

Analyze Märklin track signal

Get complementary signals from digitizing car

Load file

```
In[1]:= Directory[]  
Out[1]= /home/cmaier/scad/Märklin/analysis  
  
In[2]:= Dimensions[raw = Import["TTL.10ms.5V.csv"]]  
Out[2]= {524 290, 4}  
  
In[3]:= Take[raw, 12]  
Out[3]= {{X, CH1, CH2, }, {Second, Volt, Volt, }, {-0.131112, 5.4, 0., },  
{-0.131112, 5.4, 0., }, {-0.131111, 5.4, 0., }, {-0.131111, 5.4, 0., },  
{-0.131111, 5.4, 0., }, {-0.131111, 5.4, 0., }, {-0.131109, 5.4, 0., },  
{-0.131109, 5.4, 0., }, {-0.131108, 5.4, 0., }, {-0.131108, 5.4, 0., }}
```

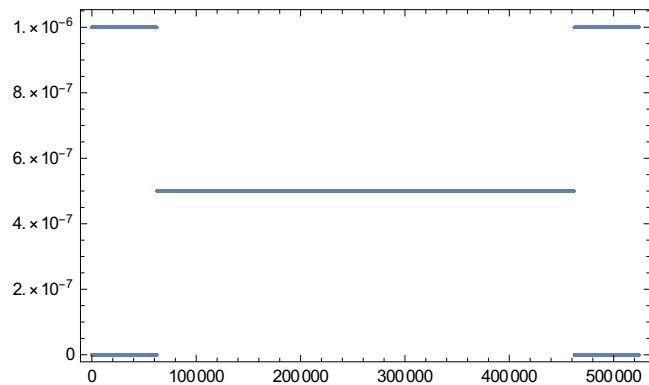
Convert into two time series

```
In[4]:= Dimensions[  
timeseries = Transpose[{{#[[1]], #[[2]]}, {#[[1]], #[[3]]}} & /@ Drop[raw, 2], {2, 1, 3}]]  
Out[4]= {2, 524 288, 2}
```

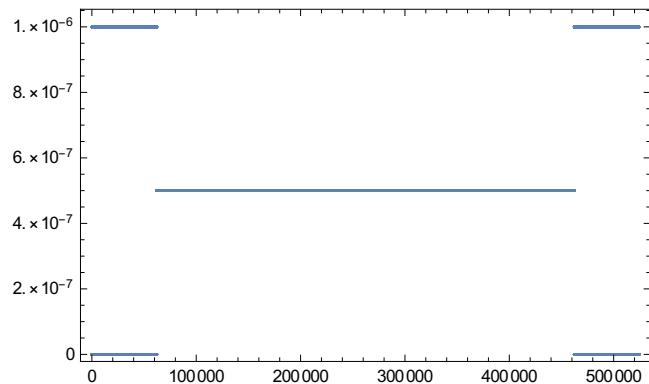
Calculate time step

```
In[5]:= Dimensions[ListConvolve[{1, -1}, First /@ #] & /@ timeseries]  
Out[5]= {2, 524 287}
```

```
In[6]:= GraphicsGrid[
{ListPlot[ListConvolve[{1, -1}, First /@ #], Frame → True, PlotRange → All] } & /@
timeseries]
```



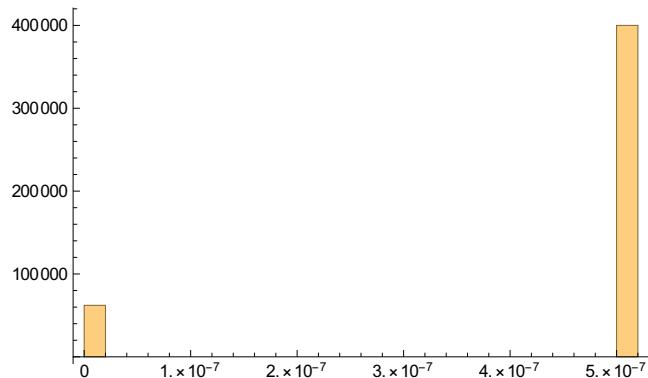
Out[6]=



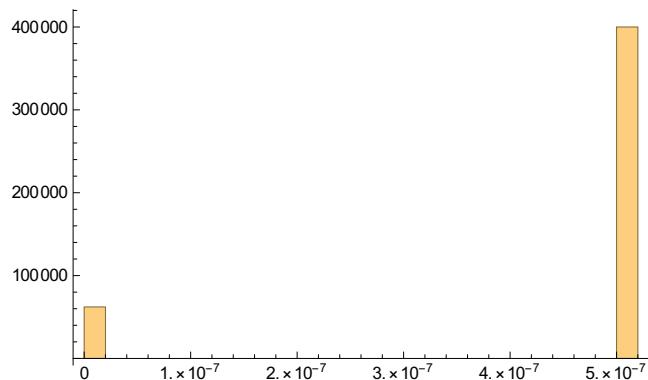
```
In[7]:= {Min[#], Max[#]} &[ListConvolve[{1, -1}, First /@ #]] & /@ timeseries
Out[7]= {{0., 1. × 10-6}, {0., 1. × 10-6}}
```

The time stamps have a quantization error because the precision at the beginning and end is only 1μs.

```
In[8]:= GraphicsGrid[
{Histogram[ListConvolve[{1, -1}, First /@ #, PlotRange -> All]]} & /@ timeseries]
```



Out[8]=

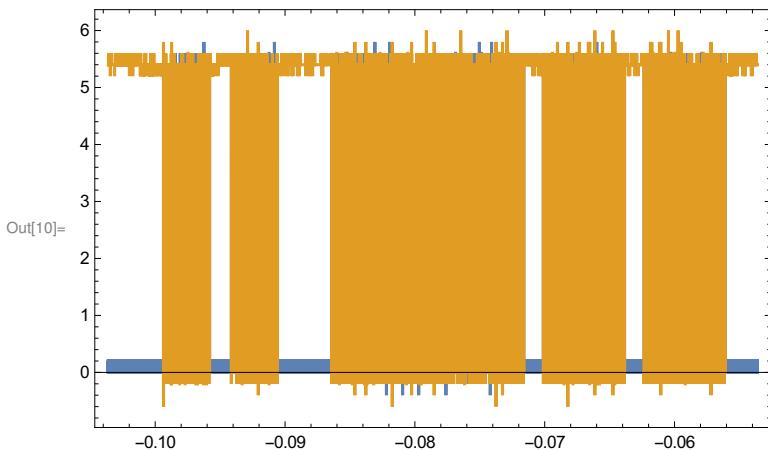


```
In[9]:= timestep = Mean[Median[ListConvolve[{1, -1}, First /@ #]]] & /@ timeseries]
```

Out[9]= $5. \times 10^{-7}$

Display

```
In[10]:= ListPlot[Take[#, {55 000, 155 000}] & /@ timeseries, Frame -> True, Joined -> True]
```

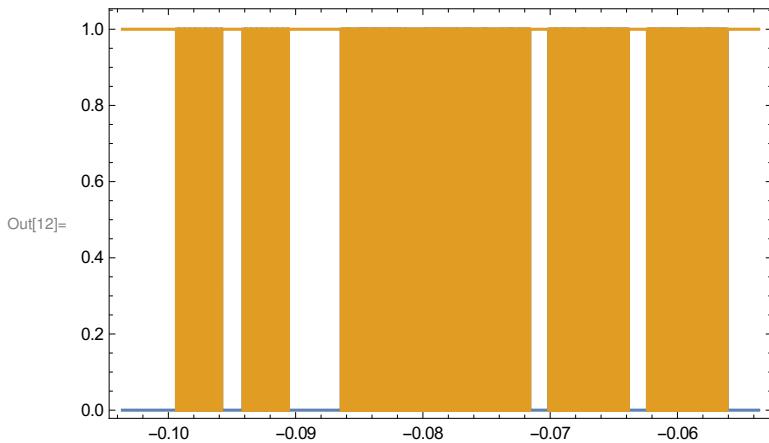


Digitize time series

```
In[11]:= Dimensions[digitized = Map[{#[[1]], HeavisideTheta[#[[2]] - 3.0001]} &, timeseries, {2}]]  
Out[11]= {2, 524 288, 2}
```

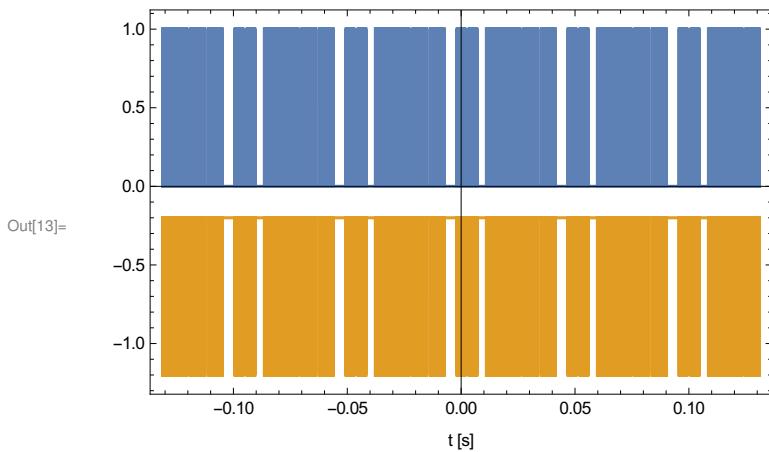
Display

```
In[12]:= ListPlot[Take[#, {55 000, 155 000}] & /@ digitized, Frame → True, Joined → True]
```



Display complementary phases

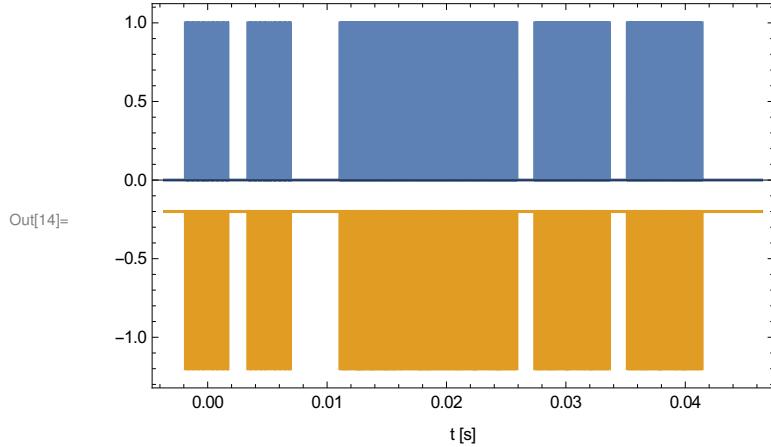
```
In[13]:= ListPlot[MapThread[Function[{series, offset}, # + {0, offset} & /@ series],  
{digitized, {0, -1.2}}], Frame → True,  
Joined → True, PlotRange → All, FrameLabel → {"t [s]", ""}]
```



The data sent repeats every 50ms

Display one 50ms data transmission

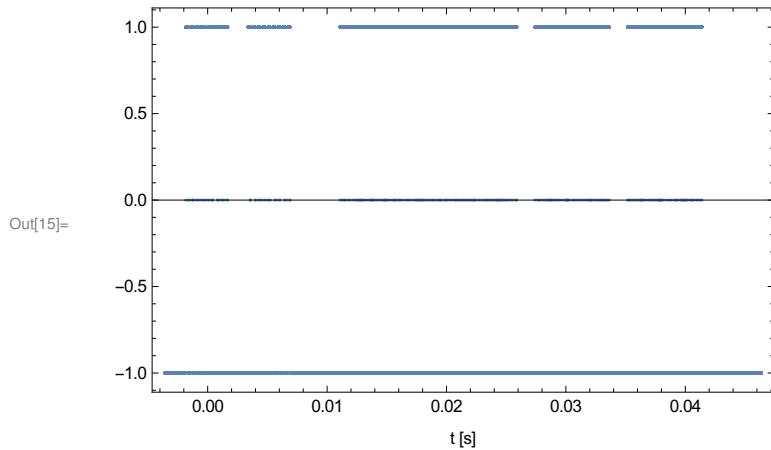
```
In[14]:= ListPlot[Take[#, {255 000, 355 000}] & /@ MapThread[
  Function[{series, offset}, # + {0, offset} & /@ series], {digitized, {0, -1.2}}],
  Frame → True, Joined → True, PlotRange → All, FrameLabel → {"t [s]", ""}]
```



Difference of time series

The zeros show where the phases do not overlap

```
In[15]:= ListPlot[
  Take[MapThread[{\frac{#1[[1]] + #2[[1]]}{2}, #1[[2]] - #2[[2]]} &, digitized], {255 000, 355 000}],
  Frame → True, Joined → False, PlotRange → All, FrameLabel → {"t [s]", ""}]
```



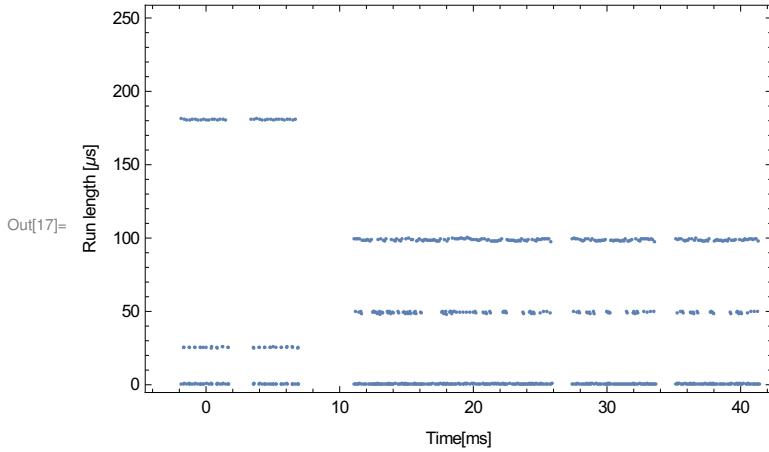
Runs of identical values for difference series

```
In[16]:= Dimensions /@
  (diffruns = SplitBy[Take[MapThread[{\frac{#1[[1]] + #2[[1]]}{2}, #1[[2]] - #2[[2]]} &, digitized],
  {255 000, 355 000}], Last])
```

```
Out[16]= {{3497, 2}, {1, 2}, {363, 2}, {1, 2}, {51, 2}, {51, 2}, {2, 2}, {362, 2}, {1, 2},
{361, 2}, {2, 2}, {51, 2}, {1, 2}, {51, 2}, {1, 2}, {361, 2}, {1, 2}, {362, 2},
{1, 2}, {51, 2}, {51, 2}, {2, 2}, {362, 2}, {1, 2}, {361, 2}, {1, 2}, {51, 2}, {1, 2},
```



```
In[17]:= ListPlot[{103 First[First[#]], 106 timestep Length[#]} & /@ diffruns,
  Frame → True, FrameLabel → {"Time[ms]", "Run length [μs]"}]
```



In the difference series, the short run lengths of the phase non-overlaps are visible.

Runs of identical values for each series


```
{197, 2}, {198, 2}, {196, 2}, {199, 2}, {197, 2}, {198, 2}, {199, 2}, {101, 2},  

{97, 2}, {199, 2}, {197, 2}, {198, 2}, {198, 2}, {102, 2}, {100, 2}, {101, 2},  

{97, 2}, {201, 2}, {99, 2}, {99, 2}, {197, 2}, {199, 2}, {199, 2}, {200, 2},  

{98, 2}, {201, 2}, {199, 2}, {101, 2}, {199, 2}, {201, 2}, {100, 2}, {200, 2},  

{199, 2}, {100, 2}, {195, 2}, {2994, 2}, {199, 2}, {101, 2}, {200, 2}, {201, 2},  

{99, 2}, {201, 2}, {100, 2}, {99, 2}, {198, 2}, {199, 2}, {196, 2}, {199, 2},  

{199, 2}, {102, 2}, {98, 2}, {201, 2}, {99, 2}, {100, 2}, {197, 2}, {201, 2},  

{100, 2}, {98, 2}, {197, 2}, {199, 2}, {196, 2}, {199, 2}, {196, 2}, {198, 2},  

{197, 2}, {199, 2}, {198, 2}, {102, 2}, {99, 2}, {102, 2}, {98, 2}, {102, 2},  

{98, 2}, {198, 2}, {197, 2}, {198, 2}, {197, 2}, {199, 2}, {197, 2}, {199, 2},  

{196, 2}, {198, 2}, {197, 2}, {198, 2}, {199, 2}, {101, 2}, {98, 2}, {198, 2},  

{196, 2}, {199, 2}, {198, 2}, {101, 2}, {100, 2}, {100, 2}, {97, 2}, {200, 2},  

{100, 2}, {99, 2}, {197, 2}, {199, 2}, {199, 2}, {201, 2}, {100, 2}, {201, 2},  

{199, 2}, {102, 2}, {199, 2}, {200, 2}, {100, 2}, {201, 2}, {198, 2}, {102, 2},  

{195, 2}, {2992, 2}, {199, 2}, {101, 2}, {198, 2}, {201, 2}, {100, 2}, {201, 2},  

{100, 2}, {99, 2}, {197, 2}, {199, 2}, {197, 2}, {199, 2}, {199, 2}, {102, 2},  

{98, 2}, {201, 2}, {99, 2}, {100, 2}, {197, 2}, {201, 2}, {100, 2}, {99, 2},  

{197, 2}, {199, 2}, {197, 2}, {198, 2}, {196, 2}, {199, 2}, {197, 2}, {198, 2},  

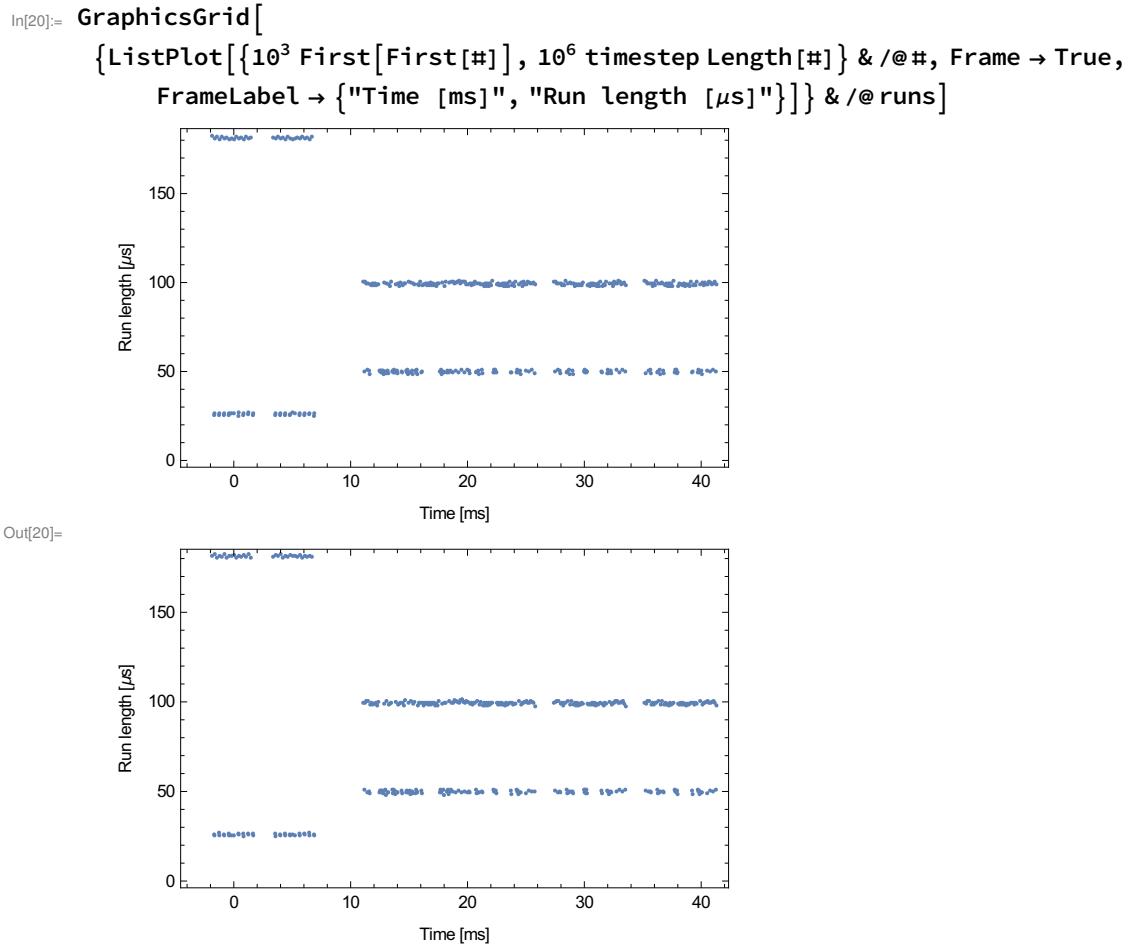
{199, 2}, {102, 2}, {100, 2}, {101, 2}, {99, 2}, {101, 2}, {97, 2}, {201, 2}, {99, 2},  

{99, 2}, {197, 2}, {199, 2}, {199, 2}, {200, 2}, {98, 2}, {201, 2}, {199, 2}, {101, 2},  

{199, 2}, {200, 2}, {100, 2}, {201, 2}, {199, 2}, {102, 2}, {196, 2}, {9961, 2}}}
```

