Analyze Märklin track signal

Get complementary signals from digitizing car

Load file

```
Directory[]
```

/home/cmaier/scad/Maerklin/analysis

```
Dimensions[raw = Import["TTL.10ms.5V.csv"]]
```

{524 290, 4}

```
Take[raw, 12]
```

```
{{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., },
 {-0.13111, 5.4, 0., }, {-0.13111, 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

```
Dimensions[
 timeseries = Transpose[{{#[[1]], #[[2]]}, {#[[1]], #[[3]]}} & /@ Drop[raw, 2], {2, 1, 3}]]
```

{2, 524 288, 2}

Calculate time step

```
Dimensions[ListConvolve[{1, -1}, First /@ #] & /@ timeseries]
```

{2, 524 287}

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

```
{Min[#], Max[#]} &[ListConvolve[{1, -1}, First /@ #]] & /@ timeseries
```

$\{\{0., 1. \times 10^{-6}\}, \{0., 1. \times 10^{-6}\}\}$

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

```
GraphicsGrid[
 {Histogram[ListConvolve[{1, -1}, First /@ #, PlotRange → All]]} & /@ timeseries]
```

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

$5. \times 10^{-7}$

Display

```
ListPlot[Take[#, {55 000, 155 000}] & /@ timeseries, Frame → True, Joined → True]
```

Digitize time series

```
Dimensions[digitized = Map[{#[[1]], HeavisideTheta[#[[2]] - 3.0001]} &, timeseries, {2}]]
```

```
{2, 524288, 2}
```

Display

```
ListPlot[Take[#, {55000, 155000}] & /@ digitized, Frame → True, Joined → True]
```

Display complementary phases

```
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

```
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

```
ListPlot[
 Take[MapThread[{(#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

