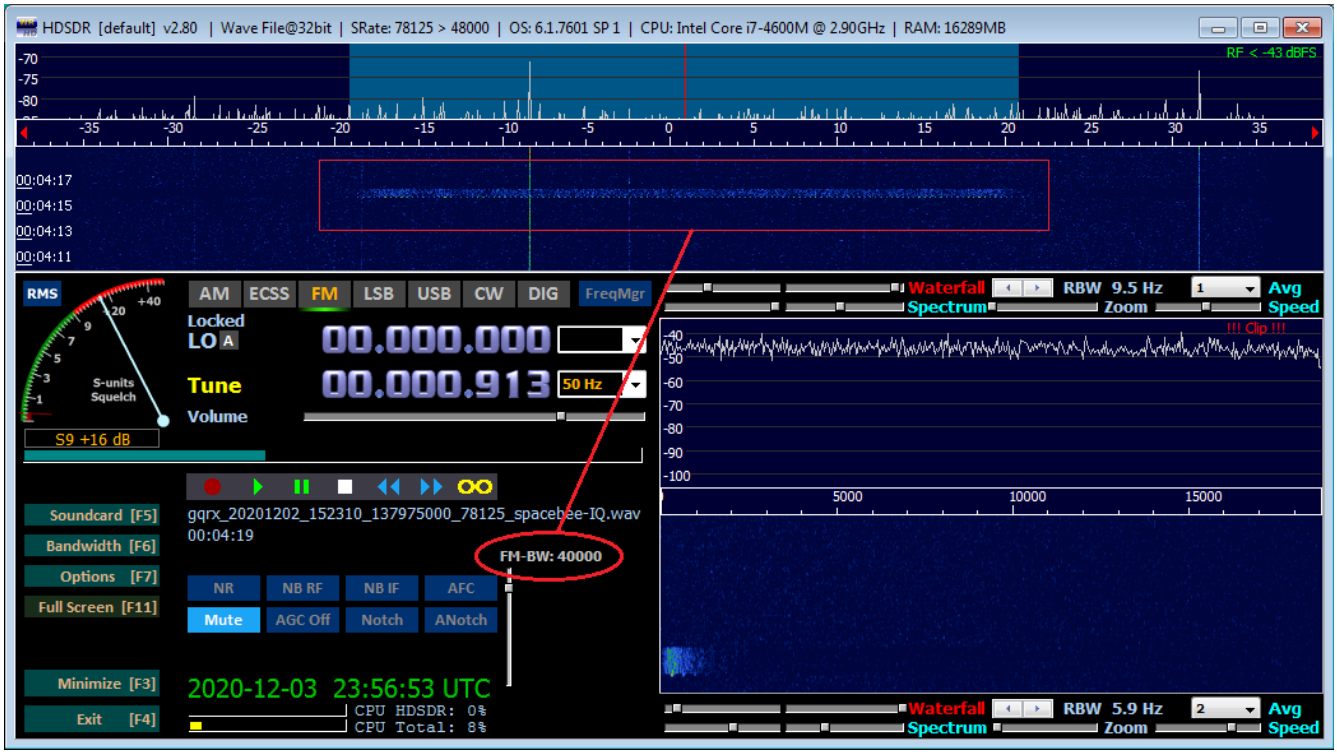
Receiving signals is cool. Decoding the data in a signal (even partially) is even cool-er.

Where do you start when trying to determine the parameters to decode an unknown signal? Well, you start with what you DO know. Then Google the rest. If this kind of thing interests you, I hope the following overview will assist you with similar efforts of your own.