In[19]:= **Length /@ runs**

Out[19]= {421, 421}



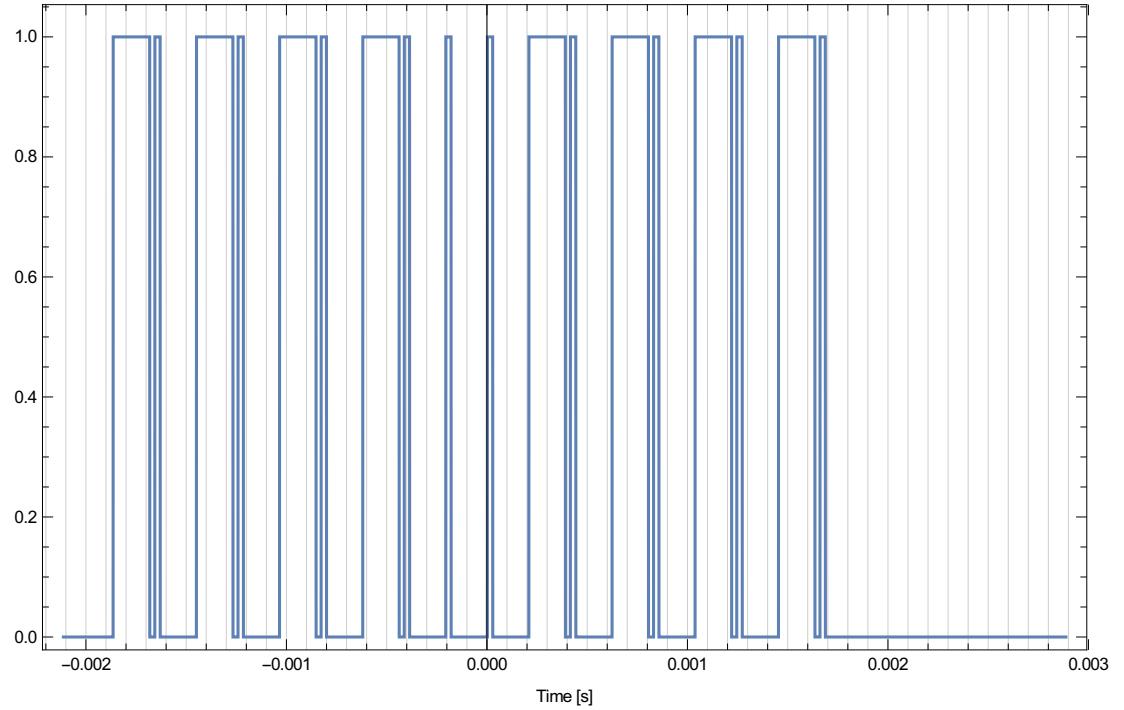
It is clearly apparent that the short modulation bursts have a different encoding timing than the remaining long and medium length bursts.

Packets within the time series

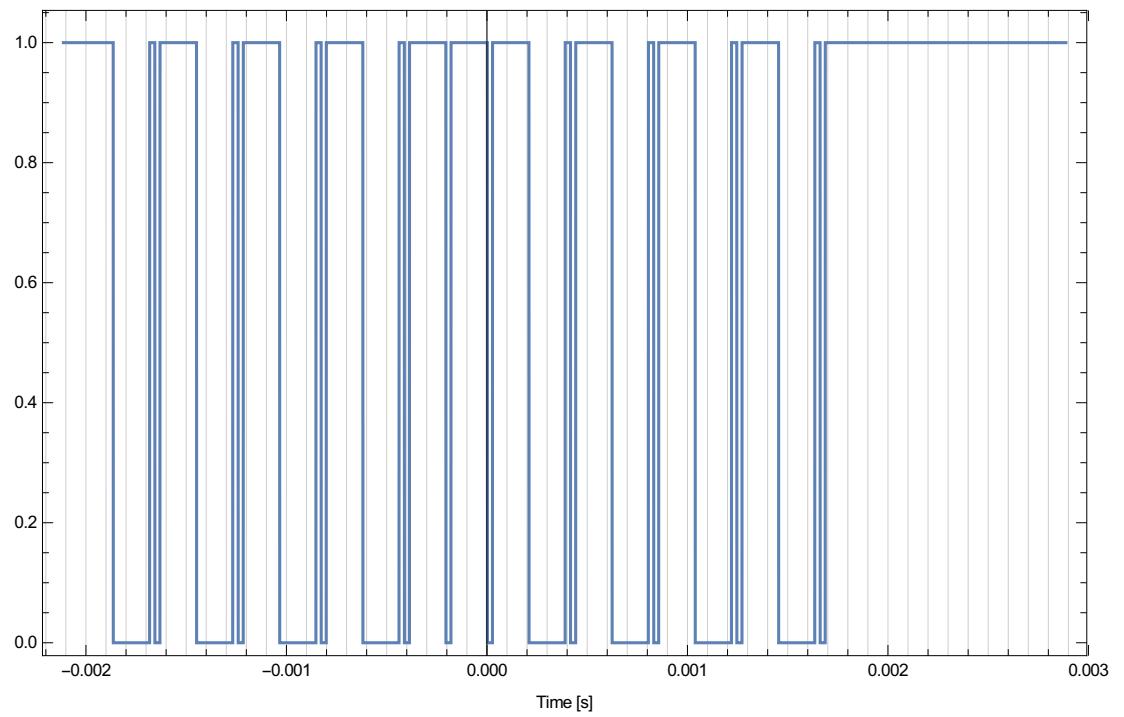
Short data packets

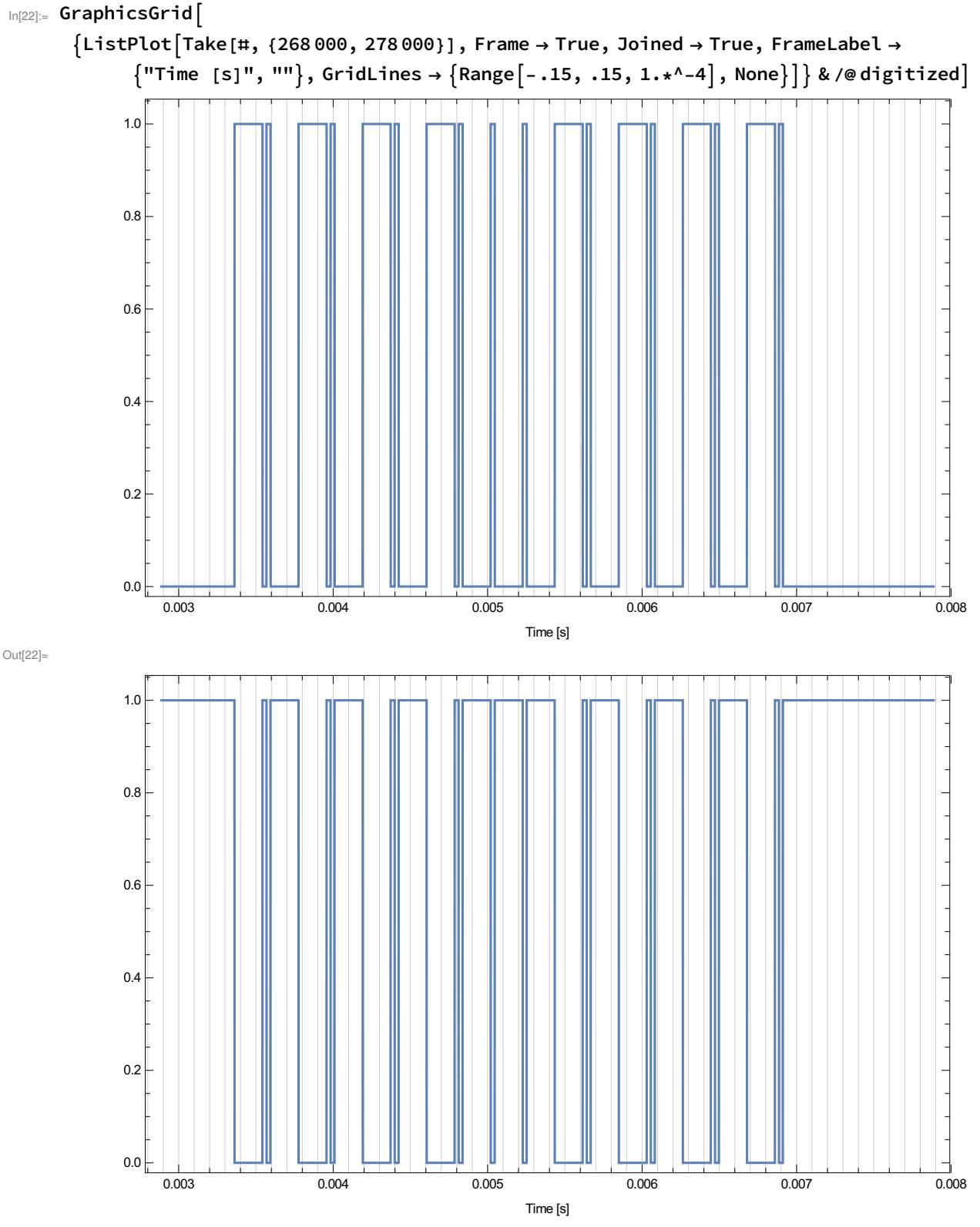
MM format, see 2.2.9 Einbettung von Steuerbefehlen im MM-Format

```
In[21]:= GraphicsGrid[
{ListPlot[Take[#, {258 000, 268 000}], Frame -> True, Joined -> True, FrameLabel ->
 {"Time [s]", ""}, GridLines -> {Range[-.15, .15, 1.*^-4], None}]} &/@digitized]
```



Out[21]=

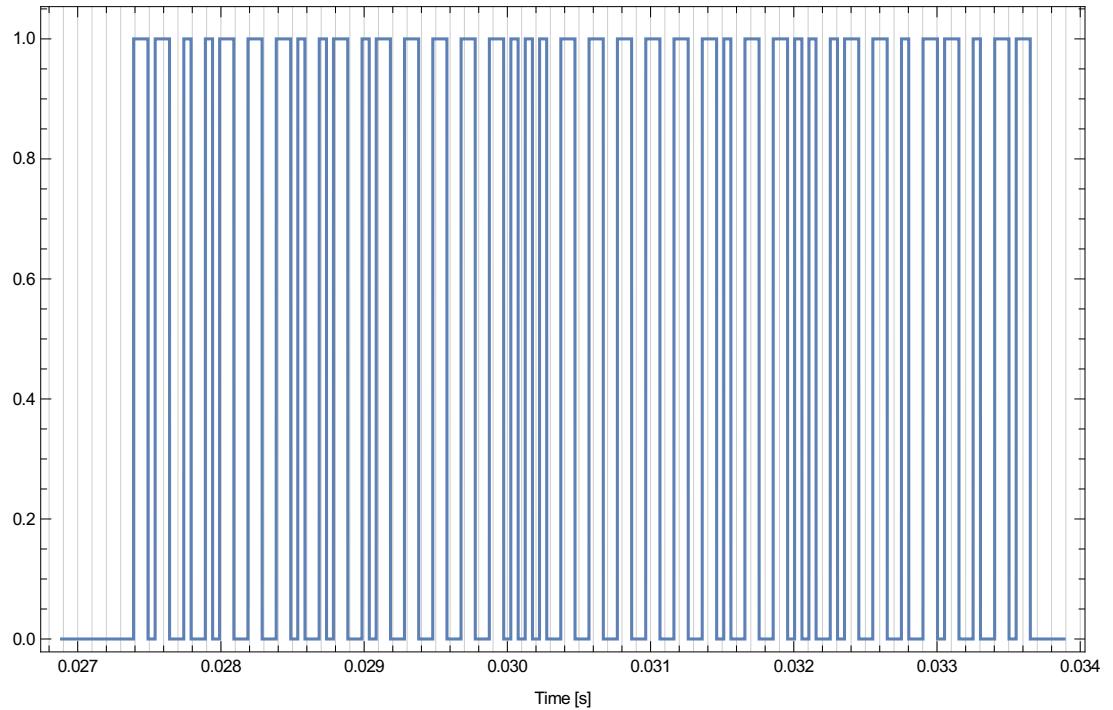




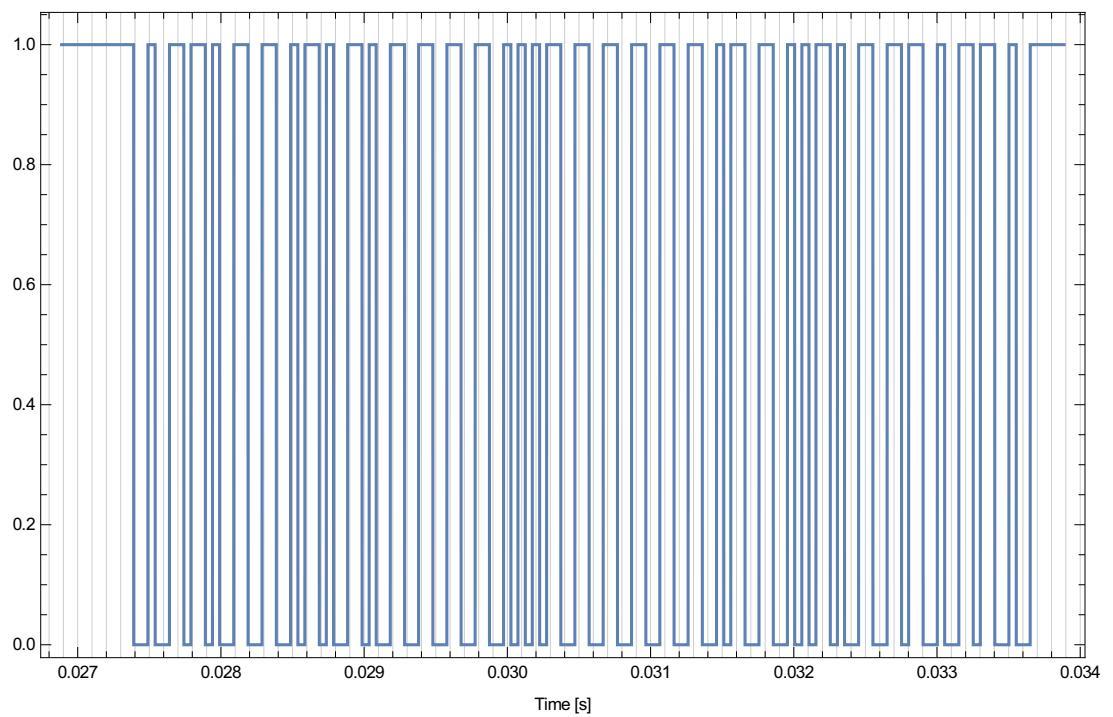
Medium length data packets

mfx signal

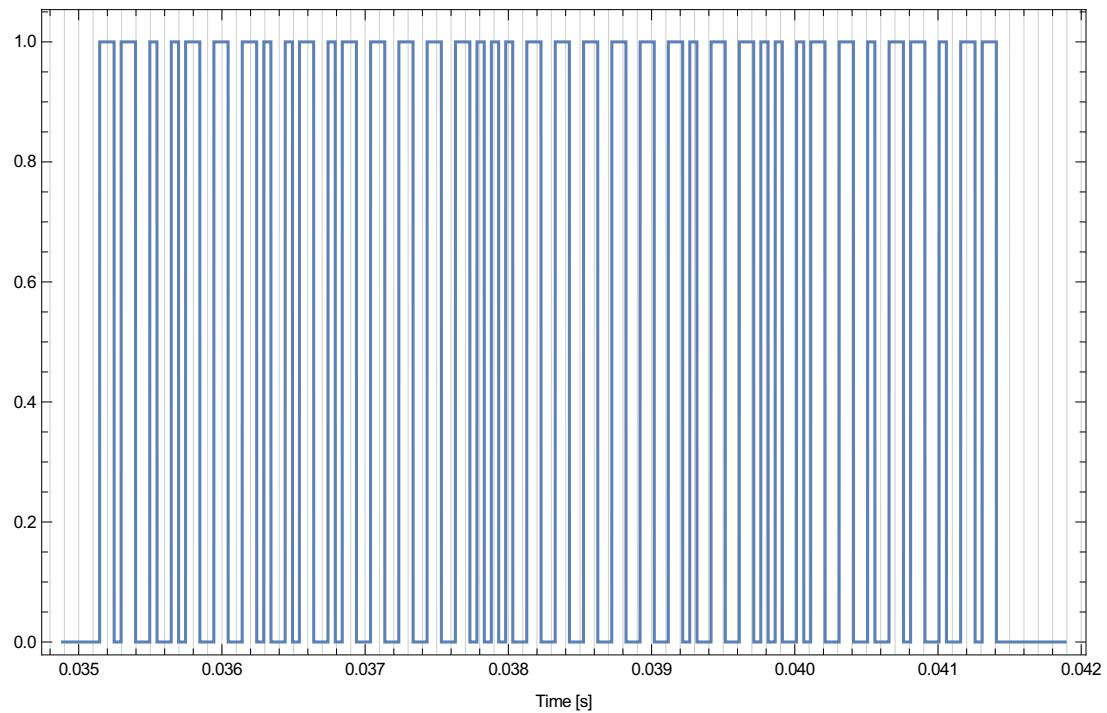
```
In[23]:= GraphicsGrid[  
{ListPlot[Take[#, {316 000, 330 000}], Frame -> True, Joined -> True, FrameLabel ->  
{"Time [s]", ""}, GridLines -> {Range[-.15, .15, 1.*^-4], None}]} &/@digitized]
```



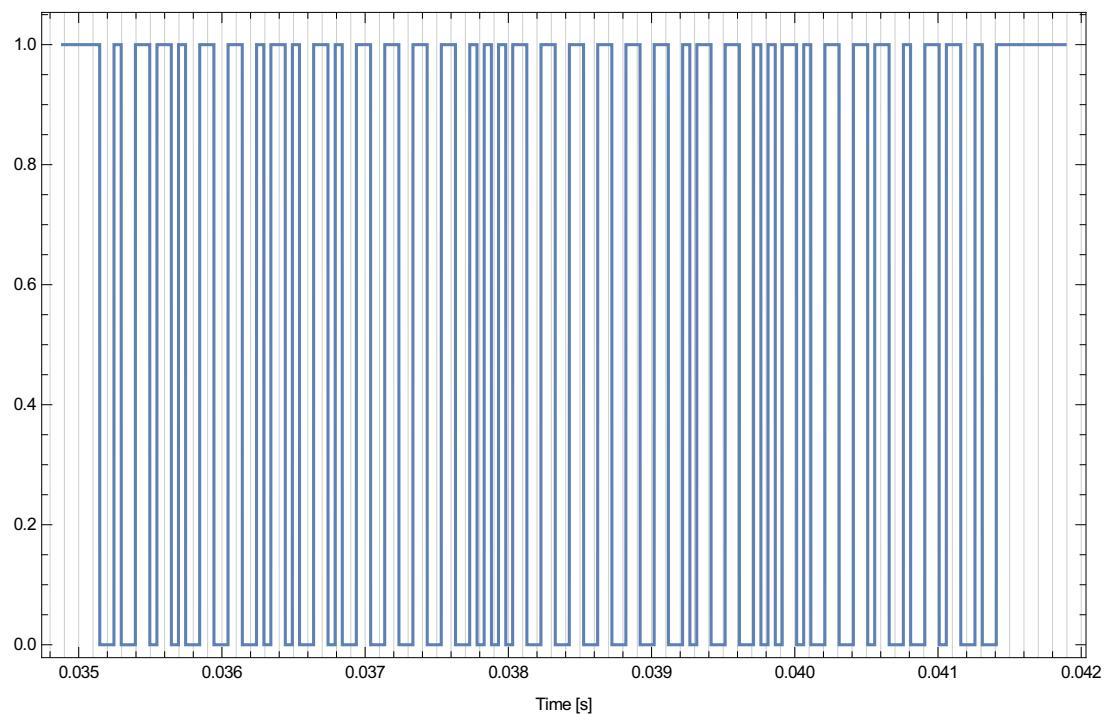
```
Out[23]=
```



```
In[24]:= GraphicsGrid[
{ListPlot[#, {332 000, 346 000}], Frame -> True, Joined -> True, FrameLabel ->
 {"Time [s]", ""}, GridLines -> {Range[-.15, .15, 1.*^-4], None}]} & /@ digitized]
```



