFi. *Wfview* pozwala też na sterowanie radiostacji lokalnie przez PC. Alternatywnie jako serwer pośredniczący może służyć także *Malina*.

W symulowanym łączu COM przez USB należy ustawić szybkość transmisji 115200 bodów gdyż przy niższych szybkościach transmisji niemożliwe jest przekazywanie do komputera treści wskaźnika wodospadowego. Wskazania wodospadu są zresztą i tak wolniejsze ze względu na niższą niż w przypadku bezpośredniego połączenia sieciowego szybkość transmisji przez złącze USB.



Rys. 7.6. Konfiguracja sprzętu dla zdalnego sterowania IC-7300

[7.1] „*SDR-Control* for Icom Radios – iOS and macOS Versions”, Pascal Villeneuve, VA2PV i Michael Crestohl, W1RC, QST 11/2022, str. 45

[7.2] „iPod-SDR-Control-App für IC-7300”, Werner Schnorrenberg, DC4KU, CQDL 4/2023, str. 23

[7.3] <https://manuals.rokosch.de/sdrcontrol>

[7.4] www.rokosch.de/sdr-control

[7.5] www.pskreporter.info

Literatura i adresy internetowe

Roczniki 2016 – 2023 Świata Radio

Roczniki 2019 – 2023 Funkamateura

Roczniki 2019 – 2023 CQDL

Roczniki 2019 – 2022 QST

Roczniki 2019 – 2023 QSP

„LoRaWAN-Knoten im IoT”, Claudius Kühnel, Elektor, Akwizgran [Aachen] 2021, ISBN 978-3-89576-467-7

Instrukcje obsługi radiostacji i programów

Strony internetowe podane na końcu rozdziałów

W serii „Biblioteka polskiego krótkofalowca” dotychczas ukazały się:

- Nr 1 – „Poradnik D-STAR”, wydanie 1 (2011), 2 (2015), 3 (2019) i 4 (2021)
- Nr 2 – „Instrukcja do programu D-RATS”
- Nr 3 – „Technika słabych sygnałów” Tom 1
- Nr 4 – „Technika słabych sygnałów” Tom 2
- Nr 5 – „Łączności cyfrowe na falach krótkich” Tom 1
- Nr 6 – „Łączności cyfrowe na falach krótkich” Tom 2
- Nr 7 – „Packet radio”
- Nr 8 – „APRS i D-PRS”
- Nr 9 – „Poczta elektroniczna na falach krótkich” Tom 1, wydanie 1 (2012)
- Nr 10 – „Poczta elektroniczna na falach krótkich” Tom 2, wydanie 1 (2012)
- Nr 11 – „Słownik niemiecko-polski i angielsko-polski” Tom 1
- Nr 12 – „Radiostacje i odbiorniki z cyfrową obróbką sygnałów” Tom 1
- Nr 13 – „Radiostacje i odbiorniki z cyfrową obróbką sygnałów” Tom 2
- Nr 14 – „Amatorska radioastronomia”
- Nr 15 – „Transmisja danych w systemie D-STAR”
- Nr 16 – „Amatorska radiometeorologia”, wydanie 1 (2013) i 2 (2017)
- Nr 17 – „Radiolatarnie małej mocy”
- Nr 18 – „Łączności na falach długich”
- Nr 19 – „Poradnik Echolinku”
- Nr 20 – „Arduino w krótkofalarstwie” Tom 1
- Nr 21 – „Arduino w krótkofalarstwie” Tom 2
- Nr 22 – „Protokół BGP w Hamnecie”
- Nr 23 – „Technika słabych sygnałów” Tom 3, wydanie 1 (2014), 2 (2016) i 3 (2017)
- Nr 24 – „Raspberry Pi w krótkofalarstwie”
- Nr 25 – „Najpopularniejsze pasma mikrofalowe”, wydanie 1 (2015) i 2 (2019)
- Nr 26 – „Poradnik DMR” wydanie 1 (2015), 2 (2016) i 3 (2019), nr 326 – wydanie skrócone (2016)
- Nr 27 – „Poradnik Hamnetu” wydanie 1 (2015) i 2 (2021)
- Nr 28 – „Budujemy Ilera” Tom 1
- Nr 29 – „Budujemy Ilera” Tom 2
- Nr 30 – „Konstrukcje D-Starowe”
- Nr 31 – „Radiostacje i odbiorniki z cyfrową obróbką sygnałów” Tom 3
- Nr 32 – „Anteny łatwe do ukrycia”
- Nr 33 – „Amatorska telemetria”, wydanie 1 (2017) i 2 (2022)
- Nr 34 – „Poradnik systemu C4FM”, wydanie 1 (2017), 2 (2019) i 3 (2021)
- Nr 35 – „Licencja i co dalej” Tom 1
- Nr 36 – „Cyfrowa Obróbka Sygnałów”
- Nr 37 – „Telewizja amatorska”
- Nr 38 – „Technika słabych sygnałów” Tom 4, wydanie 1 (2018), 2 (2020) i 3 (2022)
- Nr 39 – „Łączności świetlne”
- Nr 40 – „Radiostacje i odbiorniki z cyfrową obróbką sygnałów” Tom 4
- Nr 41 – „Licencja i co dalej” Tom 2
- Nr 42 – „Miernictwo” Tom 1
- Nr 43 – „Miernictwo” Tom 2
- Nr 44 – „Miernictwo” Tom 3
- Nr 45 – „Testy sprzętu” Tom 1
- Nr 46 – „Testy sprzętu” Tom 2
- Nr 47 – „Licencja i co dalej” Tom 3
- Nr 48 – „Jonosfera i propagacja fal”
- Nr 49 – „Anteny krótkofalowe” Tom 1, wydanie 1 (2020) i 2 (2023)
- Nr 50 – „Anteny ultrakrótkofalowe” Tom 1, wydanie 1 (2020) i 2 (2022)
- Nr 51 – „Anteny krótkofalowe” Tom 2, wydanie 1 (2020) i 2 (2023)
- Nr 52 – „Anteny ultrakrótkofalowe” Tom 2, wydanie 1 (2020) i 2 (2023)
- Nr 53 – „Anteny mikrofalowe”

- Nr 54 – „Proste odbiorniki amatorskie” Tom 1
- Nr 55 – „Proste odbiorniki amatorskie” Tom 2
- Nr 56 – „Proste nadajniki amatorskie” Tom 1
- Nr 57 – „Proste nadajniki amatorskie” Tom 2
- Nr 58 – „Mini- i mikrokomputery w krótkofalarstwie” Tom 1
- Nr 59 – „Mini- i mikrokomputery w krótkofalarstwie” Tom 2
- Nr 60 – „DX-y w C4FM”
- Nr 261 – „Poradnik DMR” Tom 1, z nru 26, wydanie 1 (2021)
- Nr 262 – „Poradnik DMR” Tom 2, z nru 26, wydanie 1 (2021)
- Nr 63 – „Testy sprzętu” Tom 3
- Nr 64 – „Poczta elektroniczna na falach krótkich”, z nrów 9 i 10, wydanie 2 (2022)
- Nr 65 – „Testy sprzętu” Tom 4
- Nr 66 – „Mieszanka firmowa” Tom 1
- Nr 67 – „Mieszanka firmowa” Tom 2