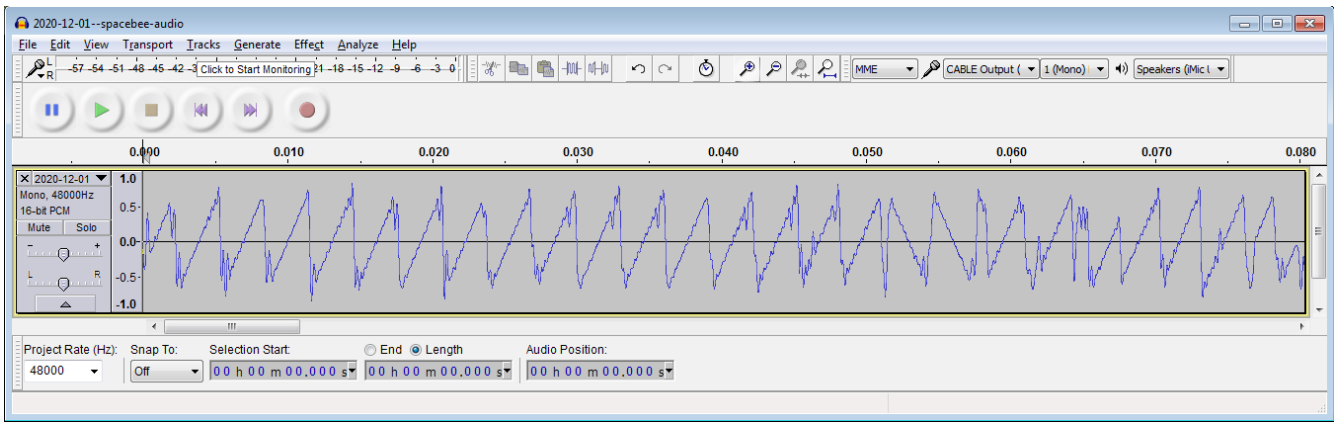
After observing wide-band downlinks from the most recent **SpaceBee** satellites in the 137 MHz band, the question of course is “can we decode it”?

On replay of a recorded I/Q file, here is one of the downlink packets in question:



So, we now have our first piece of information… the **bandwidth** is approximately **40k**.

We don’t know what the modulation is, but to start we can at least demod the signal as FM to see if it looks like anything seen previously. Here is a view in Audacity of the audio file output from an FM demod of our unknown signal:



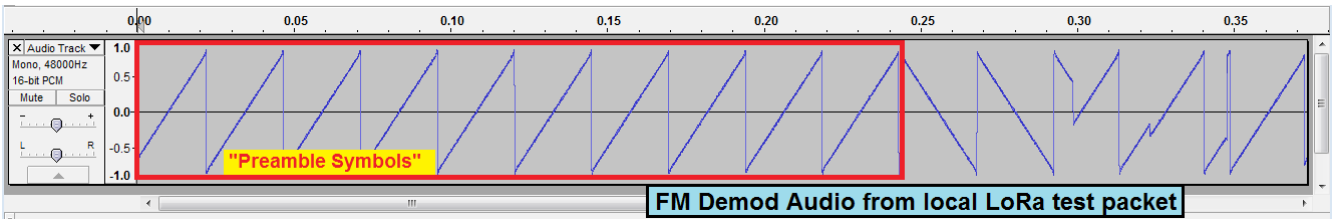
From experience, this sequence -- a number of up-chirps (“preamble”) followed by some number of less uniform up & down-chirps is recognizable as a “LoRa” packet.
Since the list of standard LoRa bandwidths is:

7.8, 10.4, 15.6, 20.8, 31.25, 41.7, 62.5, 125.0, 250.0 and 500.0 kHz

we now have our first concrete info: this signal is **LoRa** with a **bandwidth of 41.7 k**.

What else can we determine from this FM audio demod? Another LoRa parameter is the number of “Preamble” symbols that idle before the variable payload portion of the packet.