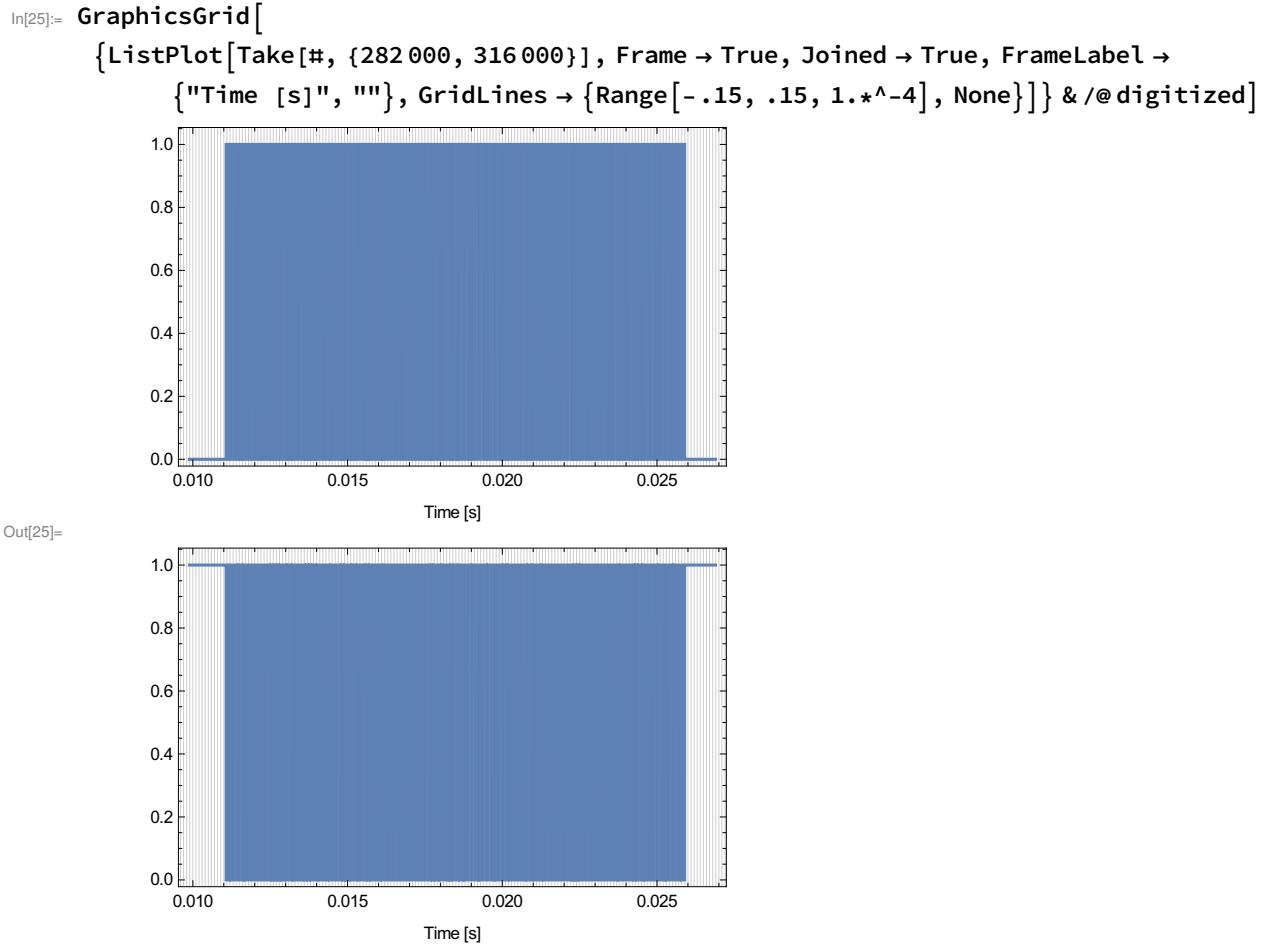
```
Out[24]=
```



manually:

SYNC, 7'A:5, drive fwd 7'0, func 16' F3 on, CRC8'b01101000, SYNC, SYNC

Long data packet



Run length decoding

bit frame (100 μ s) per sampling step

In[26]:= `timestep`
`10^-4`

Out[26]= `0.005`

Get run lengths of full traces

In[27]:= `Length /@ (fullruns = SplitBy[#, Last] & /@ digitized)`
Out[27]= `{2312, 2312}`

Run lengths in bit frames

In[28]:= `rlbits = Map[(timestep Length[#] &, fullruns, {2})]`
Out[28]= `{ {0.275, 0.995, 0.51, 0.48, 1., 0.99, 0.5, 0.5, 0.51, 0.485, 0.99, 0.985, 1.005, 0.495,
0.5, 0.985, 0.995, 0.98, 0.99, 0.985, 0.99, 0.98, 0.995, 0.98, 0.995, 0.985,`

0.995, 0.985, 0.995, 0.995, 0.51, 0.49, 0.51, 0.49, 1.005, 0.495, 0.495, 0.985,
 1., 0.495, 0.51, 0.495, 0.5, 0.995, 0.505, 0.495, 1.005, 0.5, 1., 0.99, 0.505,
 0.995, 1.005, 0.5, 1.005, 0.995, 0.505, 1.005, 1.005, 0.5, 1.005, 0.995, 0.505,
 0.99, 0.505, 0.49, 0.995, 0.985, 0.99, 0.98, 1.005, 0.495, 0.495, 0.995, 0.505,
 0.49, 0.995, 0.99, 0.51, 0.485, 0.995, 0.98, 0.995, 0.985, 0.995, 0.98, 0.995,
 0.985, 1.005, 0.495, 0.505, 0.495, 0.51, 0.5, 0.495, 0.98, 0.99, 0.98, 0.995,
 0.985, 0.99, 0.985, 0.995, 0.98, 0.995, 0.98, 1., 0.495, 0.495, 0.985, 0.995,
 0.985, 1., 0.5, 0.505, 0.5, 0.49, 0.995, 0.505, 0.49, 0.995, 0.98, 1.005, 0.99,
 0.51, 0.995, 1., 0.495, 1.005, 0.995, 0.51, 0.995, 1.005, 0.5, 0.99, 14.95,
 1.005, 0.495, 1.005, 0.995, 0.505, 0.995, 0.505, 0.48, 0.995, 0.985, 0.99,
 0.985, 1.005, 0.5, 0.5, 0.995, 0.505, 0.485, 0.995, 0.99, 0.51, 0.485, 0.995,
 0.98, 0.995, 0.98, 0.995, 0.985, 0.995, 0.98, 1.005, 0.49, 0.51, 0.495, 0.51,
 0.5, 0.495, 0.985, 0.995, 0.985, 0.995, 0.98, 0.995, 0.985, 0.99, 0.985, 0.99,
 0.985, 1.005, 0.495, 0.495, 0.985, 0.995, 0.98, 1.01, 0.495, 0.505, 0.5, 0.495,
 0.995, 0.51, 0.485, 0.99, 0.99, 1.005, 0.99, 0.505, 0.995, 1.01, 0.495, 1.,
 0.99, 0.505, 0.995, 1.01, 0.495, 0.99, 14.955, 1.005, 0.495, 1.005, 0.99, 0.505,
 0.995, 0.51, 0.485, 0.995, 0.985, 0.995, 0.98, 1.005, 0.5, 0.495, 0.995, 0.505,
 0.49, 0.995, 0.995, 0.51, 0.485, 0.995, 0.985, 0.99, 0.985, 0.995, 0.98, 0.995,
 0.98, 1.005, 0.495, 0.51, 0.5, 0.505, 0.495, 0.49, 0.985, 0.995, 0.985, 0.995,
 0.98, 0.99, 0.985, 0.995, 0.98, 0.995, 0.98, 1.005, 0.495, 0.495, 0.985, 0.995,
 0.98, 1., 0.5, 0.51, 0.495, 0.495, 0.995, 0.5, 0.49, 0.995, 0.985, 1., 0.99,
 0.505, 0.995, 1.01, 0.495, 1.005, 0.99, 0.51, 0.995, 1., 0.495, 0.99, 54.78,
 1.815, 0.255, 0.265, 1.805, 1.825, 0.255, 0.265, 1.81, 1.82, 0.255, 0.265,
 1.805, 1.82, 0.255, 0.265, 1.81, 0.265, 1.805, 0.265, 1.815, 1.82, 0.255, 0.265,
 1.805, 1.82, 0.255, 0.265, 1.805, 1.825, 0.25, 0.27, 1.81, 1.82, 0.255, 0.27,
 16.745, 1.815, 0.255, 0.265, 1.81, 1.82, 0.26, 0.26, 1.81, 1.82, 0.255, 0.27,
 1.81, 1.82, 0.255, 0.265, 1.81, 0.265, 1.805, 0.265, 1.81, 1.815, 0.255, 0.265,
 1.805, 1.82, 0.255, 0.265, 1.81, 1.82, 0.255, 0.265, 1.805, 1.82, 0.255, 0.265,
 41.645, 1.005, 0.495, 1.005, 0.995, 0.51, 0.99, 0.505, 0.485, 1., 0.985, 0.995,
 0.98, 0.995, 0.98, 1., 0.995, 0.51, 0.495, 0.51, 0.495, 0.51, 0.5, 0.505, 0.49,
 1.005, 0.49, 0.495, 0.995, 0.51, 0.48, 0.99, 0.985, 1., 0.495, 0.505, 0.5, 0.505,
 0.495, 0.495, 0.995, 0.505, 0.49, 0.995, 0.98, 1.005, 0.5, 0.49, 0.98, 0.995,
 0.99, 0.51, 0.49, 0.51, 0.485, 1.01, 0.495, 0.495, 0.99, 0.505, 0.485, 0.995,
 0.99, 0.51, 0.5, 0.505, 0.49, 0.99, 0.985, 1.01, 0.495, 0.5, 0.985, 0.995,
 0.985, 0.995, 0.985, 0.995, 0.98, 0.995, 0.985, 0.995, 0.985, 0.99, 0.98, 0.995,
 0.99, 0.505, 0.5, 0.505, 0.49, 1., 0.5, 0.49, 0.985, 1.01, 0.49, 0.495, 0.51,
 0.495, 0.99, 0.51, 0.49, 1.005, 0.495, 1.01, 0.995, 0.505, 0.99, 1.01, 0.495,
 1., 0.995, 0.505, 1.005, 1.01, 0.49, 1.005, 0.995, 0.505, 0.99, 0.505, 0.485,
 1., 0.98, 0.995, 0.98, 1.005, 0.5, 0.49, 0.995, 0.505, 0.49, 0.995, 0.99, 0.51,
 0.485, 0.99, 0.98, 0.99, 0.985, 0.995, 0.98, 0.99, 0.98, 1.005, 0.495, 0.505,
 0.495, 0.51, 0.495, 0.5, 0.98, 0.99, 0.98, 0.995, 0.985, 0.99, 0.985, 0.995,
 0.98, 0.995, 0.98, 1., 0.495, 0.495, 0.98, 1., 0.985, 1.005, 0.495, 0.505, 0.5,
 0.495, 0.995, 0.51, 0.49, 0.995, 0.98, 1., 0.995, 0.51, 0.99, 1., 0.5, 1.005,
 0.99, 0.51, 0.995, 1.005, 0.5, 0.985, 14.95, 1.005, 0.495, 1.01, 0.995, 0.5,
 0.995, 0.51, 0.485, 0.995, 0.98, 0.995, 0.98, 1.005, 0.495, 0.5, 0.995, 0.505,
 0.485, 0.995, 0.99, 0.51, 0.485, 0.995, 0.985, 0.99, 0.98, 0.995, 0.985, 0.995,
 0.985, 1., 0.5, 0.51, 0.495, 0.51, 0.495, 0.5, 0.985, 0.995, 0.985, 0.99, 0.985,
 1., 0.98, 0.99, 0.98, 0.99, 0.985, 1.005, 0.49, 0.5, 0.985, 0.99, 0.985, 1.01,
 0.495, 0.505, 0.495, 0.5, 0.99, 0.505, 0.485, 0.995, 0.985, 1.005, 0.99, 0.5,

0.995, 1.005, 0.495, 1., 0.995, 0.505, 0.995, 1.01, 0.495, 0.99, 14.955, 1.005,
 0.5, 1.005, 0.995, 0.51, 0.995, 0.505, 0.49, 0.99, 0.98, 0.99, 0.985, 1.005,
 0.495, 0.495, 0.99, 0.5, 0.49, 0.995, 0.99, 0.51, 0.49, 0.995, 0.98, 0.99,
 0.98, 0.995, 0.98, 0.995, 0.985, 1.005, 0.5, 0.505, 0.5, 0.505, 0.5, 0.495,
 0.985, 0.99, 0.98, 0.995, 0.985, 0.995, 0.985, 0.99, 0.98, 0.995, 0.985, 1.005,
 0.495, 0.5, 0.985, 0.995, 0.985, 1.005, 0.495, 0.51, 0.495, 0.5, 0.99, 0.505,
 0.485, 0.995, 0.985, 1.005, 0.995, 0.51, 0.995, 1., 0.49, 1.01, 0.995, 0.505,
 0.995, 1.005, 0.495, 0.99, 54.78, 1.815, 0.255, 0.265, 1.805, 1.82, 0.25,
 0.265, 1.81, 1.82, 0.255, 0.265, 1.81, 1.82, 0.255, 0.265, 1.81, 0.265, 1.81,
 0.27, 1.81, 1.815, 0.255, 0.265, 1.805, 1.82, 0.26, 0.26, 1.81, 1.82, 0.255,
 0.27, 1.805, 1.825, 0.255, 0.26, 16.735, 1.815, 0.255, 0.265, 1.805, 1.82,
 0.26, 0.265, 1.81, 1.82, 0.255, 0.265, 1.805, 1.82, 0.25, 0.27, 1.81, 0.26,
 1.81, 0.27, 1.805, 1.82, 0.25, 0.27, 1.81, 1.815, 0.255, 0.265, 1.805, 1.82,
 0.25, 0.265, 1.815, 1.815, 0.26, 0.265, 41.635, 1.005, 0.5, 1., 0.995, 0.505,
 0.995, 0.5, 0.49, 0.99, 0.985, 0.995, 0.98, 0.995, 0.985, 0.995, 0.995, 0.51,
 0.5, 0.505, 0.5, 0.505, 0.5, 0.51, 0.485, 1.005, 0.5, 0.495, 0.99, 0.51, 0.49,
 0.995, 0.985, 1.005, 0.5, 0.505, 0.495, 0.505, 0.495, 0.5, 0.995, 0.5, 0.485,
 0.99, 0.98, 1., 0.495, 0.5, 0.985, 0.995, 0.995, 0.51, 0.49, 0.51, 0.49, 1.005,
 0.495, 0.495, 0.995, 0.505, 0.485, 0.995, 0.995, 0.5, 0.5, 0.51, 0.485, 0.995,
 0.985, 1.005, 0.495, 0.495, 0.985, 0.995, 0.98, 1., 0.985, 0.99, 0.98, 0.995,
 0.99, 0.995, 0.985, 0.99, 0.98, 1., 0.995, 0.505, 0.5, 0.505, 0.49, 1.005, 0.5,
 0.49, 0.985, 1., 0.49, 0.51, 0.5, 0.495, 0.99, 0.51, 0.5, 1.005, 0.49, 1.01,
 0.995, 0.505, 0.995, 1.005, 0.495, 1.005, 0.99, 0.505, 1.01, 1.005, 0.49, 1.01,
 0.99, 0.505, 0.99, 0.51, 0.485, 0.995, 0.98, 0.99, 0.985, 1.005, 0.495, 0.5,
 0.995, 0.505, 0.49, 0.995, 0.995, 0.51, 0.49, 0.995, 0.985, 0.995, 0.98, 0.995,
 0.98, 0.995, 0.98, 1.005, 0.495, 0.505, 0.495, 0.505, 0.495, 0.5, 0.985, 0.995,
 0.985, 1., 0.985, 0.995, 0.985, 0.995, 0.98, 0.995, 0.985, 0.985, 1.005, 0.495, 0.5,
 0.985, 0.995, 0.98, 1.005, 0.495, 0.505, 0.495, 0.5, 0.995, 0.505, 0.48, 1.,
 0.985, 1.005, 0.995, 0.51, 0.99, 1.005, 0.5, 1.005, 0.995, 0.51, 0.995, 1.005,
 0.495, 0.99, 14.955, 1.005, 0.495, 1., 0.995, 0.505, 0.995, 0.51, 0.485, 1.,
 0.985, 0.995, 0.98, 1.005, 0.495, 0.505, 0.495, 0.5, 0.995, 0.505, 0.48, 1.,
 0.985, 1.005, 0.995, 0.51, 0.99, 1.005, 0.5, 1.005, 0.995, 0.51, 0.995, 1.005,
 0.495, 0.99, 14.955, 1.005, 0.495, 1., 0.995, 0.505, 0.995, 0.51, 0.485, 1.,
 0.985, 0.995, 0.98, 1., 0.5, 0.495, 0.995, 0.505, 0.48, 1., 0.99, 0.5, 0.49,
 0.99, 0.985, 0.995, 0.985, 0.995, 0.985, 0.99, 0.985, 1.005, 0.495, 0.51,
 0.495, 0.505, 0.495, 0.495, 0.98, 0.995, 0.98, 0.995, 0.985, 0.995, 0.98,
 1., 0.985, 0.99, 0.98, 1., 0.5, 0.49, 0.985, 0.995, 0.985, 1.005, 0.5, 0.505,
 0.495, 0.5, 0.995, 0.505, 0.485, 0.995, 0.985, 1.005, 0.995, 0.5, 1., 1.005,
 0.5, 1., 0.995, 0.505, 1., 1.005, 0.495, 0.985, 14.955, 1.005, 0.5, 1.005,
 0.995, 0.51, 0.995, 0.5, 0.485, 0.995, 0.98, 1., 0.98, 1.005, 0.49, 0.5, 0.99,
 0.505, 0.485, 0.995, 0.99, 0.505, 0.49, 0.995, 0.985, 0.99, 0.985, 1., 0.985,
 0.995, 0.98, 1., 0.5, 0.5, 0.495, 0.505, 0.49, 0.5, 0.985, 0.995, 0.98, 0.995,
 0.99, 0.995, 0.985, 0.99, 0.98, 0.995, 0.985, 0.985, 1.005, 0.495, 0.5, 0.985,
 0.995, 0.98, 1.005, 0.495, 0.505, 0.495, 0.5, 0.985, 0.995, 0.985, 1.005,
 0.99, 0.505, 0.995, 1.005, 0.5, 1.005, 0.995, 0.51, 0.995, 1.005, 0.5, 0.99,
 54.775, 1.825, 0.255, 0.265, 1.81, 1.82, 0.255, 0.265, 1.805, 1.82, 0.255,
 0.265, 1.81, 1.815, 0.255, 0.265, 1.805, 0.265, 1.81, 0.265, 1.805, 1.82, 0.25,
 0.27, 1.81, 1.815, 0.255, 0.265, 1.805, 1.82, 0.26, 0.27, 1.81, 1.815, 0.255,
 0.265, 16.735, 1.815, 0.255, 0.265, 1.81, 1.82, 0.255, 0.265, 1.81, 1.815,
 0.255, 0.265, 1.805, 1.82, 0.255, 0.26, 1.81, 0.27, 1.805, 0.265, 1.81, 1.815,
 0.255, 0.265, 1.81, 1.82, 0.255, 0.265, 1.81, 1.815, 0.26, 0.265, 1.805, 1.82,
 0.25, 0.265, 41.645, 1.005, 0.5, 1.005, 0.995, 0.51, 0.995, 0.505, 0.485, 0.99,

