Learning Objective 6.7: I can receive and decode ADS-B messages from aircraft transponders.

Show your work. Write a **Documentation** line at the end (**None** if you did not collaborate).

1. **Reading the chips.** A receiver samples a message at 2 MS/s, so each 0.5 µs half of a bit is one sample, or chip. After the preamble, the first 16 chips it reads, in pairs, are

   `10 01 01 01 10 10 01 10`

   (a) Write the 8 bits these chips carry, and the byte in hex.

   In PPM a pulse in the first half (10) is a 1 and a pulse in the second half (01) is a 0, so the bits are `1000 1101`, which is 8D in hex.

   bits: 1000 1101

   hex: 8D

   (b) The first 5 bits are the Downlink Format and the next 3 the capability. Find both, and say whether this is an ADS-B extended squitter.

   DF $= 10001_2 = 17$ and CA $= 101_2 = 5$. DF 17 is the extended squitter, so yes.

   DF: 17

   CA: 5

   (c) A noise spike makes the sixth pair read `11`: both chips high. In words, explain what the receiver does with that bit, and why the parity check matters here.

   The receiver compares the two chips and calls the bit by whichever is larger, so it guesses: with both high it may call the bit wrong. It cannot tell from the chips alone. The 24 parity bits catch the wrong bit: the remainder no longer matches, and the decoder drops the whole message rather than report a wrong altitude or address.

2. **The message.** gr-adsb's ADS-B Demod passes on these bits, written in hex:

`8DAE07C220483234D71CA031ACD8`

The 28 hex digits are the 112 bits: DF and CA (8 bits), ICAO (24), ME (56), and PI (24).

(a) Split the message into its ICAO address, ME field, and parity, in hex.

After the first byte, 8D: ICAO is the next 6 hex digits, AE07C2; ME is the next 14, 20483234D71CA0; PI is the last 6, 31ACD8.

ICAO: AE07C2

PI: 31ACD8

(b) The type code is the first 5 bits of ME. Find it, and say which kind of message this is.

ME begins with 20 hex, 0010 0000, so the type code is $00100_2 = 4$: an identification message, which carries a callsign.

4, identification

(c) After the type code and 3 category bits, the callsign is eight 6-bit characters. In the Mode S character set, 1 to 26 are A to Z, 48 to 57 are the digits 0 to 9, and 32 is a space. The eight values are 18, 3, 8, 52, 53, 49, 50, 32. Write the callsign.

18 is R, 3 is C, 8 is H, and 52, 53, 49, 50 are the digits 4, 5, 1, 2; 32 is a space. The callsign is RCH4512, a USAF airlift flight.

RCH4512

3. **Receiver settings.** The L18 receiver runs at 2 MS/s.

   (a) How many samples does one whole message take, preamble included? Set the triggered time sink's number of points to show one whole message with about 10 µs to spare.

   A message is $8 + 112 = 120$ µs, and at 2 MS/s that is 240 samples. With 10 µs (20 samples) to spare, about 260 points.

   samples: 240

   points: about 260

   (b) A classmate sets the PlutoSDR and the gr-adsb blocks to 1 MS/s to save processing. Explain, in words, why the ADS-B Demod can no longer read the bits.

   At 1 MS/s there is one sample per bit, not one per half bit, so the receiver can no longer compare the first half of each bit with the second. Each sample may land on the pulse or on the quiet half, and nothing tells it which half held the pulse. PPM needs at least two samples a bit; in GRC the ADS-B Framer stops with the error `slice step cannot be zero`.

   (c) On a run, the 20 ms plot of power shows a noise floor near 0.02 and messages whose pulses reach between 0.1 and 0.8. The ADS-B Framer's threshold is 0.3. Choose a better threshold and say what the Aircraft Table would show with each setting.

   At 0.3 the Framer finds only the stronger aircraft, whose pulses reach above it; the weaker ones never start a burst, so they never reach the table. A threshold near 0.05, above the floor but below the weakest pulses, frames every message. Much lower and the noise itself crosses it, the Framer sees pulses everywhere, and fewer real preambles are found, so even strong aircraft decode less often.

4. **What noise does.** You run `L18-adsb-rx.grc` on the simulated recording and raise the Added noise slider a step at a time.

   (a) Describe what happens to the 20 ms plot, the Messages label, and the Aircraft Table as the noise rises.

   The noise floor in the plot rises toward the spikes, and the weaker spikes sink into it. The count of messages decoded drops, and the table starts over and fills with fewer aircraft: the strongest ones stay and the weakest ones never return.

   (b) Explain, in words, why the weakest aircraft drop out first, and what that means for a real receiver.

   A weak aircraft's pulses stand only a little above the noise, so a noise peak in the quiet half of a bit beats the pulse in some pair of almost every message, the parity fails, and the message is dropped. A strong aircraft's pulses are far above the noise, so its bits survive. On a real receiver the weak aircraft are the distant ones, so the range at which aircraft decode shrinks as the noise rises.

5. **Generating and locating.** Anyone with an SDR can receive ADS-B, and a flowgraph can generate it.

   (a) The L18 generator writes a file, and `L18-adsb-tx-pluto.grc` sends the same aircraft from a PlutoSDR at 915 MHz. Explain why the course never transmits them at 1090 MHz.

   Air traffic control and aircraft collision-avoidance systems use 1090 MHz, and nothing in an ADS-B message proves who sent it, so a transmitted message could put a false aircraft on a controller's display or another aircraft's. Transmitting there is illegal without authorization and unsafe.

   (b) gr-adsb shows an aircraft's callsign as soon as one message arrives, but its position only after it holds an even and an odd position message from that aircraft. Explain, in words, why one position message is not enough, and what a receiver that used its own location as a reference instead would get wrong for an aircraft 500 nautical miles away.

   A position message carries the latitude and longitude only within one of many zones (Compact Position Reporting), not which zone. Even and odd messages use two zone sizes, and together they pin down the zone, wherever the aircraft is. A receiver with a reference picks the zone nearest the reference, which is right for any aircraft within about 180 nautical miles. An aircraft 500 nautical miles away lies in a different zone, so it would be placed in the wrong one, badly wrong.