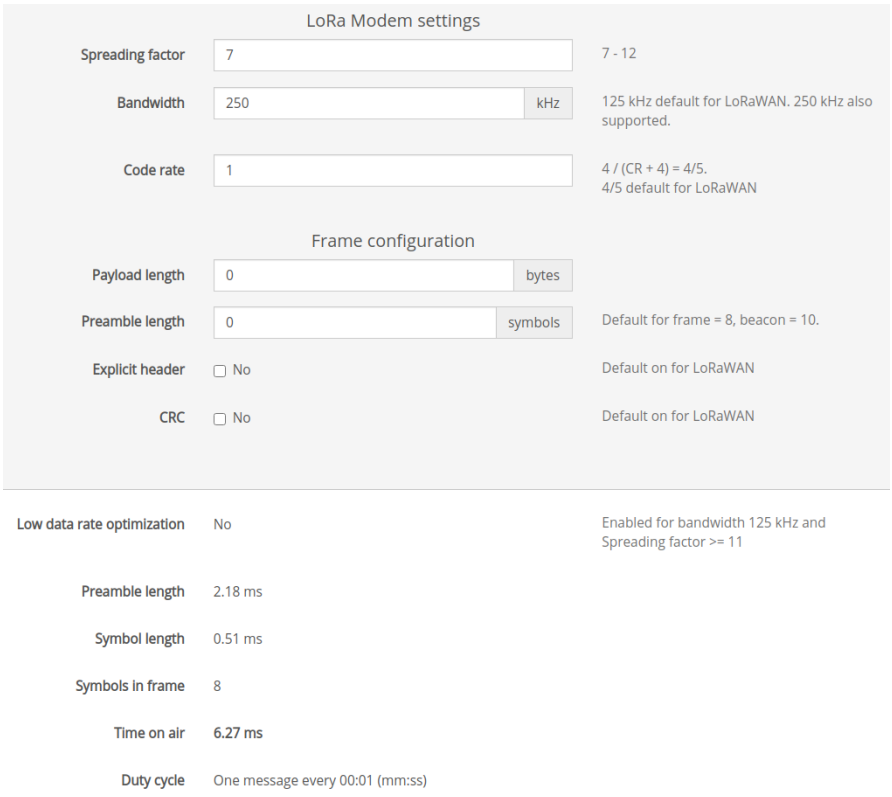
Using the LoRa hat on an Arduino Uno, through trial & error we can transmit LoRa test packets with various preamble lengths until we match the same number of idle up-chirps seen on our SpaceBee downlink. The default value of “8” was too small. The pefect match was found with the number of symbols in the preamble being “15”. Here is an example of those uniform up-chirp “preamble” symbols from a test packet:



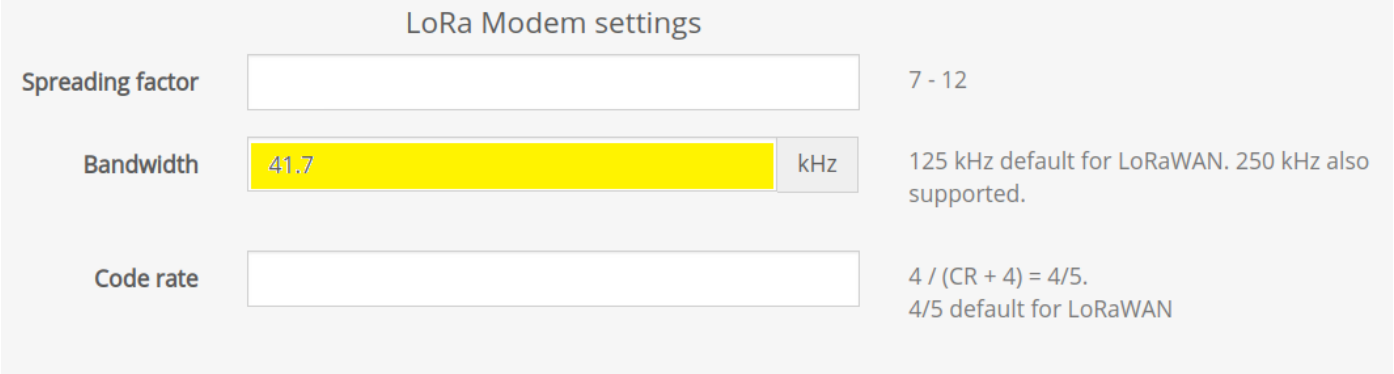
Just a reminder, LoRa is NOT “FM”, but we can use the FM demod of our signal to learn things. And no great science on the determination of the number of preamble symbols… just transmit test packets with various settings until you have a match.

