

SPREJEMANJE IN DEKODIRANJE ADS-B SIGNALA

Matej Planinšek¹, Nace Pogačar², Katja Fon Mervič³

^{1,2,3} Univerza v Ljubljani, Fakulteta za elektrotehniko
Tržaška 25, 1000 Ljubljana, Slovenija

E-pošta: mp9806@student.uni-lj.si, np6977@student.uni-lj.si, kf7808@student.uni-lj.si

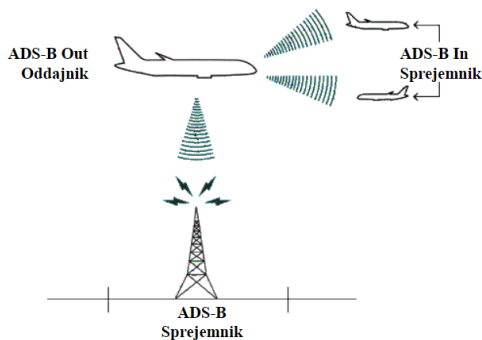
ADS-B signals; receiving and decoding

Abstract. ADS-B stands for Automatic Dependent Surveillance – Broadcast. This is a satellite-based surveillance system. Nowadays, majority of aircrafts broadcast ADS-B messages constantly. Parameters such as velocity, position and identification are transmitted through Mode S Extended Squitter (1090 MHz).

1 Uvod

S pomočjo ADS-B izboljšujemo varnost letalskega prometa, saj dobimo vse potrebne informacije za nadzor zračnega prostora v realnem času.

ADS-B sestavljata "ADS-B In" in "ADS-B Out". Z uporabo "ADS-B Out" vsako letalo preko oddajnika periodično oddaja informacije o zračni liniji, poziciji, višini in hitrosti. Te informacije so večinoma bolj natančne od informacij, ki jih lahko zberemo preko radarja. S pomočjo "ADS-B In" pa letalo sprejema ADS-B signale drugih zračnih plovil v bližini.



Slika 1. Prikaz ADS-B komunikacije

2 Struktura ADS-B sporočil

ADS-B sporočilo je dolgo 112 bitov. Sestavlja ga 5 glavnih delov:

DOWNLINK FORMAT	ZMOGLJIVOST	ICAO NASLOV	SPOROČILO	PARITETA
5 (1-5)	3 (6-8)	24 (9-32)	56 (33-88)	24 (89-112)

Slika 2. Razdelitev ADS-B sporočila na 5 delov

Sporočilo se začne z Downlink format 17 in zasede prvih 5 bitov, kar nam pove, da je to ADS-B sporočilo.

Sledijo 3 (6 - 8) biti namenjeni informaciji o levelu transponderja. Različne situacije so označene z vrednostmi od 1 do 7 (dec).

Biti od 9 do 32 so namenjeni ICAO naslovu, kjer je shranjen edinstven naslov zračnega plovila, ki ostane enak skozi celoten čas obratovanja.

V prvih 5 bitih dela s sporočilom dobimo informacijo o tipu kodiranja, s katerim lahko razberemo vsebino sporočila.

Zadnjih nekaj bitov je namenjenih pariteti.

Tabela 1. Povezava med biti in informacijo v sporočilu

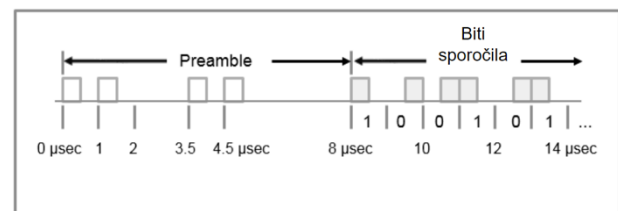
Koda	Informacija
1 - 4	Identifikacija zračnega plovila
5 - 8	Položaj na površini
9 - 18	Položaj v zraku (z višino)
19	Hitrost v zraku
20 - 22	Rezervirano
28	Status plovila
29	Informacija o ciljni lokaciji in status
31	Stanje delovnja letala

3 Postopek dekodiranja signala

Iz standarda za Mode S Extended Squitter so specifikacije transmisije transponderja sledeče:

- Frekvenca nosilca: 1090 MHz
- Modulacija: Amplitudno preklapljanje (ASK)
- Simbolna frekvenca: 2MHz (simbol: pulz ali ne pulz)
- Hitrost prenosa podatkov: 1 Mbit/s
- Dolžina sporočila: 56 usec ali 112 usec
- 24-bitni CRC checksum

Znotraj 1 usec intervala imamo dva možna vzorca. Logična 1 je ON za prvo polovico usec in OFF za drugo polovico usec. Logična 0 pa je OFF za prvo polovico in ON za drugo polovico usec.