0.985, 0.995, 0.985, 0.995, 0.985, 0.99, 0.99, 0.505, 0.5, 0.505, 0.495, 0.505,
 0.495, 0.505, 0.485, 1., 0.5, 0.49, 0.995, 0.505, 0.49, 0.99, 0.98, 1.005, 0.5,
 0.505, 0.495, 0.51, 0.495, 0.495, 0.99, 0.505, 0.49, 0.995, 0.98, 1., 0.5,
 0.495, 0.985, 0.995, 0.995, 0.505, 0.495, 0.51, 0.485, 1.005, 0.495, 0.495,
 1., 0.505, 0.485, 0.995, 0.99, 0.505, 0.495, 0.51, 0.485, 0.99, 0.985, 1.005,
 0.495, 0.5, 0.985, 0.99, 0.99, 0.995, 0.985, 0.995, 0.985, 1., 0.985, 0.995,
 0.985, 0.99, 0.98, 0.995, 0.995, 0.5, 0.5, 0.51, 0.485, 1., 0.5, 0.495, 0.98,
 1.005, 0.495, 0.505, 0.5, 0.495, 1., 0.505, 0.5, 1.005, 0.495, 1.005, 0.995,
 0.505, 1., 1.005, 0.495, 1.01, 0.995, 0.51, 1.005, 1.005, 0.495, 1.005, 0.99,
 0.51, 0.99, 0.505, 0.49, 0.995, 0.985, 0.995, 0.99, 1.005, 0.495, 0.5, 0.995,
 0.505, 0.485, 0.995, 0.99, 0.51, 0.49, 0.99, 0.98, 0.995, 0.98, 0.995, 0.98,
 0.995, 0.985, 1.01, 0.495, 0.51, 0.495, 0.51, 0.5, 0.495, 0.99, 0.995, 0.985,
 0.99, 0.985, 0.995, 0.98, 0.995, 0.98, 0.995, 0.98, 1.005, 0.495, 0.495, 0.985,
 0.995, 0.98, 1.005, 0.495, 0.51, 0.495, 0.5, 0.99, 0.505, 0.485, 1., 0.98,
 1.005, 0.99, 0.505, 0.99, 1.005, 0.5, 1., 0.995, 0.51, 0.995, 1., 0.49, 0.99,
 14.955, 1.005, 0.5, 1.005, 0.995, 0.51, 0.995, 0.505, 0.49, 0.995, 0.985, 0.99,
 0.985, 1.01, 0.495, 0.5, 0.995, 0.51, 0.485, 1., 0.995, 0.505, 0.485, 0.995,
 0.98, 0.99, 0.985, 0.99, 0.985, 0.99, 0.985, 1., 0.5, 0.51, 0.495, 0.505, 0.495,
 0.5, 0.985, 0.99, 0.98, 0.995, 0.985, 0.995, 0.985, 0.99, 0.98, 0.995, 0.98,
 1.005, 0.5, 0.495, 0.98, 0.99, 0.985, 1.005, 0.49, 0.51, 0.49, 0.495, 0.995,
 0.505, 0.49, 0.995, 0.98, 1.005, 0.995, 0.51, 0.995, 1.01, 0.495, 1.005, 0.99,
 0.51, 0.995, 1., 0.5, 0.985, 14.95, 1.01, 0.49, 1.005, 0.995, 0.505, 0.995,
 0.51, 0.49, 0.995, 0.98, 0.995, 0.985, 1.01, 0.495, 0.5, 0.995, 0.51, 0.485,
 0.995, 1., 0.505, 0.49, 0.99, 0.985, 0.995, 0.98, 0.995, 0.98, 0.995, 0.98,
 1.01, 0.495, 0.51, 0.495, 0.51, 0.49, 0.495, 0.98, 0.995, 0.99, 0.99, 0.985,
 0.995, 0.98, 0.99, 0.985, 0.99, 0.98, 1.005, 0.495, 0.5, 0.985, 0.99, 0.985,
 1.005, 0.495, 0.51, 0.495, 0.5, 0.99, 0.505, 0.485, 1., 0.985, 1., 0.99, 0.505,
 0.99, 1.005, 0.5, 1., 0.99, 0.51, 0.995, 1.005, 0.5, 0.99, 54.785, 1.815,
 0.255, 0.27, 1.805, 1.82, 0.255, 0.27, 1.81, 1.815, 0.255, 0.265, 1.805, 1.82,
 0.26, 0.265, 1.81, 0.265, 1.81, 0.265, 1.81, 1.815, 0.255, 0.265, 1.81, 1.82,
 0.255, 0.265, 1.81, 1.815, 0.255, 0.27, 1.805, 1.82, 0.255, 0.265, 16.735,
 1.82, 0.255, 0.265, 1.81, 1.82, 0.26, 0.26, 1.81, 1.815, 0.255, 0.265, 1.81,
 1.82, 0.255, 0.26, 1.81, 0.265, 1.805, 0.265, 1.805, 1.82, 0.255, 0.27, 1.805,
 1.82, 0.25, 0.265, 1.81, 1.815, 0.26, 0.265, 1.81, 1.82, 0.255, 0.265, 41.645,
 1.005, 0.5, 1.005, 0.995, 0.51, 0.995, 0.505, 0.485, 0.995, 0.98, 0.995, 0.98,
 0.995, 0.985, 0.995, 0.99, 0.51, 0.495, 0.505, 0.49, 0.505, 0.5, 0.505, 0.49,
 1.005, 0.49, 0.495, 0.995, 0.505, 0.48, 1., 0.98, 1., 0.5, 0.505, 0.5, 0.505,
 0.495, 0.49, 0.995, 0.505, 0.49, 0.995, 0.98, 1.005, 0.5, 0.49, 0.98, 0.995,
 0.99, 0.505, 0.495, 0.51, 0.485, 1.01, 0.495, 0.5, 0.995, 0.505, 0.49, 0.995,
 0.99, 0.505, 0.5, 0.505, 0.49, 0.995, 0.98, 1.01, 0.495, 0.495, 0.985, 0.995,
 0.98, 0.99, 0.985, 0.995, 0.985, 0.995, 0.98, 0.99, 0.99, 0.995, 0.985, 0.995,
 0.995, 0.51, 0.5, 0.505, 0.485, 1.005, 0.5, 0.495, 0.985, 1.005, 0.5, 0.505,
 0.495, 0.5, 0.995, 0.51, 0.495, 1.005, 0.5, 1., 0.99, 0.505, 0.995, 1., 0.5,
 1.005, 0.99, 0.505, 1.005, 1.005, 0.5, 1., 0.995, 0.51, 0.99, 0.51, 0.485,
 0.995, 0.985, 0.995, 0.985, 1.01, 0.49, 0.5, 0.995, 0.51, 0.485, 0.99, 0.995,
 0.5, 0.49, 0.99, 0.985, 0.995, 0.985, 0.995, 0.985, 0.99, 0.985, 1.005, 0.5,
 0.51, 0.495, 0.51, 0.495, 0.5, 0.985, 0.995, 0.995, 0.99, 0.995, 0.985, 0.995,
 0.995, 0.985, 0.995, 0.98, 1., 0.5, 0.49, 0.99, 0.99, 0.99, 0.985, 1., 0.5, 0.505,
 0.5, 0.49, 1., 0.505, 0.49, 0.995, 0.985, 1.005, 1., 0.505, 0.995, 1.005,

0.5, 1.005, 0.995, 0.51, 0.995, 1.005, 0.5, 0.99, 14.95, 1.005, 0.5, 1.005,
 0.99, 0.505, 0.995, 0.505, 0.49, 0.995, 0.98, 0.995, 0.985, 1.01, 0.495, 0.5,
 0.99, 0.51, 0.485, 0.995, 0.995, 0.505, 0.49, 0.995, 0.985, 0.995, 0.98, 1.,
 0.98, 0.99, 0.98, 1.005, 0.495, 0.505, 0.495, 0.505, 0.495, 0.5, 0.985, 0.995,
 0.985, 1., 0.985, 0.995, 0.985, 0.995, 0.98, 0.99, 0.985, 1., 0.495, 0.495,
 0.985, 0.995, 0.98, 1.01, 0.495, 0.505, 0.495, 0.5, 0.995, 0.505, 0.485, 0.995,
 0.985, 1.005, 0.99, 0.51, 0.995, 1., 0.49, 1.01, 0.995, 0.505, 0.99, 1.005,
 0.495, 0.99, 14.955, 1.01, 0.495, 1.005, 0.995, 0.505, 0.995, 0.51, 0.485,
 0.99, 0.985, 1., 0.98, 1.005, 0.5, 0.49, 0.995, 0.51, 0.485, 0.995, 0.99, 0.51,
 0.485, 0.99, 0.99, 0.995, 0.98, 0.99, 0.985, 0.99, 0.985, 1.005, 0.5, 0.505,
 0.5, 0.505, 0.5, 0.495, 0.985, 0.995, 0.985, 0.995, 0.985, 0.995, 0.98, 0.995,
 0.98, 0.995, 0.985, 1.005, 0.5, 0.495, 0.98, 0.995, 0.98, 1.005, 0.5, 0.505,
 0.5, 0.495, 0.995, 0.505, 0.485, 0.99, 0.985, 1., 0.99, 0.505, 0.995, 1.01,
 0.495, 1.005, 0.995, 0.5, 1., 1.005, 0.49, 0.99, 54.78, 1.815, 0.255, 0.265,
 1.805, 1.82, 0.26, 0.265, 1.81, 1.82, 0.25, 0.265, 1.81, 1.815, 0.26, 0.265,
 1.805, 0.27, 1.805, 0.265, 1.81, 1.815, 0.255, 0.27, 1.805, 1.82, 0.255, 0.26,
 1.81, 1.815, 0.255, 0.27, 1.81, 1.82, 0.255, 0.265, 16.74, 1.82, 0.255, 0.265,
 1.805, 1.82, 0.25, 0.27, 1.81, 1.815, 0.255, 0.265, 1.81, 1.82, 0.25, 0.265,
 1.81, 0.265, 1.81, 0.265, 1.81, 1.82, 0.25, 0.265, 1.805, 1.82, 0.255, 0.26,
 1.81, 1.82, 0.255, 0.265, 1.81, 1.82, 0.255, 0.26, 41.645, 1.01, 0.495, 1.005,
 0.995, 0.505, 0.995, 0.51, 0.485, 0.99, 0.985, 0.995, 0.985, 0.99, 0.98, 0.995,
 0.99, 0.505, 0.495, 0.505, 0.495, 0.51, 0.495, 0.51, 0.485, 1., 0.495, 0.49,
 0.99, 0.505, 0.49, 0.995, 0.98, 1.005, 0.495, 0.51, 0.495, 0.51, 0.49, 0.495,
 0.995, 0.51, 0.48, 0.99, 0.98, 1.005, 0.495, 0.495, 0.985, 0.99, 0.995, 0.51,
 0.5, 0.505, 0.49, 1., 0.5, 0.495, 0.995, 0.51, 0.49, 0.99, 0.995, 0.51, 0.495,
 0.505, 0.485, 0.995, 0.985, 1.005, 0.495, 0.5, 0.98, 0.995, 0.98, 0.995, 0.985,
 0.995, 0.98, 1., 0.985, 0.995, 0.98, 0.995, 0.985, 0.99, 0.995, 0.5, 0.5, 0.505,
 0.485, 1., 0.495, 0.5, 0.98, 1.005, 0.495, 0.505, 0.5, 0.495, 0.995, 0.505,
 0.495, 1.005, 0.495, 1., 0.995, 0.505, 0.99, 0.51, 0.485, 1., 0.985, 0.995, 0.98,
 1.005, 0.495, 0.5, 0.995, 0.505, 0.49, 0.99, 0.99, 0.505, 0.485, 1., 0.98,
 0.995, 0.985, 0.995, 0.985, 0.995, 0.98, 1., 0.495, 0.51, 0.495, 0.505, 0.495,
 0.495, 0.98, 0.995, 0.98, 0.995, 0.98, 0.99, 0.985, 0.99, 0.985, 0.99, 0.98,
 1.005, 0.5, 0.495, 0.985, 0.995, 0.985, 1.005, 0.5, 0.51, 0.495, 0.5, 0.995,
 0.505, 0.485, 0.995, 0.985, 1.005, 0.995, 0.505, 0.995, 1.005, 0.49, 1.005,
 0.995, 0.51, 0.99, 1., 0.5, 0.99, 14.955, 1.005, 0.5, 1.005, 0.995, 0.51,
 0.99, 0.505, 0.49, 0.995, 0.985, 0.995, 0.98, 1., 0.5, 0.495, 0.995, 0.505,
 0.49, 0.995, 0.995, 0.51, 0.485, 0.99, 0.99, 0.995, 0.98, 0.995, 0.985, 1.,
 0.98, 1.005, 0.495, 0.505, 0.5, 0.51, 0.495, 0.5, 0.985, 0.99, 0.985, 0.99,
 0.985, 0.995, 0.985, 0.99, 0.985, 0.995, 0.985, 1., 0.495, 0.495, 0.98, 0.995,
 0.985, 1.005, 0.495, 0.51, 0.495, 0.5, 0.99, 0.51, 0.48, 0.99, 0.985, 1.005,
 0.995, 0.505, 0.995, 1., 0.495, 1.005, 0.995, 0.505, 0.99, 1.005, 0.295},
{0.27, 1.005, 0.5, 0.49, 0.985, 1., 0.495, 0.51, 0.495, 0.5, 0.98, 0.99,
 0.995, 0.51, 0.49, 0.995, 0.985, 0.99, 0.98, 0.99, 0.985, 0.995, 0.985, 0.99,
 0.98, 0.995, 0.99, 0.995, 0.985, 1.005, 0.5, 0.5, 0.5, 0.495, 1., 0.5, 0.49,
 0.995, 0.99, 0.505, 0.495, 0.51, 0.485, 1.01, 0.49, 0.51, 0.995, 0.505,
 0.995, 1., 0.49, 1.01, 0.995, 0.505, 1., 1.005, 0.495, 1.015, 0.995, 0.505,
 0.995, 1.005, 0.495, 1.005, 0.495, 0.495, 0.985, 0.995, 0.985, 0.99, 0.995,
 0.505, 0.485, 1.005, 0.495, 0.495, 0.985, 1.005, 0.495, 0.5, 0.985, 0.99,

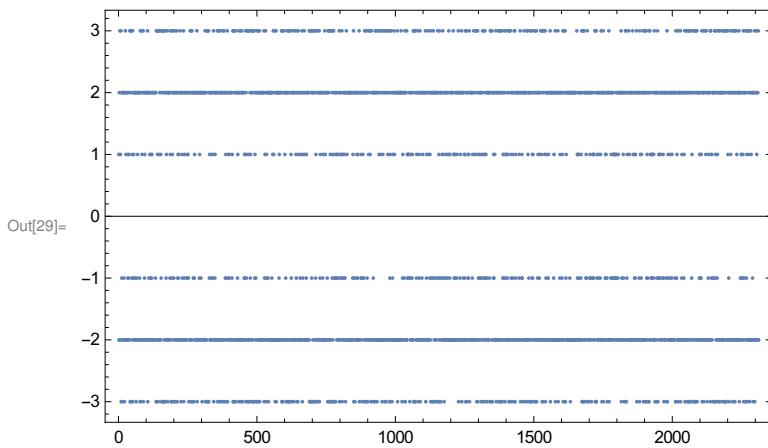
0.98, 0.995, 0.99, 0.99, 0.985, 0.995, 0.99, 0.51, 0.495, 0.505, 0.5, 0.505,
 0.49, 0.99, 0.98, 0.99, 0.985, 0.995, 0.98, 0.995, 0.985, 0.99, 0.98, 0.995,
 0.99, 0.505, 0.485, 0.99, 0.985, 0.995, 0.995, 0.51, 0.495, 0.505, 0.485,
 1.005, 0.495, 0.495, 0.99, 0.99, 0.995, 1., 0.5, 1.005, 0.99, 0.505, 0.995,
 1.005, 0.5, 1.005, 0.995, 0.505, 0.98, 14.965, 0.99, 0.51, 0.99, 1.005, 0.5,
 1.005, 0.49, 0.495, 0.98, 0.995, 0.985, 0.995, 0.995, 0.51, 0.485, 1.005,
 0.5, 0.49, 0.985, 1.005, 0.495, 0.5, 0.98, 0.995, 0.98, 0.995, 0.985, 0.995,
 0.98, 0.995, 0.99, 0.505, 0.5, 0.505, 0.5, 0.505, 0.49, 0.995, 0.985, 0.995,
 0.98, 0.995, 0.985, 0.995, 0.98, 0.995, 0.98, 0.995, 0.995, 0.5, 0.49, 0.995,
 0.98, 0.995, 0.995, 0.51, 0.495, 0.505, 0.49, 1.005, 0.495, 0.5, 0.98, 0.995,
 0.995, 1.005, 0.495, 1.005, 0.995, 0.51, 0.99, 1., 0.495, 1.005, 0.995, 0.51,
 0.98, 14.96, 1., 0.5, 0.995, 1., 0.5, 1.005, 0.5, 0.495, 0.98, 0.995, 0.985,
 0.995, 0.51, 0.485, 1., 0.5, 0.5, 0.985, 1.005, 0.495, 0.5, 0.985,
 0.99, 0.985, 0.995, 0.985, 0.99, 0.985, 0.99, 0.995, 0.505, 0.5, 0.505, 0.5,
 0.5, 0.485, 0.99, 0.985, 1., 0.985, 0.99, 0.98, 0.995, 0.98, 0.995, 0.985,
 0.99, 0.99, 0.51, 0.48, 0.995, 0.985, 0.995, 0.99, 0.505, 0.5, 0.51, 0.485,
 1., 0.495, 0.5, 0.985, 0.995, 0.99, 1., 0.495, 1.005, 0.995, 0.51, 0.995, 1.,
 0.5, 1.005, 0.99, 0.505, 0.98, 54.785, 1.81, 0.265, 0.255, 1.815, 1.815, 0.26,
 0.255, 1.82, 1.81, 0.27, 0.255, 1.815, 1.81, 0.265, 0.255, 1.82, 0.255, 1.815,
 0.255, 1.825, 1.805, 0.27, 0.255, 1.815, 1.81, 0.265, 0.25, 1.82, 1.81, 0.265,
 0.255, 1.825, 1.805, 0.27, 0.26, 16.75, 1.81, 0.265, 0.255, 1.82, 1.81, 0.265,
 0.255, 1.82, 1.81, 0.265, 0.26, 1.815, 1.815, 0.265, 0.25, 1.825, 0.255, 1.815,
 0.255, 1.815, 1.81, 0.265, 0.255, 1.815, 1.81, 0.265, 0.255, 1.82, 1.805,
 0.27, 0.255, 1.815, 1.81, 0.265, 0.255, 1.815, 1.81, 0.265, 0.255, 1.82, 1.805,
 0.27, 0.255, 1.815, 1.81, 0.265, 0.25, 41.66, 0.995, 0.505, 0.99, 1.005, 0.5,
 1.005, 0.495, 0.495, 0.99, 0.995, 0.985, 0.99, 0.98, 0.995, 0.99, 1.005, 0.495,
 0.51, 0.5, 0.505, 0.5, 0.505, 0.5, 0.495, 0.995, 0.505, 0.485, 1.005, 0.495,
 0.495, 0.98, 0.99, 0.995, 0.505, 0.495, 0.505, 0.5, 0.505, 0.485, 1.005, 0.495,
 0.5, 0.985, 0.99, 0.995, 0.505, 0.485, 0.99, 0.98, 1.005, 0.495, 0.505, 0.5,
 0.495, 0.995, 0.51, 0.48, 1.005, 0.495, 0.495, 0.985, 1., 0.5, 0.51, 0.495,
 0.495, 0.985, 0.995, 0.995, 0.51, 0.49, 0.995, 0.985, 0.995, 0.985, 0.995,
 0.98, 0.995, 0.985, 0.995, 0.98, 0.995, 0.98, 0.995, 0.985, 1., 0.495, 0.51,
 0.495, 0.495, 0.995, 0.505, 0.485, 0.995, 0.995, 0.505, 0.5, 0.505, 0.49,
 1., 0.495, 0.505, 0.99, 0.51, 0.995, 1.005, 0.5, 1., 0.995, 0.51, 0.99, 1., 0.5,
 1.015, 0.995, 0.505, 0.995, 1.005, 0.495, 1., 0.49, 0.5, 0.985, 0.995, 0.98,
 0.995, 0.995, 0.505, 0.485, 1.005, 0.495, 0.495, 0.985, 1.005, 0.495, 0.5,
 0.98, 0.99, 0.98, 0.995, 0.98, 0.995, 0.98, 0.99, 0.99, 0.51, 0.495, 0.505,
 0.5, 0.505, 0.49, 0.99, 0.98, 0.99, 0.985, 0.995, 0.98, 0.995, 0.985, 0.99,
 0.98, 0.995, 0.99, 0.5, 0.49, 0.99, 0.985, 1., 0.995, 0.5, 0.5, 0.51, 0.485,
 1.005, 0.5, 0.495, 0.985, 0.995, 0.99, 1.005, 0.5, 1., 0.99, 0.51, 0.995, 1.,
 0.5, 1.005, 0.995, 0.505, 0.975, 14.96, 1., 0.505, 0.995, 1.005, 0.495, 1.005,
 0.5, 0.495, 0.98, 0.995, 0.985, 0.99, 0.99, 0.51, 0.49, 1.005, 0.495, 0.495,
 0.985, 1., 0.5, 0.495, 0.985, 0.99, 0.985, 0.99, 0.985, 0.995, 0.985, 0.995,
 0.99, 0.51, 0.495, 0.51, 0.495, 0.51, 0.485, 1., 0.985, 0.995, 0.98, 0.995,
 0.985, 0.99, 0.985, 0.99, 0.98, 0.99, 0.995, 0.505, 0.49, 0.995, 0.98, 0.995,
 0.995, 0.51, 0.495, 0.505, 0.485, 1.005, 0.49, 0.5, 0.985, 0.995, 0.99, 1.,
 0.495, 1., 0.995, 0.51, 0.99, 1.005, 0.495, 1.005, 0.995, 0.51, 0.98, 14.965,
 0.995, 0.51, 0.995, 1.005, 0.495, 1.005, 0.5, 0.495, 0.98, 0.995, 0.98, 0.995,
 0.99, 0.51, 0.48, 1., 0.495, 0.5, 0.985, 1., 0.5, 0.495, 0.985, 0.99, 0.985,
 0.99, 0.985, 0.99, 0.98, 1., 0.995, 0.505, 0.5, 0.51, 0.495, 0.51, 0.48,