But how to determine the other LoRa parameters required to decode the signal? The following website provides a very useful calculator:

https://www.loratools.nl/#/airtime

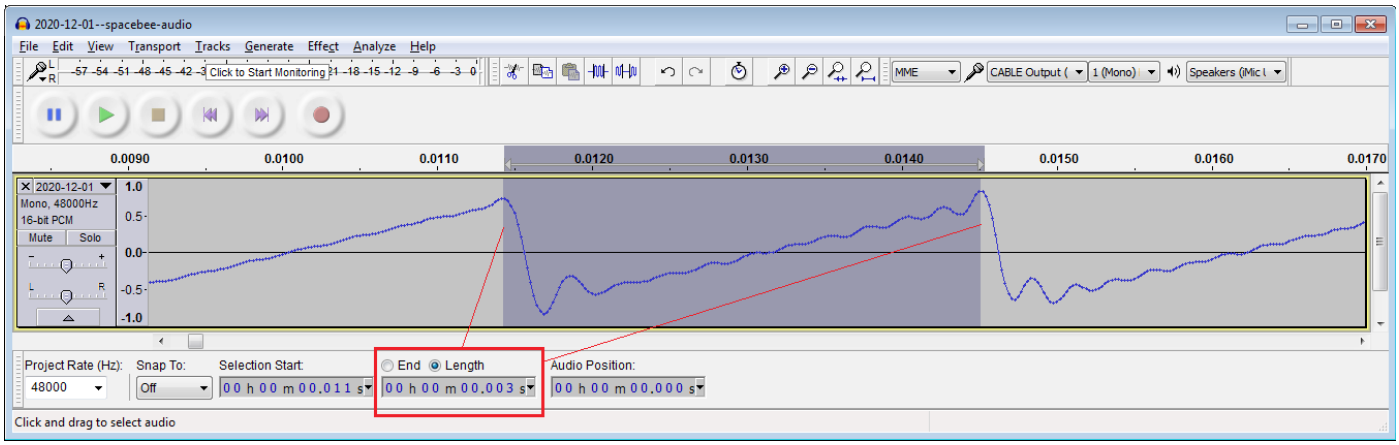


Using this website, we can benefit from the fact that there are relationships between some of the LoRa parameters. Change one and it affects one or more of the others. We have the value for Bandwidth (41.7) already, so let's fill that in.