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

```
{{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},
```

```
{51, 2}, {1, 2}, {361, 2}, {1, 2}, {51, 2}, {1, 2}, {362, 2}, {51, 2}, {2, 2},
{361, 2}, {1, 2}, {361, 2}, {2, 2}, {50, 2}, {1, 2}, {52, 2}, {1, 2}, {362, 2},
{362, 2}, {1, 2}, {51, 2}, {1, 2}, {50, 2}, {2, 2}, {361, 2}, {1, 2}, {362, 2},
{1, 2}, {52, 2}, {52, 2}, {2, 2}, {362, 2}, {1, 2}, {361, 2}, {1, 2}, {51, 2}, {1, 2},
{51, 2}, {1, 2}, {3347, 2}, {362, 2}, {1, 2}, {51, 2}, {1, 2}, {50, 2}, {2, 2},
{362, 2}, {363, 2}, {1, 2}, {51, 2}, {51, 2}, {2, 2}, {362, 2}, {1, 2}, {361, 2},
{1, 2}, {51, 2}, {1, 2}, {51, 2}, {1, 2}, {361, 2}, {1, 2}, {362, 2}, {1, 2}, {51, 2},
{51, 2}, {1, 2}, {362, 2}, {1, 2}, {52, 2}, {1, 2}, {361, 2}, {1, 2}, {51, 2}, {1, 2},
{362, 2}, {362, 2}, {1, 2}, {51, 2}, {1, 2}, {51, 2}, {1, 2}, {362, 2}, {1, 2},
{361, 2}, {2, 2}, {51, 2}, {52, 2}, {1, 2}, {362, 2}, {362, 2}, {1, 2}, {52, 2},
{1, 2}, {51, 2}, {1, 2}, {361, 2}, {1, 2}, {362, 2}, {1, 2}, {50, 2}, {1, 2}, {51, 2},
{1, 2}, {8329, 2}, {1, 2}, {199, 2}, {1, 2}, {100, 2}, {1, 2}, {199, 2}, {1, 2},
{199, 2}, {1, 2}, {99, 2}, {2, 2}, {199, 2}, {100, 2}, {1, 2}, {97, 2}, {197, 2},
{1, 2}, {197, 2}, {1, 2}, {197, 2}, {1, 2}, {197, 2}, {198, 2}, {1, 2}, {197, 2},
{1, 2}, {196, 2}, {1, 2}, {198, 2}, {1, 2}, {99, 2}, {1, 2}, {100, 2}, {100, 2},
{1, 2}, {99, 2}, {1, 2}, {99, 2}, {1, 2}, {99, 2}, {1, 2}, {98, 2}, {2, 2}, {97, 2},
{1, 2}, {198, 2}, {1, 2}, {100, 2}, {1, 2}, {96, 2}, {1, 2}, {199, 2}, {1, 2},
{99, 2}, {1, 2}, {98, 2}, {197, 2}, {1, 2}, {196, 2}, {1, 2}, {199, 2}, {1, 2},
{100, 2}, {1, 2}, {98, 2}, {2, 2}, {99, 2}, {1, 2}, {100, 2}, {1, 2}, {99, 2},
{1, 2}, {97, 2}, {1, 2}, {198, 2}, {1, 2}, {99, 2}, {1, 2}, {98, 2}, {197, 2},
{2, 2}, {196, 2}, {199, 2}, {1, 2}, {100, 2}, {98, 2}, {1, 2}, {197, 2}, {1, 2},
{196, 2}, {2, 2}, {199, 2}, {1, 2}, {99, 2}, {1, 2}, {99, 2}, {1, 2}, {99, 2}, {2, 2},
{97, 2}, {1, 2}, {199, 2}, {1, 2}, {99, 2}, {1, 2}, {97, 2}, {1, 2}, {200, 2},
{100, 2}, {1, 2}, {97, 2}, {198, 2}, {1, 2}, {198, 2}, {1, 2}, {98, 2}, {2, 2},
{99, 2}, {1, 2}, {100, 2}, {1, 2}, {97, 2}, {1, 2}, {196, 2}, {1, 2}, {197, 2},
{1, 2}, {199, 2}, {1, 2}, {99, 2}, {1, 2}, {98, 2}, {1, 2}, {197, 2}, {1, 2},
{196, 2}, {1, 2}, {198, 2}, {198, 2}, {1, 2}, {197, 2}, {1, 2}, {197, 2}, {1, 2},
{197, 2}, {1, 2}, {197, 2}, {2, 2}, {197, 2}, {198, 2}, {1, 2}, {197, 2}, {1, 2},
{196, 2}, {1, 2}, {196, 2}, {1, 2}, {196, 2}, {2, 2}, {199, 2}, {99, 2}, {1, 2},
{100, 2}, {1, 2}, {99, 2}, {2, 2}, {97, 2}, {1, 2}, {198, 2}, {1, 2}, {100, 2},
{1, 2}, {96, 2}, {2, 2}, {196, 2}, {1, 2}, {199, 2}, {1, 2}, {99, 2}, {100, 2},
{1, 2}, {100, 2}, {1, 2}, {97, 2}, {1, 2}, {200, 2}, {100, 2}, {1, 2}, {100, 2},
{1, 2}, {199, 2}, {1, 2}, {99, 2}, {199, 2}, {2, 2}, {199, 2}, {1, 2}, {99, 2},
{1, 2}, {200, 2}, {200, 2}, {1, 2}, {99, 2}, {1, 2}, {199, 2}, {2, 2}, {199, 2},
{1, 2}, {99, 2}, {2, 2}, {201, 2}, {200, 2}, {1, 2}, {99, 2}, {199, 2}, {2, 2},
{198, 2}, {1, 2}, {99, 2}, {2, 2}, {198, 2}, {100, 2}, {1, 2}, {98, 2}, {198, 2},
{1, 2}, {197, 2}, {1, 2}, {197, 2}, {1, 2}, {198, 2}, {199, 2}, {2, 2}, {99, 2},