0.995, 0.985, 0.99, 0.985, 0.995, 0.985, 0.99, 0.985, 0.99, 0.985, 0.995,
 0.995, 0.505, 0.49, 0.995, 0.985, 0.995, 0.99, 0.51, 0.495, 0.51, 0.485,
 1.005, 0.49, 0.5, 0.985, 0.99, 0.995, 1.01, 0.495, 1.005, 0.99, 0.505, 0.995,
 1.01, 0.495, 1.005, 0.99, 0.51, 0.98, 54.785, 1.81, 0.265, 0.25, 1.82, 1.805,
 0.265, 0.255, 1.82, 1.81, 0.265, 0.255, 1.82, 1.805, 0.27, 0.255, 1.815, 0.26,
 1.82, 0.26, 1.82, 1.805, 0.265, 0.255, 1.815, 1.81, 0.265, 0.255, 1.82, 1.81,
 0.265, 0.26, 1.815, 1.81, 0.265, 0.255, 16.745, 1.805, 0.265, 0.255, 1.82, 1.81,
 1.81, 0.265, 0.255, 1.825, 1.805, 0.27, 0.255, 1.815, 1.805, 0.265, 0.255,
 1.82, 0.255, 1.82, 0.255, 1.82, 1.805, 0.265, 0.26, 1.815, 1.81, 0.265, 0.25,
 1.82, 1.805, 0.265, 0.255, 1.82, 1.81, 0.265, 0.255, 41.65, 0.995, 0.505,
 0.995, 1.005, 0.495, 1., 0.495, 0.495, 0.985, 0.995, 0.985, 0.99, 0.98, 1.,
 0.985, 1.005, 0.5, 0.505, 0.5, 0.505, 0.5, 0.51, 0.495, 0.5, 0.995, 0.505,
 0.49, 1., 0.5, 0.495, 0.99, 0.995, 0.995, 0.505, 0.495, 0.51, 0.495, 0.505,
 0.485, 1.005, 0.495, 0.495, 0.98, 0.99, 0.99, 0.505, 0.49, 0.995, 0.985,
 1.005, 0.5, 0.5, 0.5, 0.5, 0.995, 0.505, 0.485, 1.005, 0.495, 0.495, 0.985,
 1., 0.495, 0.51, 0.495, 0.5, 0.985, 0.99, 0.995, 0.51, 0.485, 0.995, 0.985,
 0.99, 0.99, 0.99, 0.985, 0.99, 0.985, 0.995, 0.99, 0.995, 0.98, 0.99, 0.985,
 1.01, 0.495, 0.51, 0.495, 0.5, 0.995, 0.505, 0.485, 0.99, 0.99, 0.505, 0.5,
 0.505, 0.49, 1., 0.5, 0.505, 0.995, 0.505, 0.995, 1.01, 0.495, 1.005, 0.99,
 0.51, 0.995, 1., 0.495, 1.015, 0.995, 0.505, 0.995, 1., 0.5, 1., 0.5, 0.495,
 0.98, 0.99, 0.985, 0.995, 0.99, 0.51, 0.485, 1.01, 0.495, 0.5, 0.985, 1.005,
 0.5, 0.495, 0.985, 1., 0.985, 0.99, 0.985, 0.99, 0.98, 0.995, 0.995, 0.505,
 0.495, 0.505, 0.49, 0.51, 0.49, 0.995, 0.985, 0.995, 0.985, 0.995, 0.99,
 0.995, 0.985, 0.99, 0.98, 0.995, 1., 0.505, 0.49, 0.995, 0.98, 0.995, 0.99,
 0.505, 0.5, 0.505, 0.49, 1.005, 0.495, 0.49, 0.985, 1., 0.995, 1.005, 0.495,
 1.005, 0.995, 0.51, 0.995, 1.005, 0.495, 1.005, 1., 0.505, 0.975, 14.965,
 0.995, 0.51, 0.99, 1., 0.5, 1.005, 0.495, 0.5, 0.985, 0.995, 0.985, 0.995,
 0.99, 0.505, 0.49, 1.005, 0.495, 0.49, 0.985, 1., 0.495, 0.5, 0.98, 0.995,
 0.985, 0.995, 0.985, 0.98, 0.995, 0.99, 0.51, 0.5, 0.505, 0.495,
 0.505, 0.485, 0.99, 0.985, 0.99, 0.985, 0.995, 0.985, 0.99, 0.985, 0.995,
 0.98, 0.995, 0.99, 0.505, 0.485, 0.995, 0.985, 0.995, 0.995, 0.995, 0.505, 0.495,
 0.51, 0.485, 1.01, 0.495, 0.495, 0.985, 0.995, 0.99, 1.005, 0.495, 1.005, 1.,
 0.505, 0.995, 1., 0.5, 1.005, 0.995, 0.51, 0.975, 14.965, 0.995, 0.505, 1.,
 1.005, 0.5, 1., 0.495, 0.495, 0.985, 0.99, 0.985, 0.995, 0.99, 0.505, 0.485,
 1.005, 0.49, 0.5, 0.985, 1., 0.495, 0.495, 0.99, 0.995, 0.98, 0.995, 0.985,
 0.995, 0.985, 0.995, 0.99, 0.505, 0.495, 0.505, 0.49, 0.505, 0.49, 0.995,
 0.985, 0.99, 0.985, 0.995, 0.985, 0.995, 0.985, 0.99, 0.985, 0.995, 0.995,
 0.505, 0.49, 0.995, 0.985, 0.99, 0.99, 0.51, 0.495, 0.505, 0.49, 1., 0.5,
 0.495, 0.985, 0.995, 1., 1., 0.495, 1.005, 0.995, 0.505, 1., 1.005, 0.495,
 1.01, 0.995, 0.505, 0.98, 54.79, 1.815, 0.26, 0.255, 1.825, 1.805, 0.27,
 0.255, 1.815, 1.81, 0.26, 0.255, 1.825, 1.805, 0.265, 0.255, 1.815, 0.255,
 1.815, 0.255, 1.82, 1.805, 0.265, 0.26, 1.815, 1.81, 0.265, 0.25, 1.82, 1.81,
 0.265, 0.26, 1.825, 1.805, 0.265, 0.255, 16.74, 1.81, 0.265, 0.25, 1.82,
 1.815, 0.26, 0.255, 1.825, 1.805, 0.265, 0.255, 1.815, 1.81, 0.26, 0.255,
 1.82, 0.26, 1.815, 0.255, 1.815, 1.81, 0.265, 0.255, 1.82, 1.805, 0.265,
 0.26, 1.815, 1.81, 0.27, 0.255, 1.815, 1.81, 0.26, 0.255, 41.655, 0.995,
 0.51, 0.995, 1.005, 0.495, 1.005, 0.5, 0.49, 0.985, 0.995, 0.985, 0.99, 0.99,
 0.995, 0.98, 1., 0.495, 0.505, 0.5, 0.505, 0.495, 0.505, 0.49, 0.5, 0.99,
 0.51, 0.48, 1.005, 0.495, 0.495, 0.985, 0.99, 0.995, 0.51, 0.49, 0.51, 0.5,

0.505, 0.485, 1., 0.495, 0.495, 0.985, 0.99, 0.995, 0.505, 0.49, 0.995, 0.98,
 1.01, 0.495, 0.505, 0.495, 0.5, 0.995, 0.505, 0.485, 1.005, 0.5, 0.49, 0.99,
 1., 0.49, 0.51, 0.5, 0.495, 0.98, 0.995, 0.995, 0.505, 0.49, 0.995, 0.98,
 0.995, 0.99, 0.995, 0.985, 0.995, 0.985, 0.995, 0.99, 0.995, 0.98, 0.99,
 0.98, 1.005, 0.495, 0.51, 0.495, 0.5, 0.99, 0.51, 0.48, 0.995, 0.995, 0.5,
 0.5, 0.51, 0.485, 1.005, 0.5, 0.51, 0.995, 0.5, 0.995, 1.01, 0.495, 1.005,
 1., 0.505, 0.995, 1.01, 0.495, 1.015, 1., 0.5, 0.995, 1.005, 0.495, 1., 0.5,
 0.495, 0.99, 0.995, 0.985, 0.995, 0.995, 0.51, 0.49, 1.005, 0.495, 0.495,
 0.985, 1., 0.5, 0.495, 0.98, 0.995, 0.98, 0.995, 0.98, 0.995, 0.985, 0.995,
 0.995, 0.51, 0.5, 0.505, 0.5, 0.505, 0.49, 0.995, 0.985, 0.995, 0.985, 0.995,
 0.985, 0.99, 0.98, 0.995, 0.985, 0.99, 0.995, 0.505, 0.485, 0.995, 0.985, 0.99,
 0.99, 0.51, 0.5, 0.505, 0.485, 1.005, 0.495, 0.495, 0.985, 0.995, 0.995, 1.,
 0.49, 1.005, 0.995, 0.505, 0.995, 1.005, 0.5, 1., 0.995, 0.5, 0.975, 14.97,
 0.995, 0.505, 1., 1.005, 0.495, 1.005, 0.5, 0.495, 0.99, 0.995, 0.98, 0.995,
 0.995, 0.51, 0.49, 1.005, 0.495, 0.5, 0.985, 1.005, 0.5, 0.49, 0.985, 0.995,
 0.98, 0.995, 0.98, 0.99, 0.985, 0.995, 0.99, 0.51, 0.495, 0.51, 0.49, 0.51,
 0.49, 0.99, 0.985, 0.99, 0.985, 0.995, 0.985, 0.995, 0.98, 0.99, 0.985, 0.99,
 0.995, 0.505, 0.49, 0.99, 0.98, 0.995, 0.99, 0.505, 0.5, 0.5, 0.485, 1., 0.5,
 0.495, 0.985, 0.995, 0.995, 1.005, 0.5, 1.005, 0.995, 0.51, 0.995, 1., 0.5,
 1.005, 0.99, 0.51, 0.975, 14.96, 0.995, 0.505, 0.99, 1.005, 0.5, 1.005, 0.5,
 0.495, 0.985, 0.995, 0.985, 0.995, 0.51, 0.49, 1.005, 0.495, 0.5, 0.985, 1.,
 0.985, 1.005, 0.5, 0.495, 0.985, 0.995, 0.985, 0.99, 0.995, 0.985, 0.995, 0.99,
 0.995, 0.51, 0.5, 0.505, 0.495, 0.505, 0.485, 0.99, 0.985, 0.995, 0.985,
 0.995, 0.985, 0.99, 0.98, 0.99, 0.985, 0.99, 0.995, 0.505, 0.485, 0.995, 0.995,
 0.985, 0.995, 0.99, 0.51, 0.5, 0.505, 0.485, 1.005, 0.495, 0.495, 0.985, 0.995,
 0.995, 0.995, 1., 0.49, 1.005, 0.995, 0.505, 0.995, 1., 0.5, 1.005, 0.995,
 0.51, 0.98, 54.79, 1.81, 0.265, 0.26, 1.815, 1.81, 0.265, 0.255, 1.825, 1.805,
 0.265, 0.255, 1.815, 1.81, 0.27, 0.255, 1.82, 0.255, 1.82, 0.255, 1.815, 1.81,
 0.265, 0.255, 1.82, 1.805, 0.27, 0.255, 1.82, 1.805, 0.265, 0.26, 1.815,
 1.81, 0.265, 0.25, 16.75, 1.81, 0.265, 0.255, 1.82, 1.81, 0.265, 0.255, 1.82,
 1.805, 0.265, 0.255, 1.82, 1.81, 0.26, 0.255, 1.82, 0.25, 1.82, 0.255, 1.815,
 1.81, 0.265, 0.255, 1.82, 1.805, 0.265, 0.255, 1.815, 1.81, 0.27, 0.255, 1.82,
 1.81, 0.265, 0.255, 41.655, 0.995, 0.51, 0.995, 1.005, 0.5, 1., 0.495, 0.5,
 0.985, 0.99, 0.985, 0.99, 0.985, 0.995, 0.985, 1., 0.5, 0.505, 0.495, 0.5,
 0.495, 0.51, 0.495, 0.5, 0.995, 0.5, 0.485, 1.005, 0.495, 0.49, 0.985, 0.99,
 0.995, 0.505, 0.5, 0.51, 0.495, 0.5, 0.485, 1.005, 0.495, 0.5, 0.985, 0.99,
 0.995, 0.505, 0.485, 0.99, 0.98, 1.005, 0.495, 0.505, 0.495, 0.5, 0.995,
 0.51, 0.485, 1.01, 0.495, 0.5, 0.985, 1., 0.495, 0.505, 0.5, 0.495, 0.985,
 0.995, 0.995, 0.505, 0.485, 0.995, 0.985, 0.995, 0.98, 0.995, 0.985, 0.99,
 0.99, 0.99, 0.98, 0.995, 0.985, 1., 0.985, 1.005, 0.5, 0.505, 0.5, 0.49,
 1., 0.505, 0.49, 0.995, 0.995, 0.51, 0.49, 0.51, 0.485, 1.01, 0.495, 0.51,
 0.995, 0.505, 0.995, 1., 0.495, 1.005, 0.99, 0.505, 1., 1., 0.495, 1.015,
 0.995, 0.505, 0.995, 1.005, 0.495, 1., 0.5, 0.5, 0.985, 0.995, 0.985, 0.995,
 0.995, 0.505, 0.49, 1.005, 0.5, 0.495, 0.98, 1., 0.495, 0.495, 0.985, 0.995,
 0.985, 0.99, 0.985, 1., 0.98, 0.995, 0.995, 0.51, 0.495, 0.51, 0.5, 0.505,
 0.49, 0.995, 0.985, 0.995, 0.99, 0.995, 0.985, 0.995, 0.985, 0.99, 0.985,
 0.99, 0.995, 0.505, 0.485, 0.995, 0.985, 0.99, 0.995, 0.51, 0.495, 0.505,
 0.485, 1.005, 0.5, 0.5, 0.985, 0.995, 0.995, 1.005, 0.5, 1.005, 0.995, 0.51,
 0.995, 1.005, 0.5, 1.005, 0.995, 0.51, 0.975, 14.965, 0.995, 0.505, 0.995,

Difference in run lengths of opposite polarities in sampling steps

```
In[29]:= ListPlot[MapThread[(#1 - #2)/timestep &, rlbits], Frame -> True, PlotRange -> All]
```



Run length encoded data packets separated by run length >2 bits

```
In[30]:= rlrungs = SplitBy[#, # > 2 &] & /@ rlbits
```