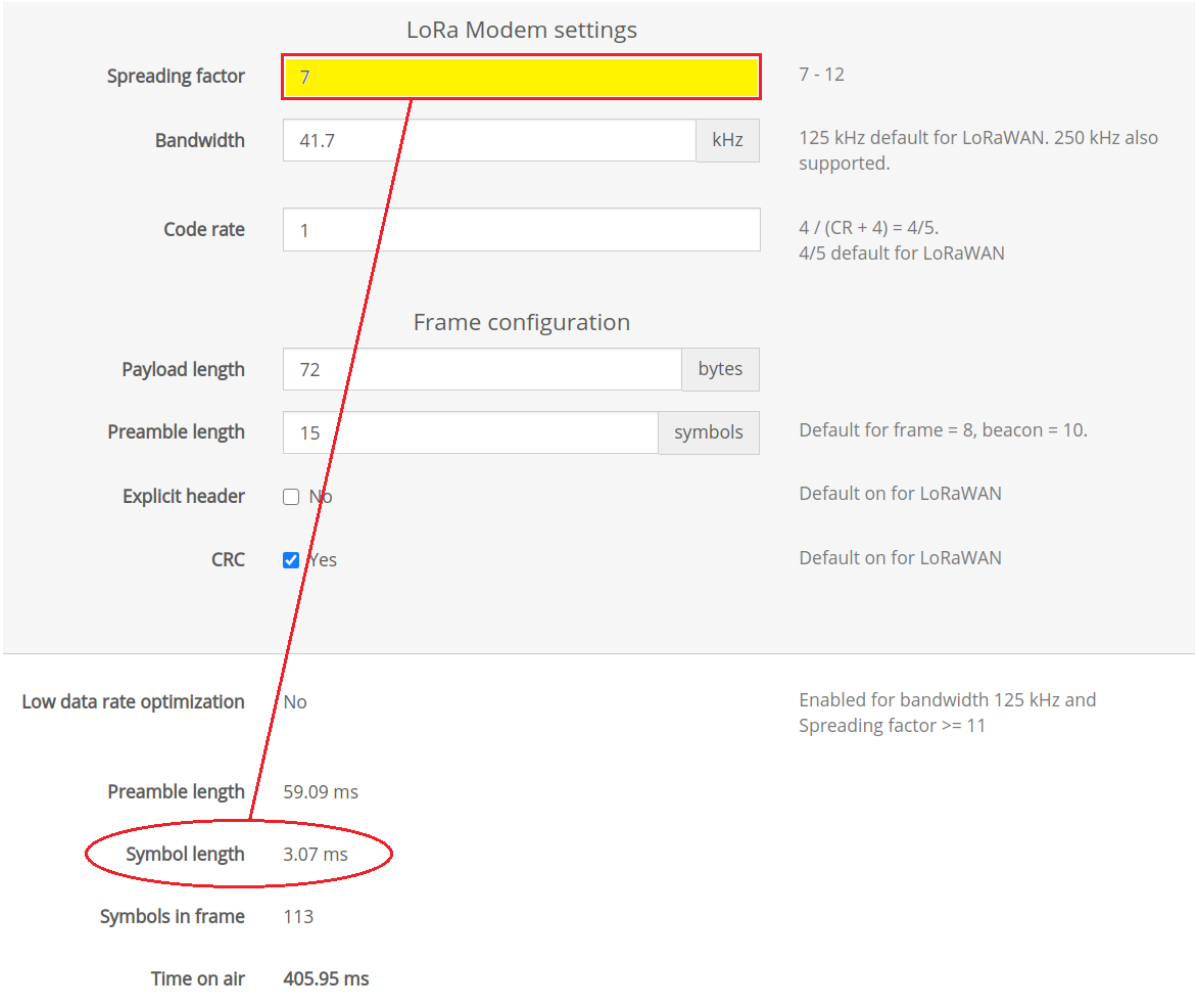
To determine the value of the other settings, we have to get creative. There is a relationship between "Spreading Factor" and the length of each symbol. So what is the width (time duration) of our symbols? Back to Audacity with that FM demod.

Zoom way in on a couple of clean up-chirps in the preamble and it looks like the length of each symbol is 3ms.



Back on the LoRa calculator website, it's a simple matter to step through the possible values of "Spreading Factor" and watch the corresponding "Symbol Length" for each one.

With a **"Spreading Factor" of "7"**, we find a match for our mystery signal's "Symbol Length". One more LoRa parameter that can be checked off.



Next on the list is the parameter “Code Rate”. There are only 4 options: 5, 6, 7, & 8.

Changes to the “Code Rate” parameter affect several things, including a packet’s on-air time, how many total symbols you end up with in the packet, and the effective Data Rate. Let’s use a couple of those to make our best estimate of the correct value for “Code Rate”.

