Advanced Computer Systems

Opening parking lot gate with CC1350

Yahav Avigal
Assaf Avital

Fall 2017

Introduction

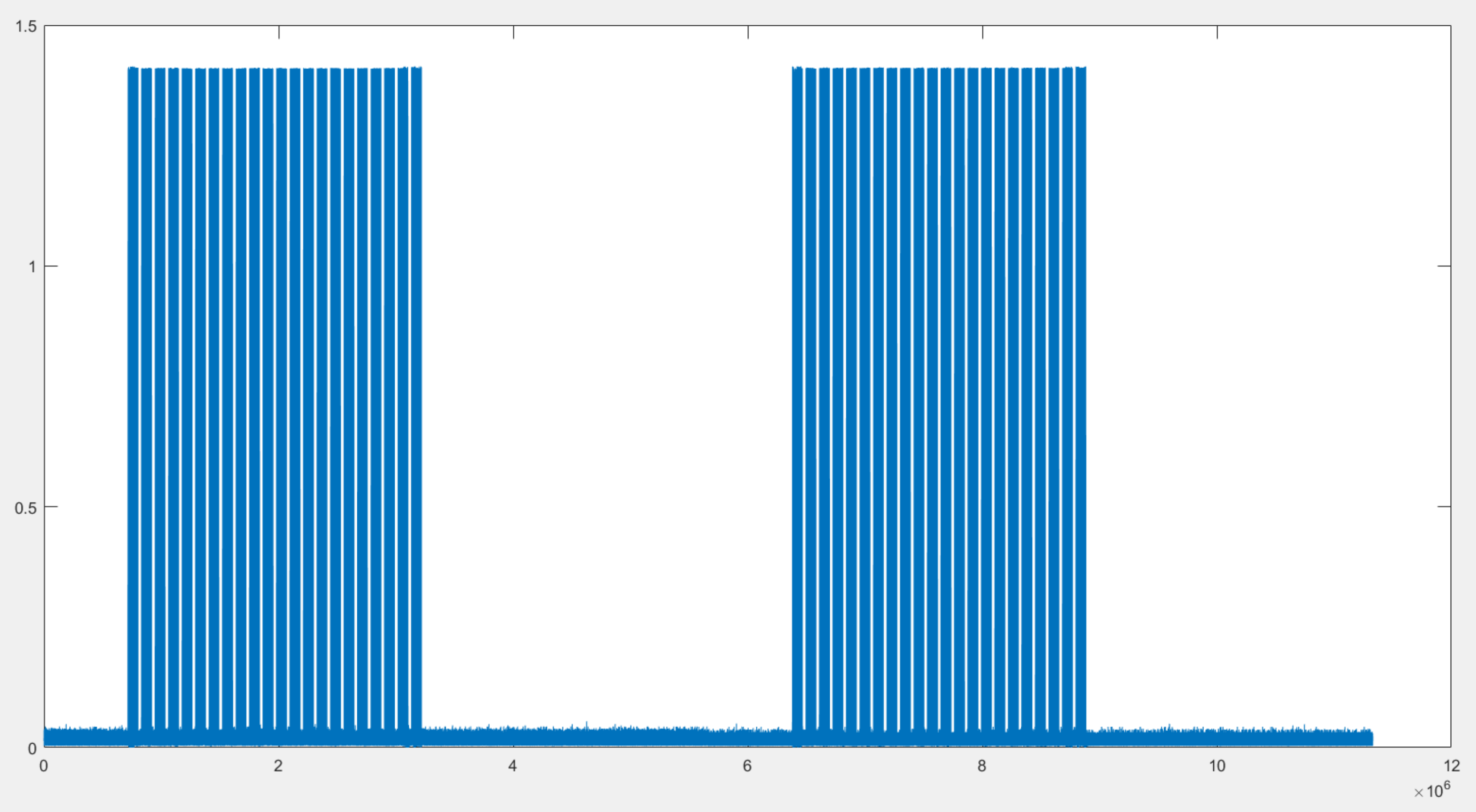
- During the Advanced Computer Systems course we have focused on embedded systems, and demonstrated the concepts and protocols on the CC1350 Launchpad by Texas Instruments.
- In the final project we decided to hack a parking lot gate by duplicating the original remote control's signal, and transmitting it from the CC1350.
- Hardware –
  - CC1350
  - RTL Dongle
- Software –
  - SDR# - signal recording
  - MATLAB – signal processing
  - SmartRF – signal transmitting
  - CCS – the CC1350 IDE

Stage 1 - Decoding raw signal

- We used SDR# on RAW mode to record the remote’s transmission at 433.91MHz

- MATLAB’s audioread() allows us to plot the sampling data as well as getting parameters such as the sampling rate (which will be used later on)

- Each blue “narrow” column is a packet, and here we can see 2 following transmissions of 22 packets. The number of packets in a transmission depends on how long the remote control’s button was pressed