Slika 3. Prikaz poteka signala

Za opazovanje zelo kratkih signalov (112 usec) v dobri resoluciji, potrebujemo SDR s stopnjo vzorčenja vsaj 2 MHz. Teoretično bi po Nyquistovem kriteriju

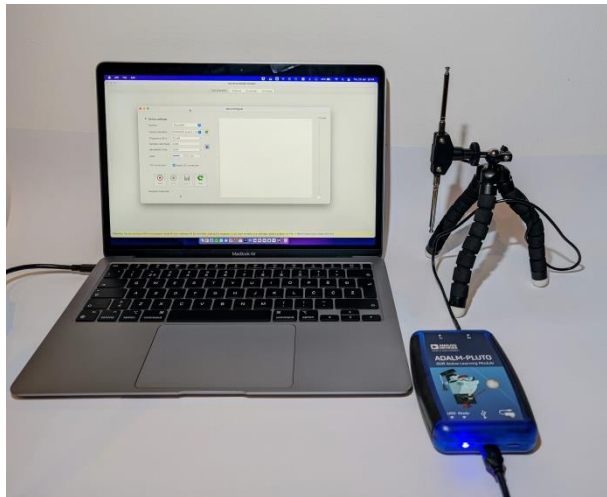
zadoščala že pasovna širina 1Mhz, vendar v praksi zaradi neidealnega razmerja SNR, to ne zadošča.

Signali, ki jih zaznamo so lahko zelo šibki, če je letalo, ki jih oddaja, daleč stran od sprejemnika. Zato potrebujemo Preamble del, da identificiramo začetek signala pa tudi High in Low amplitudo Mode S transmisije.

Komercialni programi s kompleksnejšim algoritmom dekodiranja uspešno delujejo s frekvenco vzorčenja 2Mhz. V našem primeru pa smo za namen preprostejšega dekodiranja signal vzorčili s frekvenco 4Mhz. To nam je omogočilo bolj jasno razviden signal in hkrati ni vneslo preveč visokofrekvenčnega šuma.

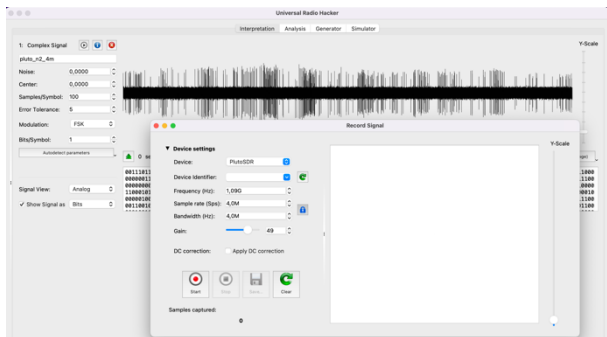
4 Meritve in obdelava

Za snemanje signala smo uporabili program Universal Radio Hacker (URH) in Pluto SDR, na katerega smo priključili anteno dolžine 7 cm. Anteno smo postavili v zunanje okolje, z odprtim pogledom na nebo, čimbolj stran od motečih dejavnikov.



Slika 4. Postavitev za snemanje signala

Centralno frekvenco smo nastavili na 1.09 GHz, stopnjo vzorčenja pa na 4 MHz.

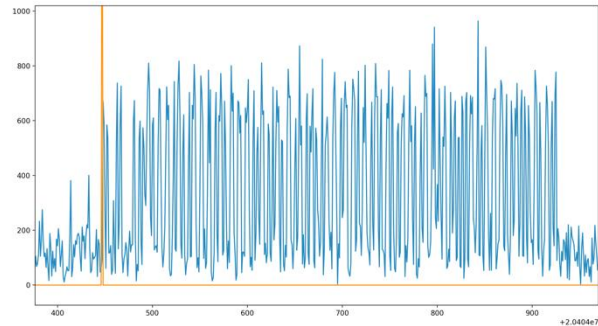


Slika 5. Nastavitve v programu URH

Prejet signal smo najprej shranili v .wav datoteko, ki je vsebovala vzorce I in Q veje. Signal smo nato uvozili v Python okolje in najprej izračunali vektor amplitude kot geometrijsko vsoto I in Q veje

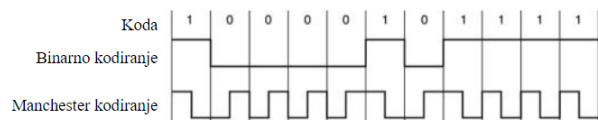
$$A = \sqrt{I^2 + Q^2} . \quad (1)$$

S pomočjo povprečja in maksimalne vrednosti amplitudnega vektorja smo izračunali prag za zaznavo preamble pulzov. Nato smo s kodo preverili celoten signal in poiskali pravi začetek podatkovnega paketa. Začetno mesto smo določili tam, kjer smo razpoznali pravilne preamble pulze, ki so segali čez prej določen prag. Začetek pravega signala smo označili z oranžno.