0.49, 1.005, 0.49, 0.495, 0.995, 0.51, 0.48, 0.99, 0.985, 1., 0.495, 0.505,
 0.5, 0.505, 0.495, 0.495, 0.995, 0.505, 0.49, 0.995, 0.98, 1.005, 0.5, 0.49,
 0.98, 0.995, 0.99, 0.51, 0.49, 0.51, 0.485, 1.01, 0.495, 0.495, 0.99, 0.505,
 0.485, 0.995, 0.99, 0.51, 0.5, 0.505, 0.49, 0.99, 0.985, 1.01, 0.495, 0.5,
 0.985, 0.995, 0.985, 0.995, 0.985, 0.995, 0.98, 0.995, 0.985, 0.995, 0.985,
 0.99, 0.98, 0.995, 0.99, 0.505, 0.5, 0.505, 0.49, 1., 0.5, 0.49, 0.985, 1.01,
 0.49, 0.51, 0.5, 0.495, 0.99, 0.51, 0.49, 1.005, 0.495, 1.01, 0.995, 0.505,
 0.99, 1.01, 0.495, 1., 0.995, 0.505, 1.005, 1.01, 0.49, 1.005, 0.995, 0.505,
 0.99, 0.505, 0.485, 1., 0.98, 0.995, 0.98, 1.005, 0.5, 0.49, 0.995, 0.505,
 0.49, 0.995, 0.99, 0.51, 0.485, 0.99, 0.98, 0.99, 0.985, 0.995, 0.98, 0.99,
 0.98, 1.005, 0.495, 0.505, 0.495, 0.51, 0.495, 0.5, 0.98, 0.99, 0.98, 0.995,
 0.985, 0.99, 0.985, 0.995, 0.98, 0.995, 0.98, 1., 0.495, 0.495, 0.98, 1., 0.985,
 1.005, 0.495, 0.505, 0.5, 0.495, 0.995, 0.51, 0.49, 0.995, 0.98, 1., 0.995,
 0.51, 0.99, 1., 0.5, 1.005, 0.99, 0.51, 0.995, 1.005, 0.5, 0.985}, {14.95},
 {1.005, 0.495, 1.01, 0.995, 0.5, 0.995, 0.51, 0.485, 0.995, 0.98, 0.995, 0.98, 1.005,
 0.495, 0.5, 0.995, 0.505, 0.485, 0.995, 0.99, 0.51, 0.485, 0.995, 0.985, 0.99, 0.98,
 0.995, 0.985, 0.995, 0.985, 1., 0.5, 0.51, 0.495, 0.51, 0.495, 0.5, 0.985, 0.995,
 0.985, 0.99, 0.985, 1., 0.98, 0.99, 0.98, 0.99, 0.985, 1.005, 0.49, 0.5, 0.985,
 0.99, 0.985, 1.01, 0.495, 0.505, 0.495, 0.5, 0.99, 0.505, 0.485, 0.995, 0.985,
 1.005, 0.99, 0.5, 0.995, 1.005, 0.495, 1., 0.995, 0.505, 0.995, 1.01, 0.495, 0.99},
 {14.955}, {1.005, 0.5, 1.005, 0.995, 0.51, 0.995, 0.505, 0.49, 0.99, 0.98,
 0.99, 0.985, 1.005, 0.495, 0.495, 0.99, 0.5, 0.49, 0.995, 0.99, 0.51, 0.49,
 0.995, 0.98, 0.99, 0.98, 0.995, 0.98, 0.995, 0.985, 1.005, 0.5, 0.505, 0.5,
 0.505, 0.5, 0.495, 0.985, 0.99, 0.98, 0.995, 0.985, 0.995, 0.985, 0.99,
 0.98, 0.995, 0.985, 1.005, 0.495, 0.5, 0.985, 0.995, 0.985, 1.005, 0.495,
 0.51, 0.495, 0.5, 0.99, 0.505, 0.485, 0.995, 0.985, 1.005, 0.995, 0.51,
 0.995, 1., 0.49, 1.01, 0.995, 0.505, 0.995, 1.005, 0.495, 0.99}, {54.78},
 {1.815, 0.255, 0.265, 1.805, 1.82, 0.25, 0.265, 1.81, 1.82, 0.255, 0.265,
 1.81, 1.82, 0.255, 0.265, 1.81, 0.265, 1.81, 0.27, 1.81, 1.815, 0.255, 0.265,
 1.805, 1.82, 0.26, 0.26, 1.81, 1.82, 0.255, 0.27, 1.805, 1.825, 0.255, 0.26},
 {16.735}, {1.815, 0.255, 0.265, 1.805, 1.82, 0.26, 0.265, 1.81, 1.82, 0.255,
 0.265, 1.805, 1.82, 0.25, 0.27, 1.81, 0.26, 1.81, 0.27, 1.805, 1.82, 0.25, 0.27,
 1.81, 1.815, 0.255, 0.265, 1.805, 1.82, 0.25, 0.265, 1.815, 1.815, 0.26, 0.265},
 {41.635}, {1.005, 0.5, 1., 0.995, 0.505, 0.995, 0.5, 0.49, 0.99, 0.985, 0.995,
 0.98, 0.995, 0.985, 0.995, 0.995, 0.51, 0.5, 0.505, 0.5, 0.505, 0.5, 0.51,
 0.485, 1.005, 0.5, 0.495, 0.99, 0.51, 0.49, 0.995, 0.985, 1.005, 0.5, 0.505,
 0.495, 0.505, 0.495, 0.5, 0.995, 0.5, 0.485, 0.99, 0.98, 1., 0.495, 0.5, 0.985,
 0.995, 0.995, 0.51, 0.49, 0.51, 0.49, 1.005, 0.495, 0.495, 0.995, 0.505, 0.485,
 0.995, 0.995, 0.5, 0.51, 0.485, 0.995, 0.985, 1.005, 0.495, 0.495, 0.985,
 0.995, 0.98, 1., 0.985, 0.99, 0.98, 0.995, 0.99, 0.995, 0.985, 0.99, 0.98,
 1., 0.995, 0.505, 0.5, 0.505, 0.49, 1.005, 0.5, 0.49, 0.985, 1., 0.49, 0.51,
 0.5, 0.495, 0.99, 0.51, 0.5, 1.005, 0.49, 1.01, 0.995, 0.505, 0.995, 1.005,
 0.495, 1.005, 0.99, 0.505, 1.01, 1.005, 0.49, 1.01, 0.99, 0.505, 0.99, 0.51,
 0.485, 0.995, 0.98, 0.99, 0.985, 1.005, 0.495, 0.5, 0.995, 0.505, 0.49, 0.995,
 0.995, 0.51, 0.49, 0.995, 0.985, 0.995, 0.98, 0.995, 0.98, 0.995, 0.98, 1.005,
 0.495, 0.505, 0.495, 0.505, 0.495, 0.5, 0.985, 0.995, 0.985, 1., 0.985, 0.995,
 0.985, 0.995, 0.98, 0.995, 0.985, 1.005, 0.495, 0.5, 0.985, 0.995, 0.98, 1.005,
 0.495, 0.505, 0.495, 0.5, 0.995, 0.505, 0.48, 1., 0.985, 1.005, 0.995, 0.51,
 0.99, 1.005, 0.5, 1.005, 0.995, 0.51, 0.995, 1.005, 0.495, 0.99}, {14.955},

```

{1.005, 0.495, 1., 0.995, 0.505, 0.995, 0.51, 0.485, 1., 0.985, 0.995, 0.98, 1.,
0.5, 0.495, 0.995, 0.505, 0.48, 1., 0.99, 0.5, 0.49, 0.99, 0.985, 0.995, 0.985,
0.995, 0.985, 0.99, 0.985, 1.005, 0.495, 0.51, 0.495, 0.505, 0.495, 0.495, 0.98,
0.995, 0.98, 0.995, 0.985, 0.995, 0.98, 1., 0.985, 0.99, 0.98, 1., 0.5, 0.49, 0.985,
0.995, 0.985, 1.005, 0.5, 0.505, 0.495, 0.5, 0.995, 0.505, 0.485, 0.995, 0.985,
1.005, 0.995, 0.5, 1., 1.005, 0.5, 1., 0.995, 0.505, 1., 1.005, 0.495, 0.985},
{14.955}, {1.005, 0.5, 1.005, 0.995, 0.51, 0.995, 0.5, 0.485, 0.995, 0.98, 1.,
0.98, 1.005, 0.49, 0.5, 0.99, 0.505, 0.485, 0.995, 0.99, 0.505, 0.49, 0.995, 0.985,
0.99, 0.985, 1., 0.985, 0.995, 0.98, 1., 0.5, 0.5, 0.495, 0.505, 0.49, 0.5, 0.985,
0.995, 0.98, 0.995, 0.99, 0.995, 0.985, 0.99, 0.98, 0.995, 0.985, 1.005, 0.495, 0.5,
0.985, 0.995, 0.98, 1.005, 0.495, 0.505, 0.5, 0.495, 0.99, 0.51, 0.485, 1., 0.985,
1.005, 0.99, 0.505, 0.995, 1.005, 0.5, 1.005, 0.995, 0.51, 0.995, 1.005, 0.5, 0.99},
{54.775}, {1.825, 0.255, 0.265, 1.81, 1.82, 0.255, 0.265, 1.805, 1.82, 0.255, 0.265,
1.81, 1.815, 0.255, 0.265, 1.805, 0.265, 1.81, 0.265, 1.805, 1.82, 0.25, 0.27, 1.81,
1.815, 0.255, 0.265, 1.805, 1.82, 0.26, 0.27, 1.81, 1.815, 0.255, 0.265}, {16.735},
{1.815, 0.255, 0.265, 1.81, 1.82, 0.255, 0.265, 1.81, 1.815, 0.255, 0.265,
1.805, 1.82, 0.255, 0.26, 1.81, 0.27, 1.805, 0.265, 1.81, 1.815, 0.255, 0.265,
1.81, 1.82, 0.255, 0.265, 1.81, 1.815, 0.26, 0.265, 1.805, 1.82, 0.25, 0.265},
{41.645}, {1.005, 0.5, 1.005, 0.995, 0.51, 0.995, 0.505, 0.485, 0.99, 0.985,
0.995, 0.985, 0.995, 0.985, 0.99, 0.99, 0.505, 0.5, 0.505, 0.495, 0.505, 0.495,
0.505, 0.485, 1., 0.5, 0.49, 0.995, 0.505, 0.49, 0.99, 0.98, 1.005, 0.5, 0.505,
0.495, 0.51, 0.495, 0.495, 0.99, 0.505, 0.49, 0.995, 0.98, 1., 0.5, 0.495,
0.985, 0.995, 0.995, 0.995, 0.505, 0.495, 0.51, 0.485, 1.005, 0.495, 0.495, 1., 0.505,
0.485, 0.995, 0.99, 0.505, 0.495, 0.51, 0.485, 0.99, 0.985, 1.005, 0.495,
0.5, 0.985, 0.99, 0.99, 0.995, 0.985, 0.995, 0.985, 1., 0.985, 0.995, 0.985,
0.99, 0.98, 0.995, 0.995, 0.5, 0.5, 0.51, 0.485, 1., 0.5, 0.495, 0.98, 1.005,
0.495, 0.505, 0.5, 0.495, 1., 0.505, 0.5, 1.005, 0.495, 1.005, 0.995, 0.505,
1., 1.005, 0.495, 1.01, 0.995, 0.51, 1.005, 1.005, 0.495, 1.005, 0.99, 0.51,
0.99, 0.505, 0.49, 0.995, 0.985, 0.995, 0.99, 1.005, 0.495, 0.495, 0.5, 0.995, 0.505,
0.485, 0.995, 0.99, 0.51, 0.49, 0.99, 0.98, 0.995, 0.98, 0.995, 0.98, 0.995,
0.985, 1.01, 0.495, 0.51, 0.495, 0.51, 0.5, 0.495, 0.99, 0.995, 0.985, 0.99,
0.985, 0.995, 0.98, 0.995, 0.98, 0.995, 0.98, 1.005, 0.495, 0.495, 0.985, 0.995,
0.98, 1.005, 0.495, 0.51, 0.495, 0.5, 0.99, 0.505, 0.485, 1., 0.98, 1.005, 0.99,
0.505, 0.99, 1.005, 0.5, 1., 0.995, 0.51, 0.995, 1., 0.49, 0.99}, {14.955},
{1.005, 0.5, 1.005, 0.995, 0.51, 0.995, 0.505, 0.49, 0.995, 0.985, 0.99, 0.985,
1.01, 0.495, 0.5, 0.995, 0.51, 0.485, 1., 0.995, 0.505, 0.485, 0.995, 0.98, 0.99,
0.985, 0.99, 0.985, 0.99, 0.985, 1., 0.5, 0.51, 0.495, 0.505, 0.495, 0.5, 0.985,
0.99, 0.98, 0.995, 0.985, 0.995, 0.985, 0.99, 0.98, 0.995, 0.98, 1.005, 0.5, 0.495,
0.98, 0.99, 0.985, 1.005, 0.49, 0.51, 0.49, 0.495, 0.995, 0.505, 0.49, 0.995, 0.98,
1.005, 0.995, 0.51, 0.995, 1.01, 0.495, 1.005, 0.99, 0.51, 0.995, 1., 0.5, 0.985},
{14.95}, {1.01, 0.49, 1.005, 0.995, 0.505, 0.995, 0.51, 0.49, 0.995, 0.98, 0.995,
0.985, 1.01, 0.495, 0.5, 0.995, 0.51, 0.485, 0.995, 1., 0.505, 0.49, 0.99, 0.985,
0.995, 0.98, 0.995, 0.98, 0.995, 1.01, 0.495, 0.51, 0.495, 0.51, 0.49, 0.495,
0.98, 0.995, 0.99, 0.99, 0.985, 0.995, 0.985, 0.985, 0.99, 0.98, 0.995, 0.98, 1.005, 0.495,
0.98, 0.995, 0.99, 0.99, 0.985, 0.995, 0.985, 0.985, 0.99, 0.98, 1.005, 0.495,
0.5, 0.985, 0.99, 0.985, 1.005, 0.495, 0.51, 0.495, 0.5, 0.99, 0.505, 0.485, 1.,
0.985, 1., 0.99, 0.505, 0.99, 1.005, 0.5, 1., 0.99, 0.51, 0.995, 1.005, 0.5, 0.5, 0.99},
{54.785}, {1.815, 0.255, 0.27, 1.805, 1.82, 0.255, 0.27, 1.81, 1.815, 0.255, 0.265,
1.805, 1.82, 0.26, 0.265, 1.81, 0.265, 1.81, 1.815, 0.255, 0.265}, {1.81, 1.82, 0.255, 0.265, 1.81, 1.815, 0.255, 0.27, 1.805, 1.82, 0.255, 0.265},

```

```

{16.735}, {1.82, 0.255, 0.265, 1.81, 1.82, 0.26, 0.26, 1.81, 1.815, 0.255, 0.265,
1.81, 1.82, 0.255, 0.26, 1.81, 0.265, 1.805, 0.265, 1.805, 1.82, 0.255, 0.27,
1.805, 1.82, 0.25, 0.265, 1.81, 1.815, 0.26, 0.265, 1.81, 1.82, 0.255, 0.265},
{41.645}, {1.005, 0.5, 1.005, 0.995, 0.51, 0.995, 0.505, 0.485, 0.995, 0.98,
0.995, 0.98, 0.995, 0.985, 0.995, 0.99, 0.51, 0.495, 0.505, 0.49, 0.505,
0.5, 0.505, 0.49, 1.005, 0.49, 0.495, 0.995, 0.505, 0.48, 1., 0.98, 1., 0.5,
0.505, 0.5, 0.505, 0.495, 0.49, 0.995, 0.505, 0.49, 0.995, 0.98, 1.005, 0.5,
0.49, 0.98, 0.995, 0.99, 0.505, 0.495, 0.51, 0.485, 1.01, 0.495, 0.5, 0.995,
0.505, 0.49, 0.995, 0.99, 0.505, 0.5, 0.505, 0.49, 0.995, 0.98, 1.01, 0.495,
0.495, 0.985, 0.995, 0.98, 0.99, 0.985, 0.995, 0.985, 0.995, 0.98, 0.99,
0.99, 0.995, 0.985, 0.995, 0.995, 0.51, 0.5, 0.505, 0.485, 1.005, 0.5, 0.495,
0.985, 1.005, 0.5, 0.505, 0.495, 0.5, 0.995, 0.51, 0.495, 1.005, 0.5, 1.,
0.99, 0.505, 0.995, 1., 0.5, 1.005, 0.99, 0.505, 1.005, 1.005, 0.5, 1., 0.995,
0.51, 0.99, 0.51, 0.485, 0.995, 0.985, 0.995, 0.985, 1.01, 0.49, 0.5, 0.995,
0.51, 0.485, 0.99, 0.995, 0.5, 0.49, 0.99, 0.985, 0.995, 0.985, 0.995, 0.985,
0.99, 0.985, 1.005, 0.5, 0.51, 0.495, 0.51, 0.495, 0.5, 0.985, 0.995, 0.99,
0.995, 0.985, 0.995, 0.985, 0.995, 0.985, 0.995, 0.98, 1., 0.5, 0.49, 0.99,
0.99, 0.985, 1., 0.5, 0.505, 0.5, 0.49, 1., 0.505, 0.49, 0.995, 0.985, 1.005,
1., 0.505, 0.995, 1.005, 0.5, 1.005, 0.995, 0.51, 0.995, 1.005, 0.5, 0.99}, ,
{14.95}, {1.005, 0.5, 1.005, 0.99, 0.505, 0.995, 0.505, 0.49, 0.995, 0.98,
0.995, 0.985, 1.01, 0.495, 0.5, 0.99, 0.51, 0.485, 0.995, 0.995, 0.505,
0.49, 0.995, 0.985, 0.995, 0.98, 1., 0.98, 0.99, 0.98, 1.005, 0.495, 0.505,
0.495, 0.505, 0.495, 0.5, 0.985, 0.995, 0.985, 1., 0.985, 0.995, 0.985,
0.995, 0.98, 0.99, 0.985, 1., 0.495, 0.495, 0.985, 0.995, 0.98, 1.01, 0.495,
0.505, 0.495, 0.5, 0.995, 0.505, 0.485, 0.995, 0.985, 1.005, 0.99, 0.51,
0.995, 1., 0.49, 1.01, 0.995, 0.505, 0.99, 1.005, 0.495, 0.99}, , {14.955},
{1.01, 0.495, 1.005, 0.995, 0.505, 0.995, 0.51, 0.485, 0.99, 0.985, 1., 0.98, 1.005,
0.5, 0.49, 0.995, 0.51, 0.485, 0.995, 0.99, 0.51, 0.485, 0.99, 0.99, 0.995, 0.98,
0.99, 0.985, 0.99, 0.985, 1.005, 0.5, 0.505, 0.5, 0.505, 0.5, 0.495, 0.985, 0.995,
0.985, 0.995, 0.985, 0.995, 0.98, 0.995, 0.995, 0.98, 0.995, 0.985, 1.005, 0.5, 0.495,
0.98, 0.995, 0.98, 1.005, 0.5, 0.505, 0.5, 0.495, 0.995, 0.995, 0.505, 0.485, 0.99,
0.985, 1.005, 0.5, 0.505, 0.485, 0.995, 0.985, 1.005, 0.99, 0.51, 0.995, 1., 0.49,
0.99, 0.505, 0.995, 1.01, 0.495, 1.005, 0.995, 0.5, 1., 1.005, 0.49, 0.99}, ,
{54.78}, {1.815, 0.255, 0.265, 1.805, 1.82, 0.26, 0.265, 1.81, 1.82, 0.25, 0.265,
1.81, 1.815, 0.26, 0.265, 1.805, 0.27, 1.805, 0.265, 1.81, 1.815, 0.255, 0.27,
1.805, 1.82, 0.255, 0.26, 1.81, 1.815, 0.255, 0.27, 1.81, 1.82, 0.255, 0.265}, ,
{16.74}, {1.82, 0.255, 0.265, 1.805, 1.82, 0.25, 0.27, 1.81, 1.815, 0.255,
0.265, 1.81, 1.82, 0.25, 0.265, 1.81, 0.265, 1.81, 0.265, 1.81, 1.82, 0.25, 0.265,
1.805, 1.82, 0.255, 0.26, 1.81, 1.82, 0.255, 0.265, 1.81, 1.82, 0.255, 0.26}, ,
{41.645}, {1.01, 0.495, 1.005, 0.995, 0.505, 0.995, 0.51, 0.485, 0.99, 0.985,
0.995, 0.985, 0.99, 0.98, 0.995, 0.99, 0.505, 0.495, 0.505, 0.495, 0.51, 0.495,
0.51, 0.485, 1., 0.495, 0.49, 0.99, 0.505, 0.49, 0.995, 0.98, 1.005, 0.495,
0.51, 0.495, 0.51, 0.49, 0.495, 0.995, 0.51, 0.48, 0.99, 0.98, 1.005, 0.495,
0.495, 0.985, 0.99, 0.995, 0.51, 0.5, 0.505, 0.49, 1., 0.5, 0.495, 0.995, 0.51,
0.49, 0.99, 0.995, 0.51, 0.495, 0.505, 0.485, 0.995, 0.985, 0.995, 1.005, 0.495,
0.5, 0.98, 0.995, 0.98, 0.995, 0.985, 0.995, 0.98, 1., 0.985, 0.995, 0.98,
0.995, 0.985, 0.99, 0.995, 0.5, 0.5, 0.505, 0.485, 1., 0.495, 0.5, 0.98, 1.005,
0.495, 0.505, 0.5, 0.495, 0.995, 0.505, 0.495, 1.005, 0.495, 1., 0.995, 0.505,
0.99, 1.005, 0.495, 1.005, 1., 0.505, 1., 1.01, 0.495, 1., 0.995, 0.505, 0.99,
0.51, 0.485, 1., 0.985, 0.995, 0.98, 1.005, 0.495, 0.495, 0.5, 0.995, 0.505, 0.49,
0.495, 0.985, 0.99, 0.995, 0.51, 0.5, 0.505, 0.495, 1., 0.495, 0.5, 0.98, 1.005,
0.495, 0.505, 0.5, 0.495, 0.995, 0.505, 0.495, 1.005, 0.495, 1., 0.995, 0.505,
0.99, 1.005, 0.495, 1.005, 1., 0.505, 1., 1.01, 0.495, 1., 0.995, 0.505, 0.99,
0.51, 0.485, 1., 0.985, 0.995, 0.98, 1.005, 0.495, 0.495, 0.5, 0.995, 0.505, 0.49,
```