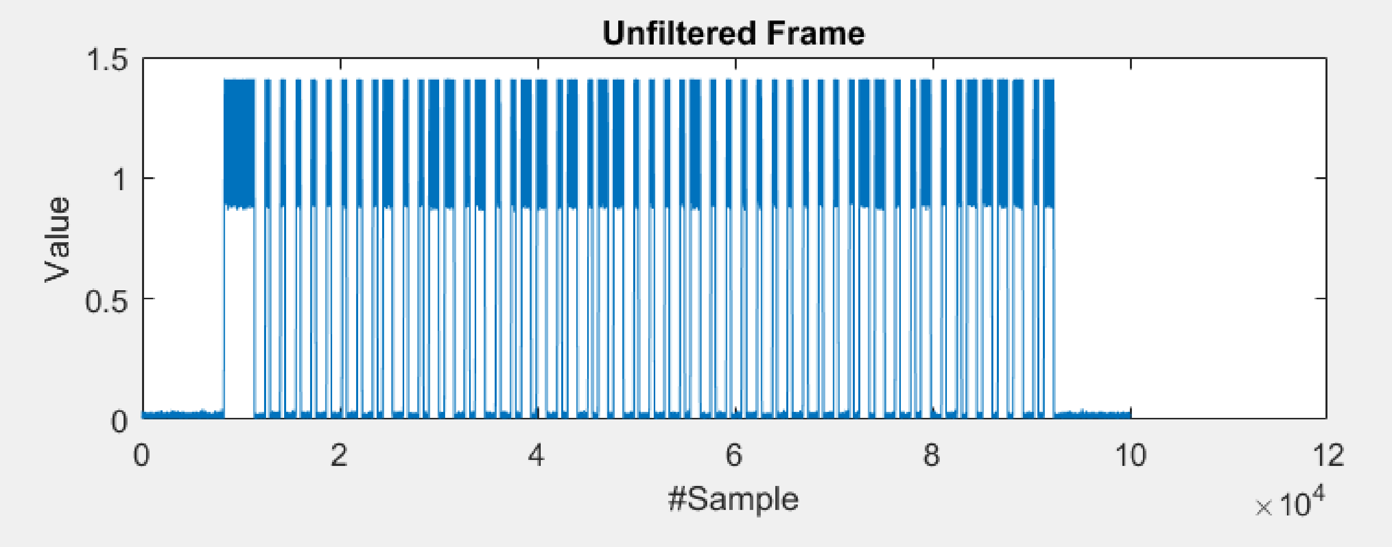


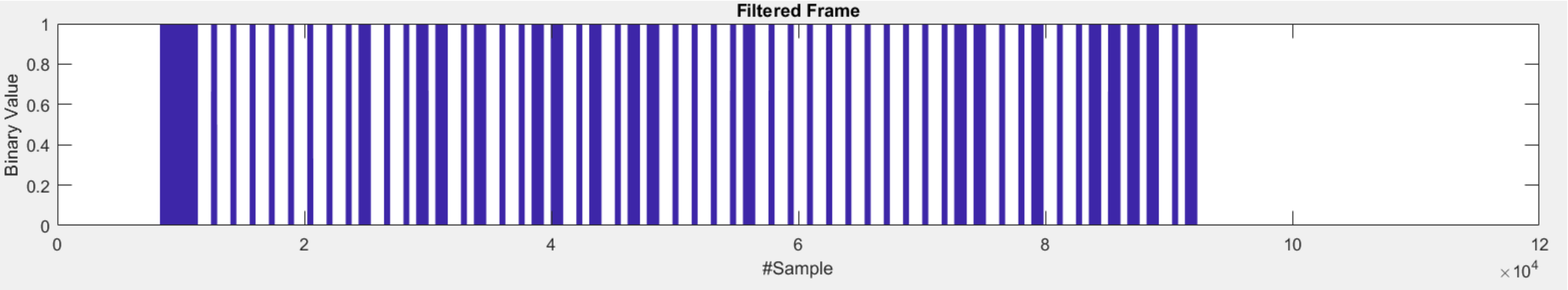
Decoding raw signal

- We arbitrarily choose one repeating packet from the entire signal to analyze.

- We apply threshold filtering to turn that frame into a logic-value array.

- Since it is an ‘OOK’ modulation, the blue columns represent ‘1’s and the white columns ‘0’s. Nevertheless, the bit lengths aren’t perfect multiplications of the shortest bit frame.





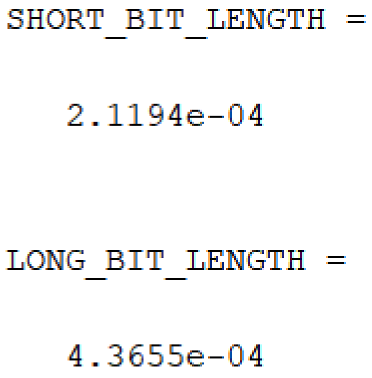
Decoding raw signal

- With the logic-value array in hand, we can try to fit a normal distribution over the width of each “bit”. The “short_bit” represents a single bit, while the “long_bit” represents a doubled bit

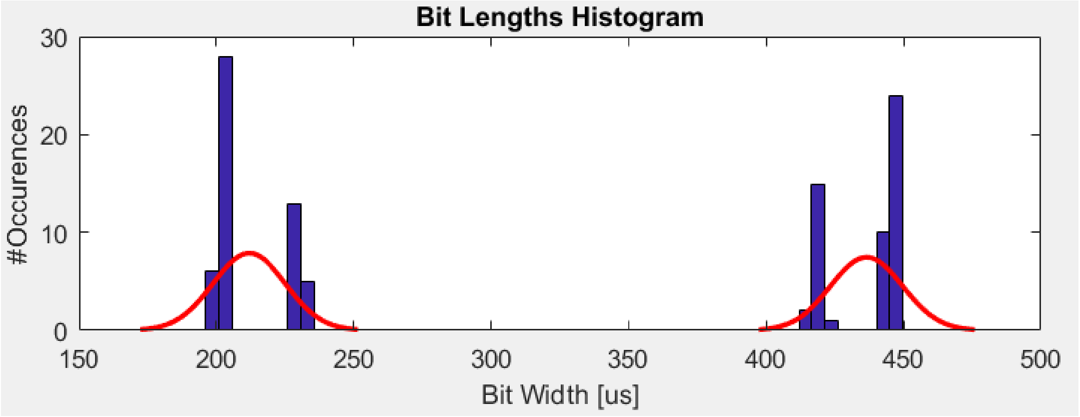
- Fortunately enough, the bits in the signal seem to satisfy:

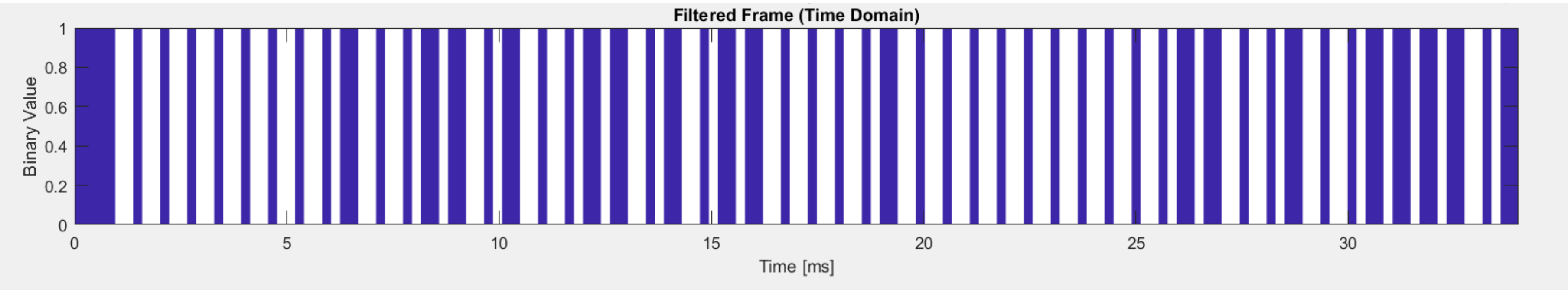
  $\mu_{wide} \approx 2\mu_{narrow}$
  $preamble \approx 5\mu_{narrow}$

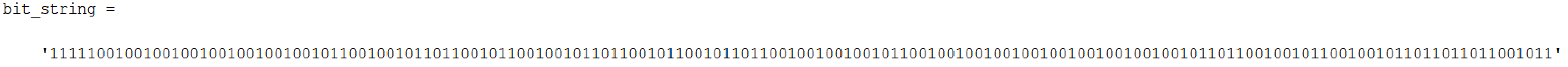
- Assuming a negligible σ^2, we can use μ to reconstruct the digital packet.



```
SHORT_BIT_LENGTH =

    2.1194e-04


LONG_BIT_LENGTH =

    4.3655e-04
```





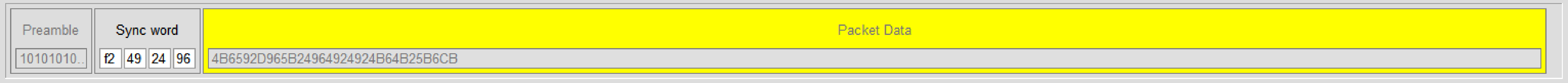


```
bit_string =

    '1111100100100100100100100100101100100101101100101100100101101100101100101101100100100101100100100100100100100100100100101101100100101100100101101101100101 1'
```

Stage 2 – Transmission from CC1350

- The binary string is the one we aspire to transmit using the CC1350.
- This string consists of a preamble, a sync word, and data (we ignore CRC).
- Arbitrarily choosing a preamble of 1 bit and a sync word of 4 bytes, we can split the packet in order for it to fit the SmartRF configurations.

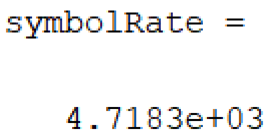
```
preamble =

    '1'

sync_word =

    'F2492496'

hex_string =

    '4B6592D965B24964924924B64B25B6CB'
```



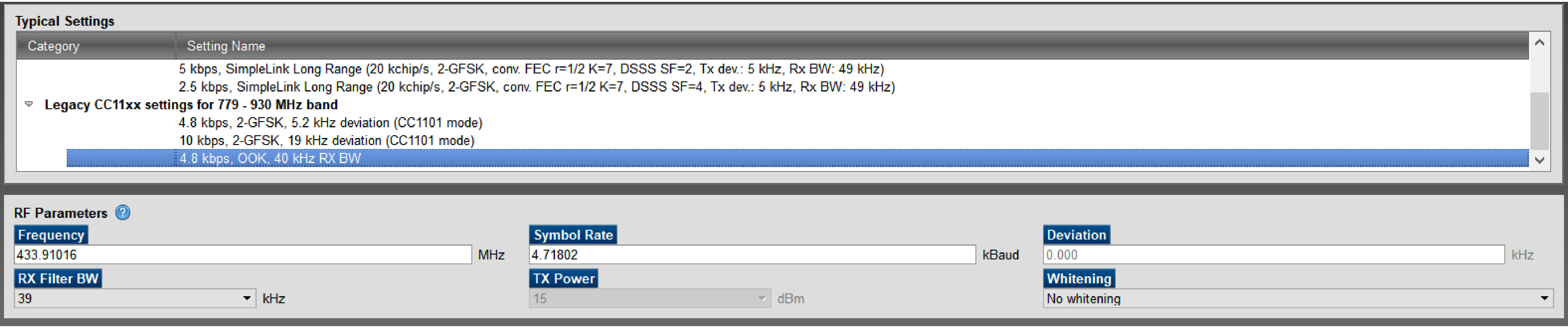
Preamble	Sync word	Packet Data
10101010..	f2 49 24 96	4B6592D965B24964924924B64B25B6CB