There is another handy LoRa calculator at the following website:

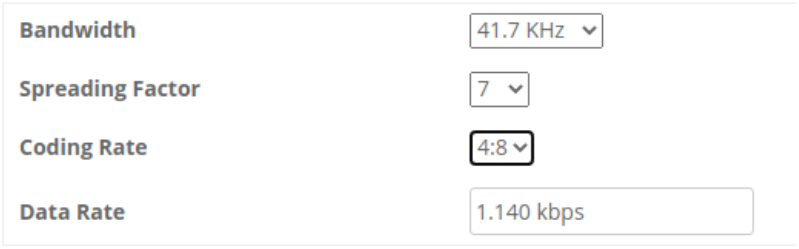
https://unsigned.io/understanding-lora-parameters/

… that allow us to see how changes to the key paramaters impact the effective data rate.

Knowing that our bandwidth is 41.7 & SF=7, it’s easy to check all 4 values for Code Rate to see what the resulting Data Rate would be. Swarm (the company behind the SpaceBee constellation) advertises a data rate of “1 kbps”.

BIT RATE	1 kbps
FREQUENCY	137-138 MHz (downlink) 148-150 MHz (uplink)

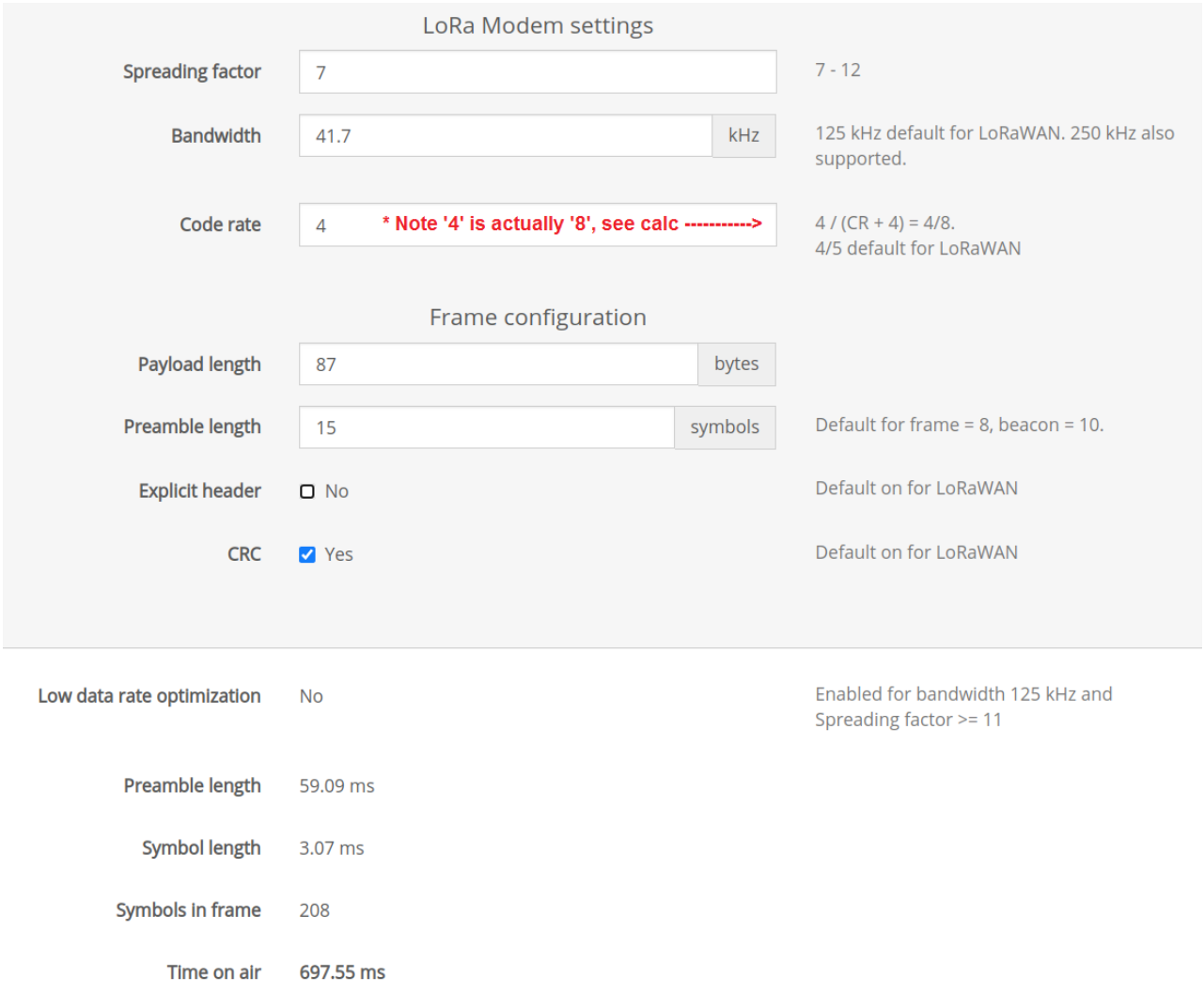
… and the Coding Rate that produces the value closest to that 1k is “8” (often listed as “4/8” - as in 4/5, 4/6, 4/7, or 4/8).