0.99, 0.99, 0.505, 0.485, 1., 0.98, 0.995, 0.985, 0.995, 0.985, 0.995, 0.98,
 1., 0.495, 0.51, 0.495, 0.505, 0.495, 0.495, 0.98, 0.995, 0.98, 0.995, 0.98,
 0.99, 0.985, 0.99, 0.985, 0.99, 0.98, 1.005, 0.5, 0.495, 0.985, 0.995, 0.985,
 1.005, 0.5, 0.51, 0.495, 0.5, 0.995, 0.505, 0.485, 0.995, 0.985, 1.005, 0.995,
 0.505, 0.995, 1.005, 0.49, 1.005, 0.995, 0.51, 0.99, 1., 0.5, 0.99}, {14.955},
 {1.005, 0.5, 1.005, 0.995, 0.51, 0.99, 0.505, 0.49, 0.995, 0.985, 0.995, 0.98,
 1., 0.5, 0.495, 0.995, 0.505, 0.49, 0.995, 0.995, 0.51, 0.485, 0.99, 0.99, 0.995,
 0.98, 0.995, 0.985, 1., 0.98, 1.005, 0.495, 0.505, 0.5, 0.51, 0.495, 0.5, 0.985,
 0.99, 0.985, 0.99, 0.985, 0.995, 0.985, 0.99, 0.985, 0.995, 0.985, 1., 0.495, 0.495,
 0.98, 0.995, 0.985, 1.005, 0.495, 0.51, 0.495, 0.5, 0.99, 0.51, 0.48, 0.99, 0.985,
 1.005, 0.995, 0.505, 0.995, 1., 0.495, 1.005, 0.995, 0.505, 0.99, 1.005, 0.295}},
 {{0.27, 1.005, 0.5, 0.49, 0.985, 1., 0.495, 0.51, 0.495, 0.5, 0.98, 0.99, 0.995,
 0.51, 0.49, 0.995, 0.985, 0.99, 0.98, 0.99, 0.98, 0.995, 0.985, 0.99, 0.98,
 0.995, 0.99, 0.995, 0.985, 1.005, 0.5, 0.5, 0.5, 0.495, 1., 0.5, 0.49, 0.995,
 0.99, 0.505, 0.495, 0.51, 0.485, 1.01, 0.49, 0.51, 0.995, 0.505, 0.995, 1.,
 0.49, 1.01, 0.995, 0.505, 1., 1.005, 0.495, 1.015, 0.995, 0.505, 0.995, 1.005,
 0.495, 1.005, 0.495, 0.495, 0.985, 0.995, 0.985, 0.99, 0.995, 0.505, 0.485,
 1.005, 0.495, 0.495, 0.985, 1.005, 0.495, 0.5, 0.985, 0.99, 0.98, 0.995, 0.99,
 0.99, 0.985, 0.995, 0.99, 0.51, 0.495, 0.505, 0.5, 0.505, 0.49, 0.99, 0.98, 0.99,
 0.985, 0.995, 0.98, 0.995, 0.985, 0.99, 0.98, 0.995, 0.99, 0.505, 0.485, 0.99,
 0.985, 0.995, 0.995, 0.51, 0.495, 0.505, 0.485, 1.005, 0.495, 0.495, 0.99, 0.99,
 0.995, 1., 0.5, 1.005, 0.99, 0.505, 0.995, 1.005, 0.5, 1.005, 0.995, 0.505, 0.98},
 {14.965}, {0.99, 0.51, 0.99, 1.005, 0.5, 1.005, 0.49, 0.495, 0.98, 0.995,
 0.985, 0.995, 0.995, 0.51, 0.485, 1.005, 0.5, 0.49, 0.985, 1.005, 0.495,
 0.5, 0.98, 0.995, 0.98, 0.995, 0.985, 0.995, 0.98, 0.995, 0.99, 0.505, 0.5,
 0.505, 0.5, 0.505, 0.49, 0.995, 0.985, 0.995, 0.98, 0.995, 0.985, 0.995,
 0.98, 0.995, 0.98, 0.995, 0.995, 0.5, 0.49, 0.995, 0.98, 0.995, 0.995, 0.51,
 0.495, 0.505, 0.49, 1.005, 0.495, 0.5, 0.98, 0.995, 0.995, 1.005, 0.495,
 1.005, 0.995, 0.51, 0.99, 1., 0.495, 1.005, 0.995, 0.51, 0.98}, {14.96},
 {1., 0.5, 0.995, 1., 0.5, 1.005, 0.5, 0.495, 0.98, 0.995, 0.985, 0.995, 0.995,
 0.51, 0.485, 1., 0.5, 0.5, 0.985, 1.005, 0.495, 0.5, 0.985, 0.99, 0.985, 0.995,
 0.985, 0.99, 0.985, 0.99, 0.995, 0.505, 0.5, 0.505, 0.5, 0.5, 0.485, 0.99, 0.985,
 1., 0.985, 0.99, 0.98, 0.995, 0.98, 0.995, 0.985, 0.99, 0.99, 0.51, 0.48, 0.995,
 0.985, 0.995, 0.99, 0.505, 0.5, 0.51, 0.485, 1., 0.495, 0.5, 0.985, 0.995, 0.99,
 1., 0.495, 1.005, 0.995, 0.51, 0.995, 1., 0.5, 1.005, 0.99, 0.505, 0.98}, {54.785},
 {1.81, 0.265, 0.255, 1.815, 1.815, 0.26, 0.255, 1.82, 1.81, 0.27, 0.255, 1.815,
 1.81, 0.265, 0.255, 1.82, 0.255, 1.815, 0.255, 1.825, 1.805, 0.27, 0.255, 1.815,
 1.81, 0.265, 0.25, 1.82, 1.81, 0.265, 0.255, 1.825, 1.805, 0.27, 0.26}, {16.75},
 {1.81, 0.265, 0.255, 1.82, 1.81, 0.265, 0.255, 1.82, 1.81, 0.265, 0.26, 1.815,
 1.815, 0.265, 0.25, 1.825, 0.255, 1.815, 0.255, 1.815, 1.81, 0.265, 0.255,
 1.815, 1.81, 0.265, 0.255, 1.82, 1.805, 0.27, 0.255, 1.815, 1.81, 0.265, 0.25},
 {41.66}, {0.995, 0.505, 0.99, 1.005, 0.5, 1.005, 0.495, 0.495, 0.99, 0.995,
 0.985, 0.99, 0.98, 0.995, 0.99, 1.005, 0.495, 0.51, 0.5, 0.505, 0.5, 0.505,
 0.5, 0.495, 0.995, 0.505, 0.485, 1.005, 0.495, 0.495, 0.98, 0.99, 0.995,
 0.505, 0.495, 0.505, 0.5, 0.505, 0.485, 1.005, 0.495, 0.5, 0.985, 0.99, 0.995,
 0.505, 0.485, 0.99, 0.98, 1.005, 0.495, 0.505, 0.5, 0.495, 0.995, 0.51, 0.48,
 1.005, 0.495, 0.495, 0.985, 1., 0.5, 0.51, 0.495, 0.495, 0.985, 0.995, 0.995,
 0.51, 0.49, 0.995, 0.985, 0.995, 0.985, 0.995, 0.98, 0.995, 0.985, 0.995,
 0.98, 0.995, 0.98, 0.995, 0.985, 1., 0.495, 0.51, 0.495, 0.495, 0.995, 0.505,

1.82, 1.805, 0.265, 0.255, 1.815, 1.81, 0.27, 0.255, 1.82, 1.81, 0.265, 0.255},
 {41.655}, {0.995, 0.51, 0.995, 1.005, 0.5, 1., 0.495, 0.5, 0.985, 0.99, 0.985,
 0.99, 0.985, 0.995, 0.985, 1., 0.5, 0.505, 0.495, 0.5, 0.495, 0.51, 0.495,
 0.5, 0.995, 0.5, 0.485, 1.005, 0.495, 0.49, 0.985, 0.99, 0.995, 0.505, 0.5,
 0.51, 0.495, 0.5, 0.485, 1.005, 0.495, 0.5, 0.985, 0.99, 0.995, 0.505, 0.485,
 0.99, 0.98, 1.005, 0.495, 0.505, 0.495, 0.5, 0.995, 0.51, 0.485, 1.01, 0.495,
 0.5, 0.985, 1., 0.495, 0.505, 0.5, 0.495, 0.985, 0.995, 0.995, 0.505, 0.485,
 0.995, 0.985, 0.995, 0.98, 0.995, 0.985, 0.99, 0.99, 0.99, 0.98, 0.995, 0.985,
 1., 0.985, 1.005, 0.5, 0.505, 0.5, 0.49, 1., 0.505, 0.49, 0.995, 0.995, 0.51,
 0.49, 0.51, 0.485, 1.01, 0.495, 0.51, 0.995, 0.505, 0.995, 1., 0.495, 1.005,
 0.99, 0.505, 1., 1., 0.495, 1.015, 0.995, 0.505, 0.995, 1.005, 0.495, 1.,
 0.5, 0.5, 0.985, 0.995, 0.985, 0.995, 0.995, 0.505, 0.49, 1.005, 0.5, 0.495,
 0.98, 1., 0.495, 0.495, 0.985, 0.995, 0.985, 0.99, 0.985, 1., 0.98, 0.995,
 0.995, 0.51, 0.495, 0.51, 0.5, 0.505, 0.49, 0.995, 0.985, 0.995, 0.99, 0.995,
 0.985, 0.995, 0.985, 0.99, 0.985, 0.99, 0.995, 0.505, 0.485, 0.995, 0.985,
 0.99, 0.995, 0.51, 0.495, 0.505, 0.485, 1.005, 0.5, 0.5, 0.985, 0.995, 0.995,
 1.005, 0.5, 1.005, 0.995, 0.51, 0.995, 1.005, 0.5, 1.005, 0.995, 0.51, 0.975},
{14.965}, {0.995, 0.505, 0.995, 1., 0.495, 1.005, 0.5, 0.495, 0.985, 0.995,
 0.985, 0.995, 1., 0.505, 0.485, 1.005, 0.495, 0.5, 0.985, 1., 0.5, 0.5,
 0.985, 0.995, 0.985, 0.99, 0.985, 0.99, 0.985, 0.99, 0.995, 0.505, 0.495,
 0.505, 0.495, 0.505, 0.49, 0.995, 0.985, 0.995, 0.995, 0.99, 0.995,
 0.985, 0.99, 0.98, 0.99, 0.995, 0.5, 0.49, 0.995, 0.985, 0.99, 0.995,
 0.51, 0.495, 0.505, 0.49, 1.005, 0.495, 0.495, 0.985, 0.99, 0.995, 1.005,
 0.495, 1.005, 0.995, 0.5, 1., 1., 0.5, 1., 0.995, 0.505, 0.98}, {14.965},
{0.995, 0.51, 0.995, 1.005, 0.495, 1.005, 0.495, 0.5, 0.98, 0.995, 0.985,
 0.995, 0.995, 0.505, 0.485, 1.005, 0.5, 0.49, 0.985, 1.005, 0.495, 0.5,
 0.98, 1., 0.98, 0.99, 0.985, 0.995, 0.98, 0.99, 1., 0.505, 0.5, 0.505,
 0.5, 0.51, 0.485, 0.99, 0.99, 0.995, 0.98, 0.995, 0.985, 0.995, 0.985,
 0.99, 0.985, 0.995, 0.995, 0.505, 0.485, 0.995, 0.98, 0.995, 0.995,
 0.505, 0.5, 0.51, 0.485, 1., 0.5, 0.49, 0.985, 0.995, 0.99, 1., 0.495,
 1.005, 1., 0.505, 0.995, 1., 0.495, 1.005, 0.995, 0.505, 0.98}, {54.79},
{1.805, 0.265, 0.255, 1.815, 1.81, 0.265, 0.255, 1.825, 1.805, 0.265, 0.255,
 1.815, 1.81, 0.27, 0.255, 1.815, 0.26, 1.815, 0.255, 1.815, 1.81, 0.265, 0.26,
 1.815, 1.81, 0.26, 0.255, 1.82, 1.805, 0.265, 0.26, 1.82, 1.81, 0.265, 0.255},
{16.75}, {1.81, 0.26, 0.255, 1.82, 1.805, 0.265, 0.26, 1.815, 1.81, 0.265, 0.255,
 1.82, 1.805, 0.265, 0.255, 1.82, 0.255, 1.82, 0.255, 1.82, 1.805, 0.265, 0.255,
 1.815, 1.81, 0.26, 0.255, 1.82, 1.81, 0.265, 0.255, 1.82, 1.81, 0.26, 0.255},
{41.655}, {0.995, 0.51, 0.995, 1.005, 0.495, 1.005, 0.495, 0.5, 0.98, 0.995,
 0.98, 0.995, 0.98, 0.995, 0.98, 1.005, 0.495, 0.505, 0.495, 0.505, 0.495,
 0.51, 0.495, 0.5, 0.99, 0.5, 0.485, 1., 0.495, 0.5, 0.985, 0.99, 0.99, 0.51,
 0.495, 0.51, 0.495, 0.505, 0.485, 1.005, 0.495, 0.495, 0.98, 0.99, 0.99, 0.51,
 0.485, 0.995, 0.98, 1.005, 0.5, 0.51, 0.495, 0.495, 0.995, 0.505, 0.49, 1.005,
 0.5, 0.495, 0.985, 1.005, 0.495, 0.51, 0.495, 0.495, 0.985, 0.995, 0.99,
 0.51, 0.49, 0.99, 0.985, 0.99, 0.98, 0.995, 0.99, 0.99, 0.985, 0.995, 0.985,
 0.995, 0.985, 0.995, 0.98, 1., 0.495, 0.51, 0.495, 0.495, 0.985, 0.995, 0.99,
 1., 0.495, 1.005, 0.99, 0.51, 0.995, 1.005, 0.5, 1.01, 0.995, 0.51, 0.99, 1.,
 0.5, 1., 0.5, 0.495, 0.985, 1., 0.985, 0.99, 0.99, 0.51, 0.49, 1.005, 0.495,
 0.5, 0.98, 1., 0.495, 0.495, 0.985, 0.995, 0.985, 0.98, 0.995, 0.995, 0.985,

```
0.995, 0.99, 0.505, 0.495, 0.51, 0.49, 0.51, 0.48, 0.995, 0.985, 0.99, 0.98,
0.99, 0.985, 0.995, 0.98, 0.99, 0.98, 0.995, 0.995, 0.51, 0.48, 0.995, 0.99,
0.995, 0.995, 0.51, 0.5, 0.505, 0.485, 1.01, 0.495, 0.495, 0.985, 0.995, 0.99,
1.01, 0.49, 1.01, 0.995, 0.5, 0.995, 1.005, 0.495, 1., 0.995, 0.51, 0.98}, {14.965}, {0.995, 0.51, 0.995, 1.005, 0.495, 1.005, 0.495, 0.5, 0.985, 0.99, 0.985, 0.99, 0.995, 0.505, 0.49, 1.005, 0.495, 0.495, 0.99, 1.005, 0.495, 0.5, 0.98, 0.995, 0.985, 0.995, 0.995, 0.995, 0.5, 0.51, 0.495, 0.51, 0.485, 0.995, 0.985, 0.995, 0.98, 0.995, 0.98, 1., 0.98, 0.995, 0.98, 0.995, 0.995, 0.505, 0.485, 0.99, 0.985, 0.99, 0.995, 0.51, 0.5, 0.505, 0.49, 1., 0.495, 0.495, 0.98, 0.995, 0.995, 1.005, 0.49, 1.005, 0.995, 0.505, 0.995, 1.005, 0.495, 1., 0.99, 0.305}}}
```

In[31]:= Dimensions[rllruns]

Out[31]= {2, 53}

In[32]:= Map[Length, rllruns, {2}]

Out[32]= {{135, 1, 77, 1, 77, 1, 35, 1, 35, 1, 191, 1, 77, 1, 77, 1, 35, 1, 35, 1, 191, 1, 77, 1, 77, 1, 35, 1, 35, 1, 191, 1, 77, 1, 77, 1, 35, 1, 35, 1, 191, 1, 77, 1, 76}, {135, 1, 77, 1, 77, 1, 35, 1, 35, 1, 191, 1, 77, 1, 77, 1, 35, 1, 35, 1, 191, 1, 77, 1, 77, 1, 35, 1, 35, 1, 191, 1, 77, 1, 77, 1, 35, 1, 35, 1, 191, 1, 77, 1, 76}}

Length of packet separators in 100 μ s bit frames

In[33]:= TableForm[Transpose[Select[#, Length[#] <= 1 &] & /@ rllruns]]

Out[33]//TableForm=

14.95	14.965
14.955	14.96
54.78	54.785
16.745	16.75
41.645	41.66
14.95	14.96
14.955	14.965
54.78	54.785
16.735	16.745
41.635	41.65
14.955	14.965
14.955	14.965
54.775	54.79
16.735	16.74
41.645	41.655
14.955	14.97
14.95	14.96
54.785	54.79
16.735	16.75
41.645	41.655
14.95	14.965
14.955	14.965
54.78	54.79
16.74	16.75
41.645	41.655
14.955	14.965

Run length encoded packets

Display number of runs per packet

```
In[34]:= TableForm[  
  Transpose[Map[Length, packetruns = Select[#, Length[#] > 1 &] & /@ rlruns, {2}]]]  
  
Out[34]//TableForm=
```

135	135
77	77
77	77
35	35
35	35
191	191
77	77
77	77
35	35
35	35
191	191
77	77
77	77
35	35
35	35
191	191
77	77
77	77
35	35
35	35
191	191
77	77
77	77
35	35
35	35
191	191
76	76

Runs in $25\mu s$ time unit multiples

```
In[35]:= RoundRunTime = Round[4 #] &  
Out[35]= Round[4 #1] &  
  
In[36]:= MatrixForm[  
  Transpose[mmruns = RoundRunTime[Select[#, Length[#] == 35 &] & /@ packetruns]]]  
  
Out[36]//MatrixForm=
```

$$\begin{pmatrix} 7 & 7 \\ 1 & 1 \\ 1 & 1 \\ 7 & 7 \\ 7 & 7 \\ 1 & 1 \\ 1 & 1 \\ 7 & 7 \\ 7 & 7 \\ 1 & 1 \\ 1 & 1 \\ 7 & 7 \\ 7 & 7 \\ 1 & 1 \\ 1 & 1 \\ 7 & 7 \\ 1 & 1 \end{pmatrix}$$

$$\left(\begin{array}{c|c} 1 & 1 \\ 7 & 7 \\ 7 & 7 \\ 1 & 1 \\ 1 & 1 \\ 7 & 7 \\ 7 & 7 \\ 1 & 1 \\ 1 & 1 \end{array} \right)$$

```
In[37]:= MatrixForm[
Transpose[mediumruns = RoundRunTime[Select[#, Length[#] == 77 &] &/@ packetruns]]]
```

Out[37]//MatrixForm=

$$\left(\begin{array}{c|c} 4 & 4 \\ 2 & 2 \\ 4 & 4 \\ 4 & 4 \\ 2 & 2 \\ 4 & 4 \\ 2 & 2 \\ 2 & 2 \\ 4 & 4 \\ 4 & 4 \\ 4 & 4 \\ 4 & 4 \\ 4 & 4 \\ 2 & 2 \\ 2 & 2 \\ 4 & 4 \\ 2 & 2 \\ 2 & 2 \\ 4 & 4 \\ 4 & 4 \\ 4 & 4 \\ 4 & 4 \\ 4 & 4 \\ 4 & 4 \\ 4 & 4 \\ 2 & 2 \\ 2 & 2 \\ 2 & 2 \\ 2 & 2 \\ 2 & 2 \\ 2 & 2 \\ 4 & 4 \\ 4 & 4 \\ 4 & 4 \\ 4 & 4 \\ 4 & 4 \\ 4 & 4 \\ 4 & 4 \\ 4 & 4 \\ 4 & 4 \\ 4 & 4 \\ 4 & 4 \\ 4 & 4 \\ 2 & 2 \\ 2 & 2 \\ 2 & 2 \\ 2 & 2 \\ 2 & 2 \\ 2 & 2 \\ 4 & 4 \\ 4 & 4 \\ 4 & 4 \\ 4 & 4 \\ 4 & 4 \\ 4 & 4 \\ 4 & 4 \\ 4 & 4 \\ 4 & 4 \\ 4 & 4 \\ 4 & 4 \\ 4 & 4 \\ 4 & 4 \\ 2 & 2 \\ 2 & 2 \\ 2 & 2 \\ 2 & 2 \\ 2 & 2 \\ 2 & 2 \end{array} \right)$$