Stage 2 – Transmission from CC1350

- We calculate $\frac{1}{\mu_{narrow}}$ to get the symbol rate (measured in Bauds).
- We use the OOK legacy setting, with no whitening.



```
symbolRate =

   4.7183e+03
```



Typical Settings

Category | Setting Name

5 kbps, SimpleLink Long Range (20 kchip/s, 2-GFSK, conv. FEC r=1/2 K=7, DSSS SF=2, Tx dev.: 5 kHz, Rx BW: 49 kHz)

2.5 kbps, SimpleLink Long Range (20 kchip/s, 2-GFSK, conv. FEC r=1/2 K=7, DSSS SF=4, Tx dev.: 5 kHz, Rx BW: 49 kHz)

Legacy CC11xx settings for 779 - 930 MHz band

4.8 kbps, 2-GFSK, 5.2 kHz deviation (CC1101 mode)

10 kbps, 2-GFSK, 19 kHz deviation (CC1101 mode)

4.8 kbps, OOK, 40 kHz RX BW

RF Parameters

Frequency: 433.91016 MHz

Symbol Rate: 4.71802 kBaud

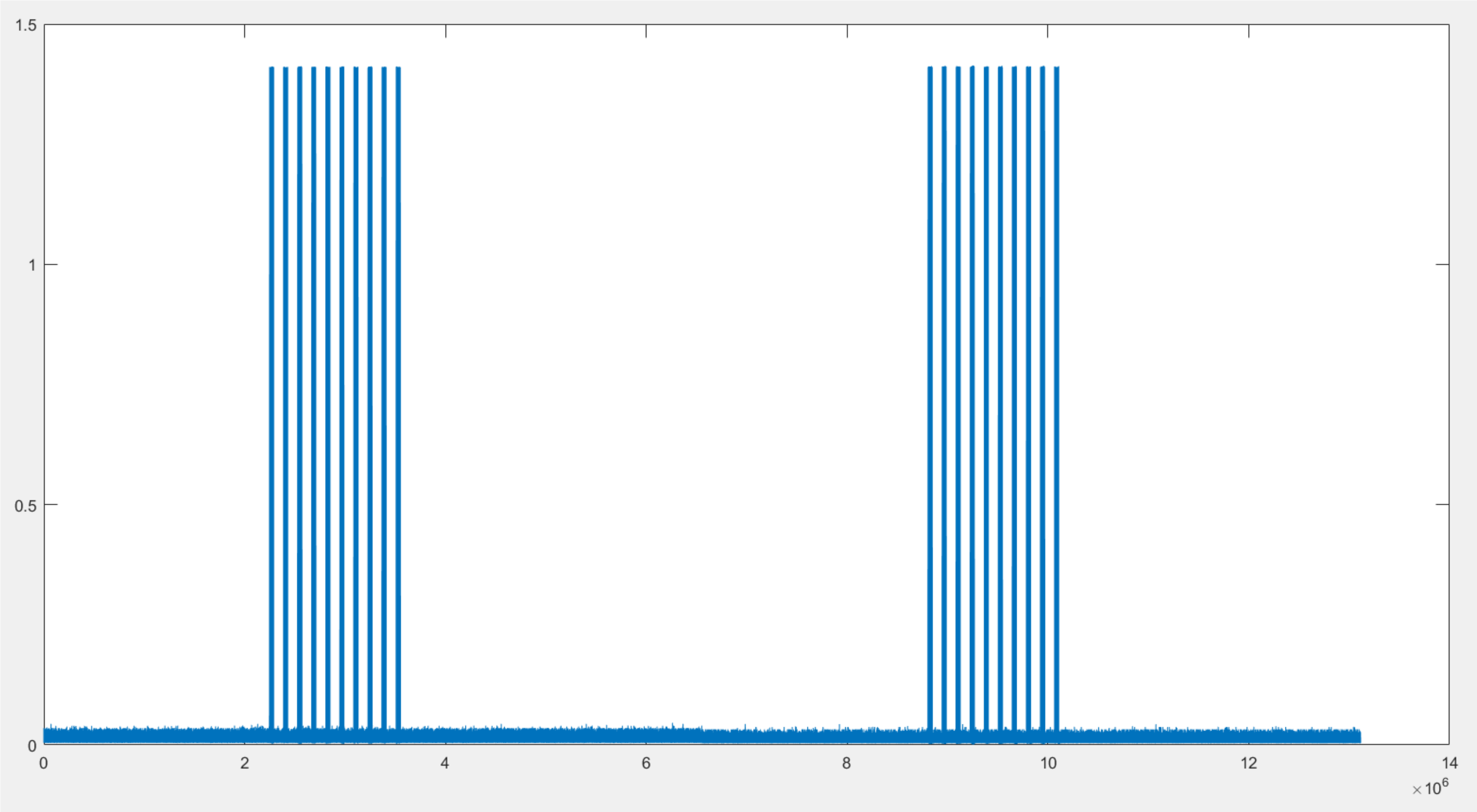
Deviation: 0.000 kHz

RX Filter BW: 39 kHz

TX Power: 15 dBm

Whitening: No whitening

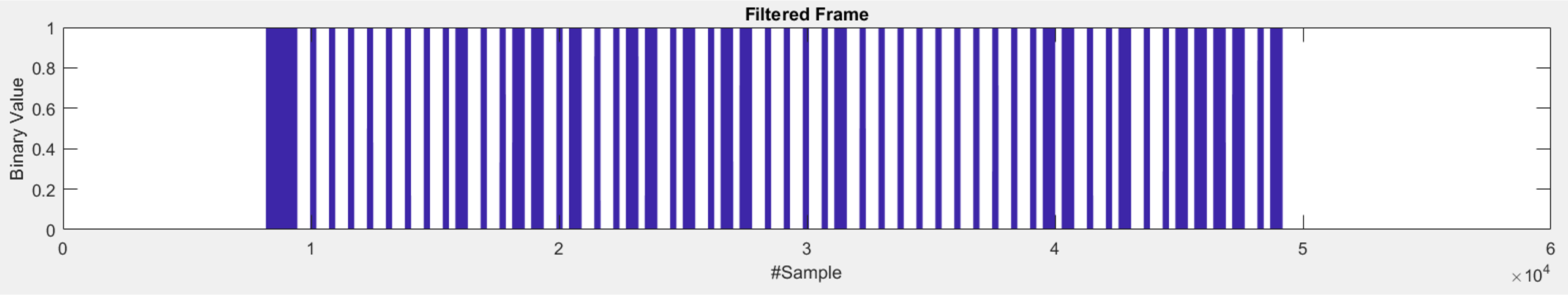
Analyzing the transmission – attempt #1



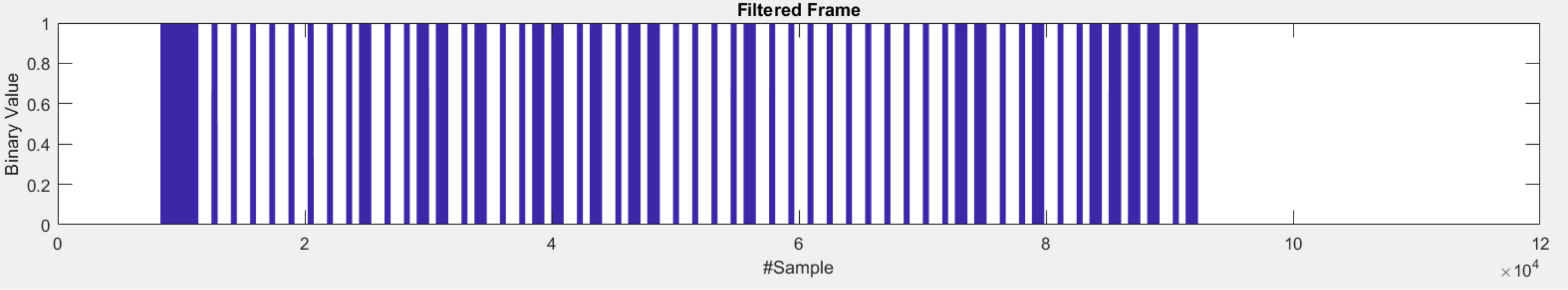
Analyzing the transmission – attempt #1

- First glance, after applying filter we find the packet to perfectly match the remote control TX (logic-wise).

CC1350 TX:



Remote control TX:

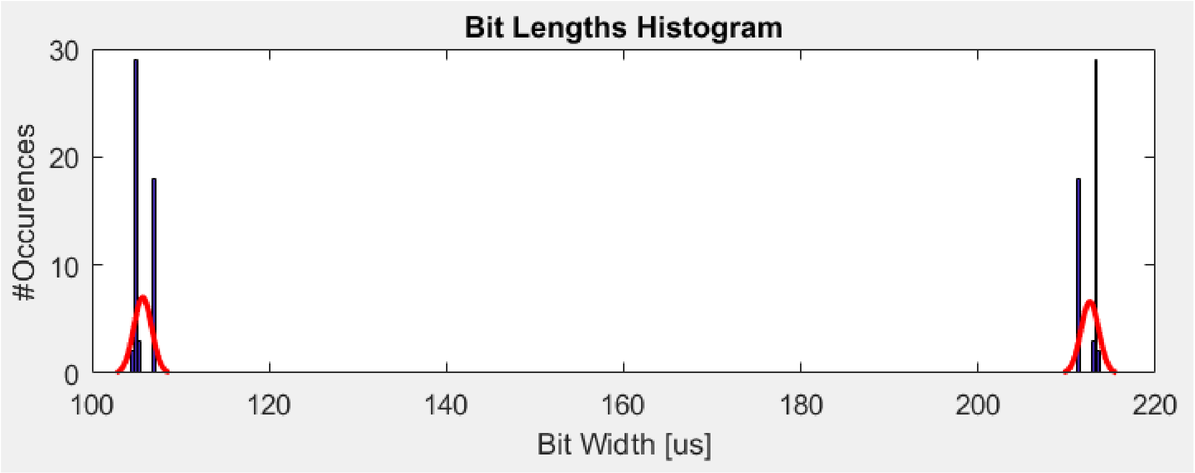


Analyzing the transmission – attempt #1

- Fitting the normal distribution, we find that the value of μ_{narrow} was not as expected, but in fact **half** of it.