Slika 6. Košček .wav datoteke kjer smo zaznali preamble pulze – začetek označen z oranžno

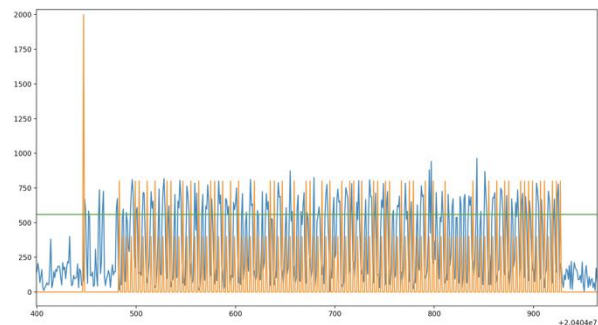
Za tem smo analizirali del signala za zaznanim začetkom paketa. Preverili smo ga na fiksno frekvenco zaradi lažjega postavljanja indeksov, nato pa na mestih, kjer smo pričakovali simbole, dekodirali po Manchester kodiranju. Pri tem načinu kodiranja razpoznamo bit kot 1, če je povprečje prve polperiode časa enega bita višje od druge polperiode. In obratno za 0.



Slika 7. Prikaz Manchester kodiranja

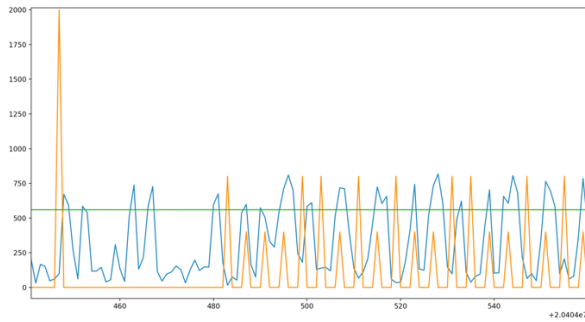
Za preverjanje delovanja smo na grafu izrisali amplitudi vektor, vrednost praga, pravilni začetek paketov in dekodirane bite.

Signal smo prikazali z modro, prag za zaznavanje preamble z zeleno, z oranžno pa dekodirane bite. Vrednost 400 na y osi predstavlja bit 0, vrednost 800 bit 1 in vrednost 2000 začetek pravilnega paketa.



Slika 8. Signal (modra), prag (zeleno) in dekodiran signal (oranžna)

Signal smo približali, ter še sami preverili, če dekodiranje deluje pravilno.



Slika 9. Približan začetek signala; signal (modra), prag (zelen) in dekodiran signal (oranžna)

Po dekodiranju vseh 112 bitov, smo podatkovni paket preverili s pomočjo CRC funkcije knjižnice pyModeS, ki smo jo uporabili za dekodiranje paketov. Če je bil paketek pravilen, smo ga dekodirali v berljiv format in izpisali v terminal.

Postopek smo ponavljali do konca posnetega signala.

Spodaj je prikazan primer izpisa nadmorske višine, identitete letala (callsign) in hitrosti.

```
loaded file...
Sample rate 4.000000 MS/s
Cnt samples 31064064
Duration: 7.766016 s
number of samples: 31064064 which is: 31.064064 Msamples
Processing IQ data...
calculating threshold params...
avg, max, thr 120.5866225896025 1876.6984307554583 559.6145746310665
finding ADS-B frames...
Got valid frame ##### ---->:)
Message: 8d4ca9aa58bf02dbfd7e8c3e84d3
ICAO address: 4ca9aa
Downlink Format: 17
Protocol: Mode-S Extended Squitter (ADS-B)
Type: Airborne position (with barometric altitude)
CPR format: Even
CPR Latitude: 0.7148284912109375
CPR Longitude: 0.747161865234375
Altitude: 37000 feet
```

Slika 10. Del izpisa v terminal; Izpis višine

```
Got valid frame ##### ---->:)
Message: 8d4ca9aa234994b4d965204567b7
ICAO address: 4ca9aa
Downlink Format: 17
Protocol: Mode-S Extended Squitter (ADS-B)
Type: Identification and category
Callsign: RYR46VT_
```

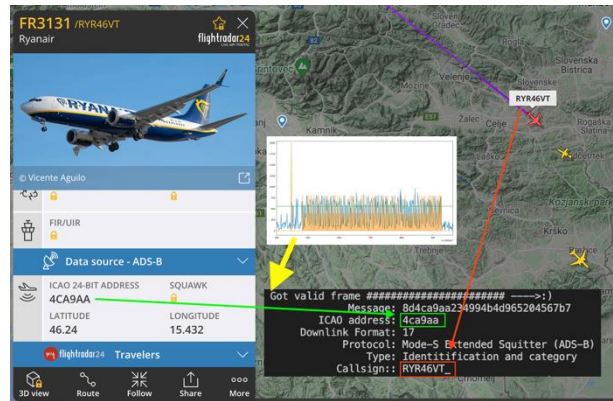
Slika 11. Izpis identitete letala

```
Got valid frame ##### ---->:)
Message: 8d4ca9aa99117ba490040809d9f2
ICAO address: 4ca9aa
Downlink Format: 17
Protocol: Mode-S Extended Squitter (ADS-B)
Type: Airborne velocity
Speed: 477 knots
Track: 127.59 degrees
Vertical rate: 0 feet/minute
Type: Ground speed
```