4	4
4	4
4	4
2	2
2	2
2	2
2	2
4	4
2	2
2	2
4	4
4	4
4	4
4	4
2	2
4	4
4	4
2	2
4	4
4	4
2	2
4	4
4	4
2	2
4	4
4	4
2	2
4	4
4	4
2	2
4	4
4	4
2	2
4	4
4	4
2	2
4	4
4	4
2	2
4	4
4	4
2	2
2	2
2	2
2	2
2	2
2	2
4	4
4	4
2	2

4	4	
2	2	
4	4	
2	2	
2	2	
4	4	
4	4	
4	4	
4	4	
4	4	
2	2	
2	2	
4	4	
2	2	
2	2	
4	4	
4	4	
2	2	
2	2	
4	4	
4	4	
4	4	
4	4	
4	4	
4	4	
4	4	
4	4	
2	2	
2	2	
2	2	
2	2	
2	2	
2	2	
4	4	
4	4	
4	4	
4	4	
4	4	
4	4	
4	4	
4	4	
2	2	
2	2	
4	4	
4	4	
4	4	
4	4	
2	2	
2	2	
2	2	
2	2	
4	4	
2	2	
2	2	
4	4	
4	4	
4	4	
2	2	
4	4	

4	4
2	2
4	4
4	4
2	2
4	4
4	4
2	2
4	4
4	4
2	2
4	4
4	4
2	2
2	2
4	4
2	2
2	2
4	4
4	4
2	2
2	2
4	4
4	4
4	4
4	4
4	4
4	4
4	4
4	4
2	2
2	2
2	2
2	2
2	2
2	2
4	4
4	4
4	4
4	4
4	4
4	4
4	4
4	4
4	4
4	4
4	4
2	2
2	2
4	4
4	4
4	4
4	4
2	2
-	-

2	2
2	2
2	2
2	2
2	2
2	2
4	4
4	4
4	4
4	4
4	4
4	4
4	4
4	4
4	4
4	4
4	4
4	4
4	4
4	4
4	4
2	2
2	2
4	4
4	4
4	4
4	4
2	2
2	2
2	2
2	2
2	2
2	2
4	4
2	2
2	2
4	4
4	4
4	4
2	2
4	4
4	4
2	2
4	4
4	4
2	2
2	2
4	4
2	2
2	2
4	4
4	4
(4)	(4)
2	2
4	4
4	4
2	2
4	4
2	2
2	2
4	4
4	4
4	4
4	4
2	2
2	2
2	2
4	4
2	2
2	2
4	4
4	4

4	4
4	4
4	4
4	4
4	4
2	2
2	2
4	4
2	2
2	2
4	4
4	4
2	2
2	2
4	4
4	4
4	4
4	4
4	4
4	4
4	4
4	4
4	4
4	4
4	4
2	2
2	2
2	2
2	2
2	2
4	4
4	4
4	4
4	4
4	4
4	4
4	4
4	4
4	4
2	2
2	2
4	4
4	4
4	4
2	2
2	2
2	2
2	2
4	4
2	2
2	2
4	4
4	4
4	4
2	2
4	4
4	4
2	2
4	4
4	4
2	2

$$\left(\begin{array}{c} 4 \\ 4 \\ 2 \\ 4 \end{array} \right) \quad \left(\begin{array}{c} 4 \\ 4 \\ 2 \\ 4 \end{array} \right)$$

```
In[38]:= MatrixForm[  
    Transpose[longruns = RoundRunTime[Select[#, Length[#] == 191 &] & /@ packetruns]]]
```

Out[38]//MatrixForm=

4	4
2	2
4	4
4	4
2	2
4	4
2	2
2	2
4	4
4	4
4	4
4	4
4	4
4	4
2	2
2	2
2	2
2	2
2	2
2	2
2	2
4	4
2	2
2	2
4	4
2	2
2	2
4	4
4	4
2	2
2	2
2	2
2	2
2	2
2	2
4	4
2	2
2	2
4	4
4	4
4	4
2	2
2	2
2	2
4	4
4	4
4	4
2	2
2	2
2	2
2	2
2	2
4	4
2	2

2	2
4	4
4	4
4	4
4	4
4	4
2	2
2	2
4	4
2	2
2	2
4	4
4	4
2	2
2	2
4	4
4	4
4	4
4	4
4	4
4	4
4	4
4	4
2	2
2	2
2	2
2	2
2	2
2	2
4	4
4	4
4	4
4	4
4	4
4	4
4	4
4	4
4	4
2	2
2	2
2	2
2	2
4	4
4	4
4	4
4	4
2	2
2	2
4	4
4	4
4	4
2	2
4	4
4	4
2	2
4	4
4	4
2	2
4	4
4	4

2	2
4	4
4	4
2	2
4	4
4	4
2	2
4	4
4	4
2	2
2	2
4	4
4	4
4	4
4	4
4	4
4	4
4	4
2	2
2	2
2	2
2	2
2	2
2	2
2	2
2	2
4	4
2	2
2	2
4	4
4	4
2	2
2	2
2	2
2	2
2	2
2	2
4	4
2	2
2	2
4	4
4	4
2	2
2	2
4	4
4	4
4	4
2	2
2	2
2	2
2	2
4	4
2	2
2	2
4	4
2	2
2	2

	4	4
	4	4
	2	2
	2	2
	2	2
	2	2
	4	4
	4	4
	4	4
	2	2
	2	2
	4	4
	4	4
	4	4
	4	4
	4	4
	4	4
	4	4
	4	4
	4	4
	4	4
	4	4
	4	4
	2	2
	2	2
	2	2
	2	2
	4	4
	2	2
	2	2
	4	4
	4	4
	2	2
	2	2
	2	2
	2	2
	4	4
	2	2
	2	2
	4	4
	4	4
	2	2
	4	4
	4	4
	2	2
	4	4
	4	4
	2	2
	4	4
	2	2
	2	2
	4	4
	4	4
	4	4
	4	4

4	4
4	4
2	2
2	2
4	4
2	2
2	2
4	4
4	4
2	2
2	2
4	4
4	4
4	4
4	4
4	4
4	4
4	4
4	4
4	4
4	4
4	4
4	4
2	2
2	2
2	2
2	2
2	2
2	2
4	4
4	4
4	4
4	4
4	4
4	4
4	4
4	4
2	2
2	2
4	4
4	4
4	4
4	4
2	2
2	2
2	2
2	2
4	4
2	2
2	2
4	4
4	4
4	4
2	2
4	4
4	4
2	2
4	4
4	4
2	2
4	4
4	4
2	2
4	4
4	4
2	2

\ 4)	\ 4)
4	4
2	2
4	4
4	4
2	2
4	4
2	2
2	2
4	4
4	4
4	4
4	4
4	4
4	4
4	4
2	2
2	2
2	2
2	2
2	2
2	2
2	2
2	2
2	2
4	4
2	2
2	2
4	4
4	4
2	2
2	2
2	2
2	2
2	2
4	4
2	2
2	2
4	4
4	4
2	2
2	2
4	4
4	4
2	2
2	2
2	2
2	2
4	4
2	2
2	2
4	4
4	4
2	2
2	2

2	2
2	2
4	4
4	4
4	4
2	2
2	2
4	4
4	4
4	4
4	4
4	4
4	4
4	4
4	4
4	4
4	4
4	4
4	4
4	4
2	2
2	2
2	2
2	2
4	4
2	2
2	2
4	4
4	4
4	4
2	2
2	2
2	2
2	2
4	4
2	2
2	2
4	4
2	2
4	4
4	4
2	2
4	4
4	4
2	2
4	4
4	4
2	2
2	2
4	4
4	4
4	4
4	4
2	2
2	2

4	4
2	2
2	2
4	4
4	4
4	4
4	4
4	4
4	4
4	4
4	4
4	4
4	4
4	4
4	4
4	4
4	4
4	4
4	4
2	2
2	2
2	2
2	2
4	4
2	2
2	2
4	4
4	4
2	2
2	2
2	2
2	2
2	2
4	4
2	2
2	2
4	4
4	4
2	2
4	4
4	4
2	2
2	2
4	4
4	4
4	4
4	4
2	2
2	2
4	4
2	2
2	2
4	4
4	4
4	4
4	4
2	2
2	2
4	4
2	2
2	2
4	4

4
4
4
4
4
4
4
4
4
4
2
2
2
2
2
2
2
4
4
4
4
4
4
4
4
4
4
4
4
4
4
4
4
4
2
2
2
4
4
4
4
2
2
2
2
2
2
2
4
2
2
2
4
4
4
4
2
2
2
4
4
4
4
2
4
4
2
4
4
2
4
4
2
4
4
2
4
4
2
4

```
In[39]:= truncruns = RoundRunTime[  
    Select[#, Function[l, And @@ (l != # & /@ {1, 35, 77, 191})] [Length[#]] &] & /@  
    packetruns]
```

```
In[40]:= Function[l, And @@ (l != # & /@ {1, 35, 77, 191})] [35]
```

Out[40]= False

Parse runs

Shotgun mfx parser

```
In[41]:= mfpARSER = Function[{in, out},
  If[Length[in] ≥ 6 ∧ {4, 2, 4, 4, 2, 4} == Take[in, 6],
    {Drop[in, 6], Append[out, SYNC]},
    If[Length[in] ≥ 2 ∧ {2, 2} == Take[in, 2], {Drop[in, 2], Append[out, 1]}, 
      If[{4} == Take[in, 1],
        {Drop[in, 1], Append[out, 0]}, {Drop[in, 1], Append[out, △]}]]]
  ]
]

Out[41]= Function[{in, out},
  If[Length[in] ≥ 6 && {4, 2, 4, 4, 2, 4} == Take[in, 6], {Drop[in, 6], Append[out, SYNC]}, 
    If[Length[in] ≥ 2 && {2, 2} == Take[in, 2], {Drop[in, 2], Append[out, 1]}, If[{4} ==
      Take[in, 1], {Drop[in, 1], Append[out, 0]}, {Drop[in, 1], Append[out, △]}]]]]]
```

Parse by iteration over run length encoded input

Shotgun MM parser

```
In[43]:= MMparser = Function[{in, out},
  If[Length[in] ≥ 2,
    Which[{1, 7} == Take[in, 2], {Drop[in, 2], Append[out, 0]}, {7, 1} == Take[in, 2], {Drop[in, 2], Append[out, 1]}, True, {Drop[in, 1], Append[out, △]}],
    If[Length[in] == 1, Which[{1} == Take[in, 1], {Drop[in, 1], Append[out, 0]}, {7} == Take[in, 1], {Drop[in, 1], Append[out, 1]}, True, {Drop[in, 1], Append[out, △]}], {Drop[in, 1], Append[out, △]}]]]
  ]
Out[43]= Function[{in, out}, If[Length[in] ≥ 2,
  Which[{1, 7} == Take[in, 2], {Drop[in, 2], Append[out, 0]}, {7, 1} == Take[in, 2], {Drop[in, 2], Append[out, 1]}, True, {Drop[in, 1], Append[out, △]}],
  If[Length[in] == 1, Which[{1} == Take[in, 1], {Drop[in, 1], Append[out, 0]}, {7} == Take[in, 1], {Drop[in, 1], Append[out, 1]}, True, {Drop[in, 1], Append[out, △]}], {Drop[in, 1], Append[out, △]}]]]
```

Parse by iteration over run length encoded input

```
In[44]:= NestWhile[MMparser @@ # &, {mmruns[[1, 1]], {}}, Length[First[#]] > 0 &]
Out[44]= {{}, {1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0}}
```

Apply parsers

Parse over medium length runs

```
In[45]:= mediumbitsequences =
  Map[Last[NestWhile[mfxfparser@@# &, {#, {}}, Length[First[#]] > 0 &]] &,
  mediumruns, {2}]
```

```
In[46]:= Map[Last[NestWhile[MMparser @@ # &, {#, {}}, Length[First[#]] > 0 &]] &,
mediumruns, {2}]
```


Parse over long runs

Parse over MM runs

```
In[49]:= Map[Last[NestWhile[mfparser @@ # &, {#, {}}, Length[First[#]] > 0 &]] &, mmpars, {2}]
```


Parse over truncated runs

Parse delimited mfx/DCC packets

Eliminate duplicate sequences

It turns out in this case that there is only one packet of each length, which has been repeated

```
In[53]:= longbitseq = Flatten[Union[Union /@ longbitsequences], 2]
Out[53]= {SYNC, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 0, 1, 0, 1, 0, 0, 0, 1, 1, 1, 0,
1, 0, 0, 0, 1, 0, 0, 0, 1, 1, 0, 1, 0, 1, 0, 0, 1, 1, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 1, 1, 0, 1, SYNC, SYNC, SYNC,
1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0}
```

Split by SYNC sequences

```
In[57]:= mediumbits = Select[SplitBy[mediumbitseq, NumberQ], Union[#] != {SYNC} &]

Out[57]= {{1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1,
           1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0}}
```

CRC function for mfx

The second series in the long packet is the same as the medium series.

Address parser

Parse the different mfx address formats

configuration parser

Parse the subcommands of the mfx configuration command

```
In[66]:= configanalyzer =
Function[{cmd, in, out},
Which[{1, 0, 1} == cmd, If[Length[in] < 48, {ERROR, in, Append[out, ERROR]}, {CMD, Drop[in, 48], Append[out, {CENTRALUID, FromDigits[Take[in, {1, 32}], 2}, COUNT, FromDigits[Take[in, {33, 48}], 2]}]}],
{1, 0, 0} == cmd, If[Length[in] < 32, {ERROR, in, Append[out, ERROR]}, {CMD, Drop[in, 32], Append[out, {PING, FromDigits[Take[in, {1, 32}], 2]}]}]],
{0, 1, 0} == cmd, If[Length[in] < 38, {ERROR, in, Append[out, ERROR]}, {CMD, Drop[in, 38], Append[out, {MATCHBITS,
FromDigits[Take[in, {1, 6}], 2], UID, FromDigits[Take[in, {7, 38}], 2]}]}],
True, {ERROR, in, {UNASSIGNEDCONFIG, cmd}}]
]
]

Out[66]= Function[{cmd, in, out},
Which[{1, 0, 1} == cmd, If[Length[in] < 48, {ERROR, in, Append[out, ERROR]}, {CMD, Drop[in, 48], Append[out, {CENTRALUID, FromDigits[Take[in, {1, 32}], 2}, COUNT, FromDigits[Take[in, {33, 48}], 2]}]}],
{1, 0, 0} == cmd, If[Length[in] < 32, {ERROR, in, Append[out, ERROR]}, {CMD, Drop[in, 32], Append[out, {PING, FromDigits[Take[in, {1, 32}], 2]}]}]],
{0, 1, 0} == cmd, If[Length[in] < 38, {ERROR, in, Append[out, ERROR]}, {CMD, Drop[in, 38], Append[out, {MATCHBITS,
FromDigits[Take[in, {1, 6}], 2], UID, FromDigits[Take[in, {7, 38}], 2]}]}],
True, {ERROR, in, {UNASSIGNEDCONFIG, cmd}}]]
```

command parser

Parse the different mfx commands

```
In[67]:= commandanalyzer =
Function[{in, out}, Which[Length[in] < 3, {ERROR, in, out}, {0, 0, 0} == Take[in, 3],
If[Length[in] < 7, {ERROR, in, Append[out, ERROR]}, {CMD, Drop[in, 7], Append[out,
{DRV3, If[in[4] == 0, FWD, REV, ERROR], FromDigits[16 Take[in, {5, 7}], 2]}]}], {0, 0, 1} == Take[in, 3], If[Length[in] < 11, {ERROR, in, Append[out, ERROR]}, {CMD, Drop[in, 11], Append[out,
{DRV7, If[in[4] == 0, FWD, REV, ERROR], FromDigits[Take[in, {5, 11}], 2]}]}], {0, 1, 0} == Take[in, 3], If[Length[in] < 7, {ERROR, in, Append[out, ERROR]}, {CMD, Drop[in, 7], Append[out, {FUNC4, Take[in, {4, 7}]}]}], {1, 0, 0} == Take[in, 3], If[Length[in] < 12, {ERROR, in, Append[out, ERROR]}, {CMD, Drop[in, 12], Append[out, {FUNC, FromDigits[Take[in, {4, 10}], 2]], in[12]}]}], {1, 1, 1} == Take[in, 3], If[Length[in] < 6, {ERROR, in, Append[out, ERROR]}, configanalyzer[Take[in, {4, 6}], Drop[in, 6], out]], {0, 1, 1, 0} == Take[in, 4], If[Length[in] < 12, {ERROR, in, Append[out, ERROR]}, {CMD, Drop[in, 12], Append[out, {FUNC8, Take[in, {5, 12}]}]}], {0, 1, 1, 1} == Take[in, 4], If[Length[in] < 20, {ERROR, in, Append[out, ERROR]}, {CMD, Drop[in, 20], Append[out, {FUNC16, Take[in, {5, 20}]}]}], True, {ERROR, in, Append[out, UNASSIGNEDCMD]}]]
Out[67]= Function[{in, out}, Which[Length[in] < 3, {ERROR, in, out}, {0, 0, 0} == Take[in, 3],
If[Length[in] < 7, {ERROR, in, Append[out, ERROR]}, {CMD, Drop[in, 7], Append[out,
{DRV3, If[in[4] == 0, FWD, REV, ERROR], FromDigits[16 Take[in, {5, 7}], 2]}]}], {0, 0, 1} == Take[in, 3], If[Length[in] < 11, {ERROR, in, Append[out, ERROR]}, {CMD, Drop[in, 11], Append[out,
{DRV7, If[in[4] == 0, FWD, REV, ERROR], FromDigits[Take[in, {5, 11}], 2]}]}], {0, 1, 0} == Take[in, 3], If[Length[in] < 7, {ERROR, in, Append[out, ERROR]}, {CMD, Drop[in, 7], Append[out, {FUNC4, Take[in, {4, 7}]}]}], {1, 0, 0} == Take[in, 3], If[Length[in] < 12, {ERROR, in, Append[out, ERROR]}, {CMD, Drop[in, 12], Append[out, {FUNC, FromDigits[Take[in, {4, 10}], 2]], in[12]}]}], {1, 1, 1} == Take[in, 3], If[Length[in] < 6, {ERROR, in, Append[out, ERROR]}, configanalyzer[Take[in, {4, 6}], Drop[in, 6], out]], {0, 1, 1, 0} == Take[in, 4], If[Length[in] < 12, {ERROR, in, Append[out, ERROR]}, {CMD, Drop[in, 12], Append[out, {FUNC8, Take[in, {5, 12}]}]}], {0, 1, 1, 1} == Take[in, 4], If[Length[in] < 20, {ERROR, in, Append[out, ERROR]}, {CMD, Drop[in, 20], Append[out, {FUNC16, Take[in, {5, 20}]}]}], True, {ERROR, in, Append[out, UNASSIGNEDCMD]}]]
```

mfx bit series parser

Top level parser for SYNC and CRC stripped mfx packets

```
In[68]:= mfxanalyzer = Function[{state, in, out},
  Switch[state,
    ERROR, {ERROR, {}, out},
    ADDR, addressanalyzer[in, out],
    CMD, commandanalyzer[in, out],
    _, {ERROR, in, out}]
]

Out[68]= Function[{state, in, out}, Switch[state,
  ERROR, {ERROR, {}, out},
  ADDR, addressanalyzer[in, out],
  CMD, commandanalyzer[in, out],
  _, {ERROR, in, out}]]
```

Parsing results of unique mfx packets

The drive command is transmitted 3 times more often than the central station UID and count broadcast

```
In[69]:= NestWhile[mfxanalyzer @@ # &, {ADDR, #, {}}, Length[#[[2]]] > 0 &] & /@ longseries
Out[69]= {{CMD, {}}, {{A7, 0}, {CENTRALUID, 1195666018, COUNT, 3}}}, {CMD, {}},
{{A7, 5}, {DRV7, FWD, 0}, {FUNC16, {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0}}}}}}
```

```
In[70]:= Print[MatrixForm[Last[NestWhile[mfxanalyzer @@ # &,
{ADDR, #, {}}, Length[#[[2]]] > 0 &]]]] & /@ longseries;
{{A7, 0}
{CENTRALUID, 1195666018, COUNT, 3}}
{{A7, 5}
{DRV7, FWD, 0}
{FUNC16, {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0}}}}
```