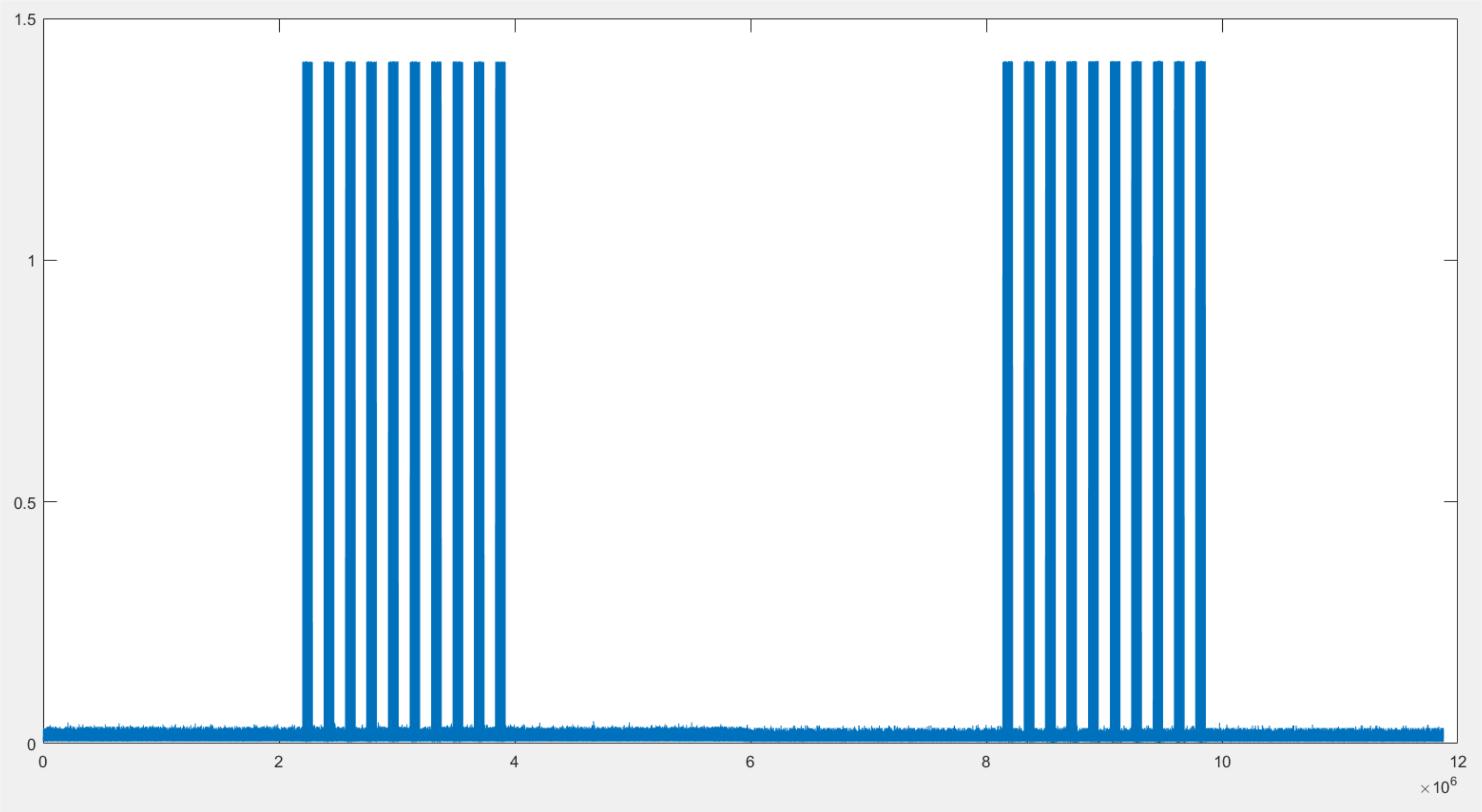
- Since $\mu_{wide} \approx 2\mu_{narrow}$ still holds, we will now try to retransmit the packet with each bit **doubled** in length.

- Why not simply setting $\mathrm{symbolRate}' = 0.5 \cdot \mathrm{symbolRate}$? We actually noticed that using a low value for symbolRate results in terrible transmissions, with virtually no influence on µ



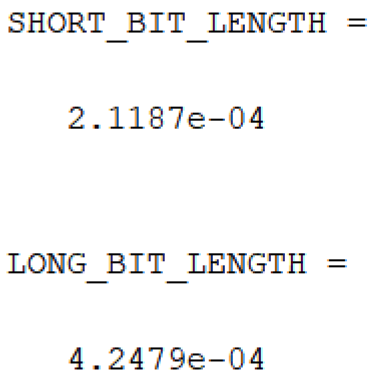
```
SHORT_BIT_LENGTH =

   1.0573e-04


LONG_BIT_LENGTH =

   2.1260e-04
```