Slika 12. Izpis hitrosti letala

Ker smo želeli preveriti pravilnost delovanja naših meritev, smo na spletni strani FlightRadar24 preverili, če to letalo res leti nad nami.

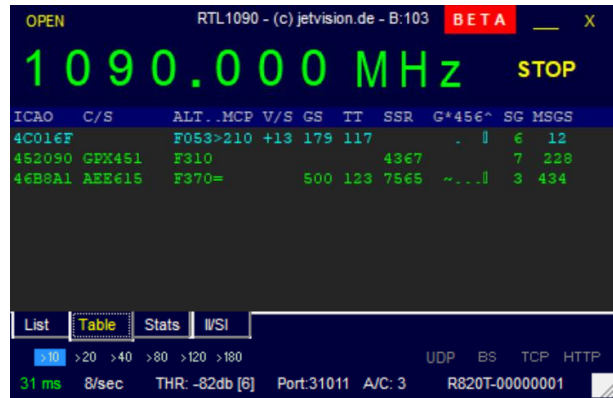
Kot lahko vidimo iz slike 13 je bilo letalo RYR46VT (ICAO: 4CA9AA) takrat res v bližini. Kar pomeni, da naš algoritem deluje pravilno.



Slika 13. Prikaz na spletni strani FlightRadar24

5 Dodatek

Program, ki naredi vse zgoraj opisano, na trgu že obstaja pod imenom 'RTL1090'. Zaradi veliko kompleksnejšega algoritma od našega, deluje tudi s SDRji, ki imajo najvišjo stopnjo vzorčenja 2 MHz. Prav tako zna dekodirati zelo šibke signale, ki jih nam ni uspelo.



Slika 14. Prikaz delovanja RTL1090

Kot primer še enega komercialnega sistema, ki deluje na osnovi ADS-B sporočil, smo uporabljali tudi FlightRadar24. Ta podatke zajema iz mreže sprejemnikov, ki se ji lahko pridruži vsak. Zahteve za pridružitve so namenska strojna oprema ali naprava RTL-SDR priključena na Raspberry Pi računalnik, ki izvaja že pripravljeno programsko opremo specifično za FlightRadar24.

6 Zaključek

V okviru te seminarske naloge smo uspešno sprejeli in dekodirali ADS-B signale letal v bližini in preverili pravilnost podatkov s sistemom FlightRadar24.

Ugotovili smo, da je v spletni literaturi zelo pomankljivo opisana radiofrekvenčna stran sporočil, predvsem glede tipa modulacije in spektra signala. Manjkajoče podatke smo s pomočjo opazovanja signala dopolnili, nato pa določili teoretični potek in spekter signala.

Po prvotnih neuspehih poskusih zajema in dekodiranja signala z napravo RTL-SDR smo signal uspešno zajeli z naprednejšim Pluto SDR. Ta nudi višjo

frekvenco vzorčenja, kar nam je omogočilo enostavnejše dekodiranje signala. Na primeru programa RTL1090 je očitno, da je zajem pri manjši vzorčni frekvenci izvedljiv, vendar potrebuje kompleksnejše algoritme, ki bi jih lahko implementirali v prihodnosti.

Signal smo za to nalogo posneli in nato analizirali posnetek, kar nam je omogočilo hiter razvoj algoritma na konsistentnih podatkih, hkrati pa tudi enostavno vizualizacijo.

V nadaljevanju bi lahko sistem izboljšali do te mere, da bi deloval v realnem času.

Literatura

- [1] Junzi Sun, Jacco Hoekstra, Joost EllerBroek, ADS-B Decoding Guide, Release 0.3, 30. januar 2017, Delft, Nederland
- [2] Junzi Sun, ADS-B Basics
<https://mode-s.org/decode/content/ads-b/1-basics.html>
- [3] ADS-B Airplane Tracking Tutorial
https://wiki.analog.com/resources/eval/user-guides/picozed_sdr/tutorials/adsb, 14. februar 2021
- [4] Flightradar24- how does it work? Part 2, ADS-B protocol
<https://habr.com/en/post/447078/>, 7. april 2019
- [5] Decoding ads-b with free qirx software
<https://dk8ok.org/2021/04/17/decoding-ads-b-with-free-qirx-software/>, 17. April 2021