It’s worth noting that the highest Coding Rate (8) gives you the most reliable communications integrity. So, it’s a good choice for a satellite service.

I’ll also mention that CR=8 works out extremely well when you do the math on total packet duration (~600+ ms) & number of symbols (~200+) in my SpaceBee packet sample. This explanation is long enough already without documenting all that, though.

So that fills in most of the fields on our original calculator website:



I am assuming “yes” for CRC. But as usual the real test is whether this dog will hunt. Do we get decodes?

After applying these settings to my LoRa-equipped Arduino, on the very next SpaceBee pass I did indeed get decodes!

arduino-decodes.txt

```
[03:26:42:144] Listening for downlink on 137.975 ......
[03:26:42:175]
[03:26:42:175] BW=41.7    SF=7    CR=8
[03:26:42:206] Preamble_Len=15  SyncWord=0x12
[03:26:42:242] -----------------------------------
[03:26:42:278]
[03:31:20:895] Got CRC error!
[03:31:20:912] Received 87 bytes:
[03:31:20:940] 0x0  <break>
[03:31:20:940]
[03:31:20:942] 0x0  <break>
[03:31:20:945]
[03:31:20:945] 0xAC <0xac>
[03:31:20:971] 0xDE <0xde>
[03:31:20:971] 0xA7 <0xa7>
[03:31:20:971] 0xFF <0xff>
[03:31:20:977] 0x37 7
[03:31:21:005] 0x7A z
[03:31:21:005] 0x2F /
[03:31:21:007] 0xBE <0xbe>
[03:31:21:041] 0x67 g
[03:31:21:041] 0xFC <0xfc>
[03:31:21:041] 0xEF <0xef>
[03:31:21:041] 0xFF <0xff>
[03:31:21:047] 0x75 u
[03:31:21:072] 0xB3 <0xb3>
[03:31:21:072] 0x13 <0x13>
[03:31:21:072] 0xA7 <0xa7>
[03:31:21:081] 0x66 f
[03:31:21:108] 0xEE <0xee>
[03:31:21:108] 0xFB <0xfb>
[03:31:21:108] 0x7F <0x7f>
[03:31:21:117] 0x75 u
[03:31:21:121] 0x9F <0x9f>
[03:31:21:149] 0x7F <0x7f>
[03:31:21:149] 0x43 C
[03:31:21:149] 0xA8 <0xa8>
```

… why the CRC error? Don’t know. Is the decode valid? Don’t know. Are all my parameters correct? Probably not.

But it’s a start and I was getting NO decodes with other settings. So, progress is progress.

I hope some of the methods used in this search might be helpful to others.

-Scott, K4KDR
 December 2020

Footnote: There is always “more to the story”. Some months ago when this last bunch of SpaceBees were deployed, I actually got some LoRa decodes using DIFFERENT settings. The fact that my old settings no longer worked is what prompted this search for the current parameters.