Analyzing the transmission – attempt #2

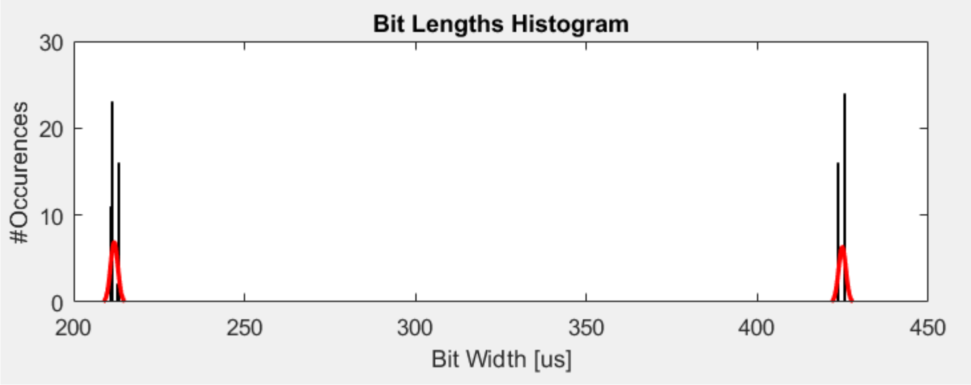


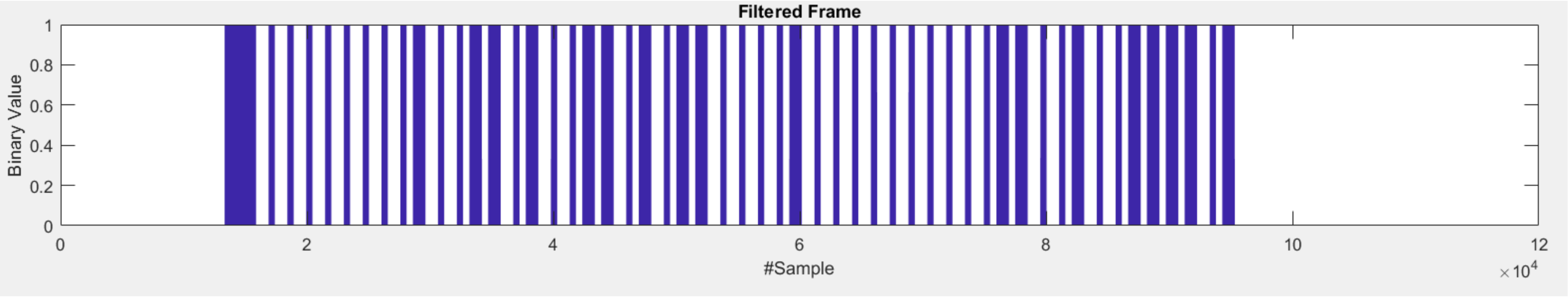
Analyzing the transmission – attempt #2

- We've successfully reached a close enough value for μ, without damaging the contents of the packet.



```
SHORT_BIT_LENGTH =

   2.1187e-04


LONG_BIT_LENGTH =

   4.2479e-04
```



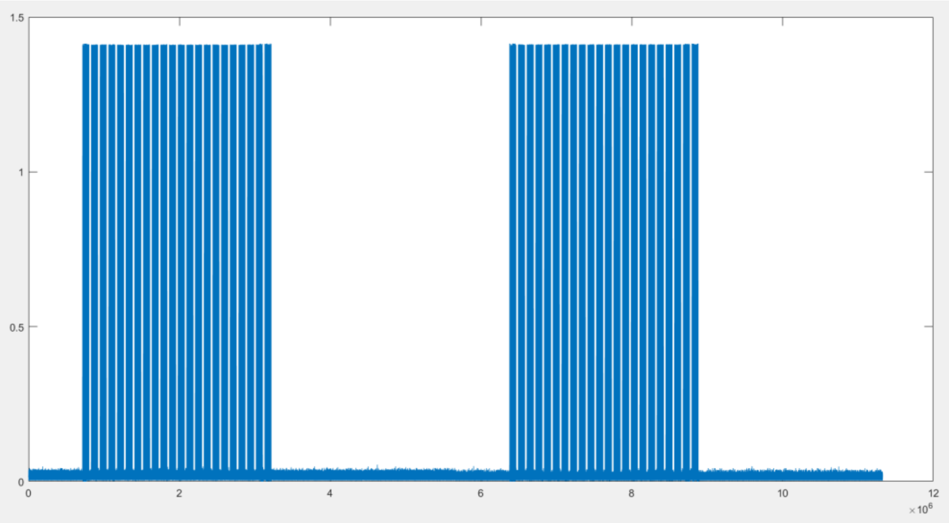


Analyzing the retransmission – attempt #2

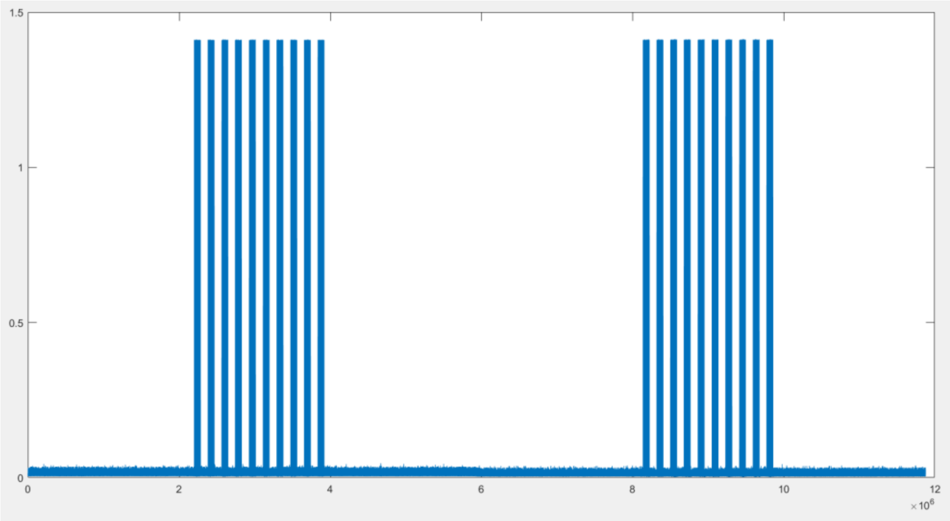
- However, this TX was yet to successfully open our parking lot gate.

- Looking at the raw packets, we identify a possible cause – the pause length between each individual frame. The original (remote control) TX shows a tiny pause, while the CC1350 transmits frames which are by-far sparser.

- We again fit a normal distribution to the pause lengths, resulting in $\mu_{pause} \approx 13ms$. This value can be used in SmartRF to achieve an improved TX.



Original TX



CC1350 TX

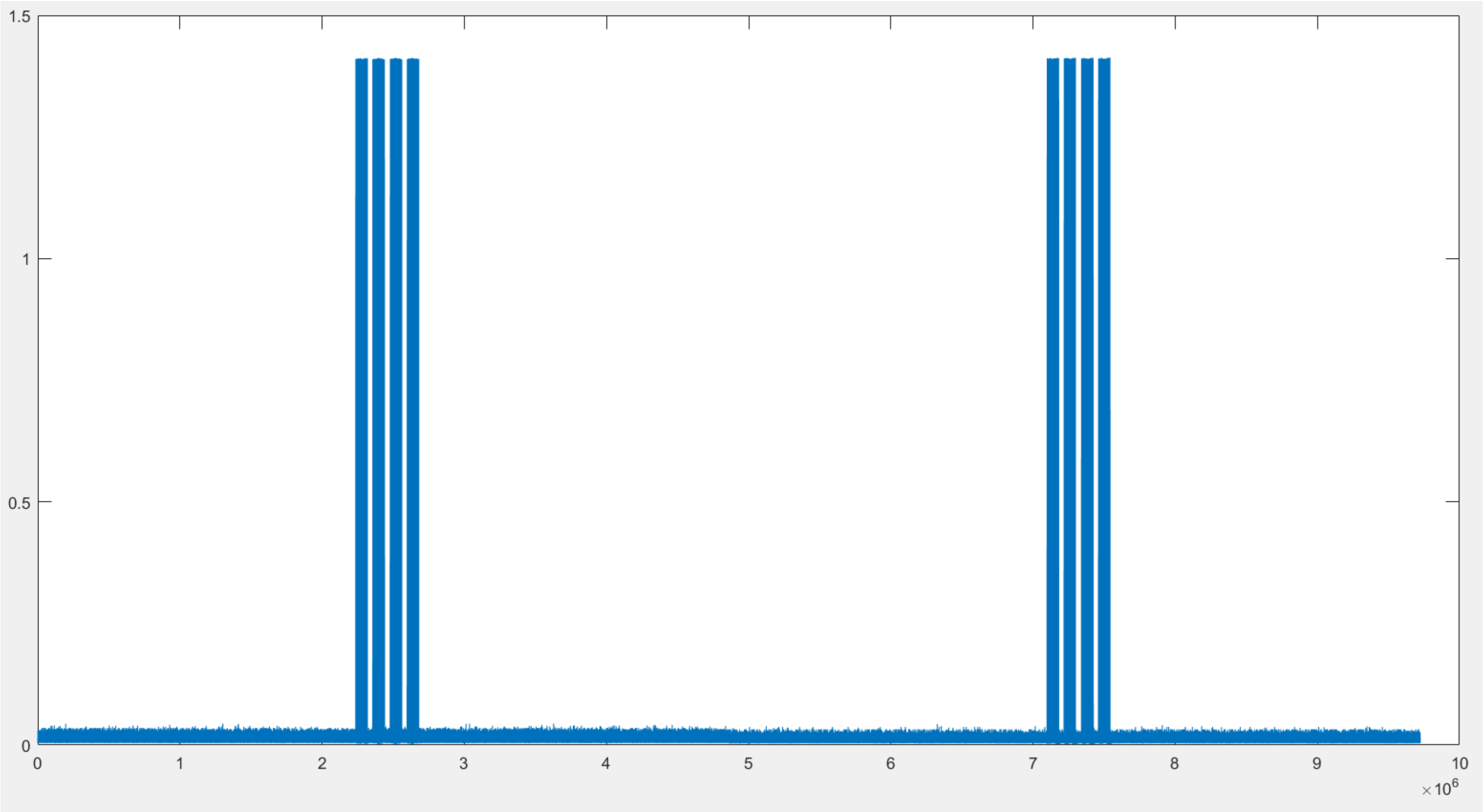
Analyzing the transmission – attempt #3

- Time between individual frames is now as expected.

- Values for μ have not changed.

- Logic values were not damaged.

But apparently –
that's not enough for our gate. ☹



Transmission configuration

At this points we decided to focus on our SmartRF TX configuration instead.

We tried different configurations for:

- Symbol rate
- Frequency
- Infinite packet count
- Whitening
- Different modulations (OOK, FSK, GFSK)
- Different parking lot gates + a 12V gate-simulating µC (sponsored by our gate-opening rival team)

But still, the gate remains closed.

Conclusions

To sum it up, we are left with some tough questions –

- Are we leaving out any important parameters?
- Are we bounded by the specifications of the CC1350 or the limitations of the human brain?
- Is our project feasible considering modern gate-opening protocols?

For now, those questions remain open, unlike the gate…