{1, 2}, {98, 2}, {1, 2}, {199, 2}, {1, 2}, {99, 2}, {1, 2}, {97, 2}, {1, 2},
{197, 2}, {1, 2}, {198, 2}, {1, 2}, {100, 2}, {1, 2}, {98, 2}, {196, 2}, {2, 2},
{196, 2}, {1, 2}, {196, 2}, {2, 2}, {196, 2}, {1, 2}, {196, 2}, {2, 2}, {196, 2},
{1, 2}, {197, 2}, {1, 2}, {197, 2}, {1, 2}, {199, 2}, {2, 2}, {99, 2}, {1, 2},
{100, 2}, {1, 2}, {99, 2}, {1, 2}, {100, 2}, {1, 2}, {100, 2}, {98, 2}, {1, 2},
{198, 2}, {197, 2}, {2, 2}, {197, 2}, {197, 2}, {1, 2}, {197, 2}, {1, 2}, {197, 2},
{1, 2}, {196, 2}, {1, 2}, {196, 2}, {2, 2}, {196, 2}, {1, 2}, {197, 2}, {1, 2},
{196, 2}, {1, 2}, {199, 2}, {1, 2}, {99, 2}, {1, 2}, {97, 2}, {1, 2}, {197, 2},
{1, 2}, {197, 2}, {1, 2}, {196, 2}, {1, 2}, {198, 2}, {2, 2}, {99, 2}, {1, 2},
{100, 2}, {1, 2}, {99, 2}, {1, 2}, {97, 2}, {2, 2}, {198, 2}, {1, 2}, {99, 2},
{1, 2}, {97, 2}, {1, 2}, {197, 2}, {2, 2}, {196, 2}, {1, 2}, {199, 2}, {1, 2},
{198, 2}, {1, 2}, {98, 2}, {2, 2}, {198, 2}, {1, 2}, {199, 2}, {1, 2}, {100, 2},
```

```
{199, 2}, {1, 2}, {199, 2}, {1, 2}, {100, 2}, {1, 2}, {199, 2}, {199, 2}, {1, 2},
{98, 2}, {1, 2}, {195, 2}, {2, 2}, {2991, 2}, {1, 2}, {199, 2}, {1, 2}, {100, 2},
{200, 2}, {1, 2}, {199, 2}, {1, 2}, {99, 2}, {2, 2}, {199, 2}, {100, 2}, {1, 2},
{98, 2}, {198, 2}, {1, 2}, {197, 2}, {1, 2}, {196, 2}, {1, 2}, {197, 2}, {1, 2},
{199, 2}, {2, 2}, {99, 2}, {1, 2}, {98, 2}, {1, 2}, {199, 2}, {1, 2}, {99, 2},
{2, 2}, {97, 2}, {1, 2}, {197, 2}, {2, 2}, {199, 2}, {100, 2}, {1, 2}, {97, 2},
{197, 2}, {2, 2}, {196, 2}, {1, 2}, {196, 2}, {1, 2}, {197, 2}, {1, 2}, {196, 2},
{1, 2}, {197, 2}, {197, 2}, {1, 2}, {197, 2}, {1, 2}, {198, 2}, {1, 2}, {100, 2},
{1, 2}, {99, 2}, {2, 2}, {99, 2}, {1, 2}, {98, 2}, {2, 2}, {99, 2}, {1, 2}, {98, 2},
{1, 2}, {197, 2}, {197, 2}, {1, 2}, {196, 2}, {1, 2}, {197, 2}, {1, 2}, {197, 2},
{1, 2}, {197, 2}, {1, 2}, {197, 2}, {1, 2}, {196, 2}, {1, 2}, {196, 2}, {1, 2},
{197, 2}, {1, 2}, {196, 2}, {1, 2}, {199, 2}, {1, 2}, {100, 2}, {98, 2}, {1, 2},
{196, 2}, {1, 2}, {196, 2}, {1, 2}, {197, 2}, {1, 2}, {198, 2}, {2, 2}, {98, 2},
{1, 2}, {100, 2}, {1, 2}, {98, 2}, {1, 2}, {97, 2}, {1, 2}, {199, 2}, {100, 2},
{1, 2}, {98, 2}, {197, 2}, {2, 2}, {196, 2}, {1, 2}, {199, 2}, {1, 2}, {199, 2},
{1, 2}, {100, 2}, {1, 2}, {199, 2}, {1, 2}, {199, 2}, {2, 2}, {99, 2}, {1, 2},
{199, 2}, {1, 2}, {198, 2}, {1, 2}, {100, 2}, {1, 2}, {199, 2}, {1, 2}, {198, 2},
{1, 2}, {100, 2}, {1, 2}, {195, 2}, {1, 2}, {2990, 2}, {1, 2}, {199, 2}, {2, 2},
{98, 2}, {1, 2}, {198, 2}, {2, 2}, {199, 2}, {100, 2}, {1, 2}, {199, 2}, {1, 2},
{100, 2}, {1, 2}, {98, 2}, {197, 2}, {2, 2}, {196, 2}, {1, 2}, {197, 2}, {1, 2},
{197, 2}, {1, 2}, {199, 2}, {2, 2}, {99, 2}, {1, 2}, {98, 2}, {1, 2}, {199, 2},
{1, 2}, {99, 2}, {2, 2}, {97, 2}, {1, 2}, {197, 2}, {1, 2}, {200, 2}, {100, 2},
{1, 2}, {98, 2}, {197, 2}, {1, 2}, {197, 2}, {1, 2}, {197, 2}, {1, 2}, {196, 2},
{1, 2}, {196, 2}, {2, 2}, {196, 2}, {1, 2}, {197, 2}, {1, 2}, {196, 2}, {1, 2},
{199, 2}, {2, 2}, {99, 2}, {1, 2}, {100, 2}, {1, 2}, {99, 2}, {1, 2}, {99, 2},
{2, 2}, {98, 2}, {1, 2}, {97, 2}, {1, 2}, {196, 2}, {1, 2}, {197, 2}, {1, 2},
{198, 2}, {197, 2}, {1, 2}, {197, 2}, {1, 2}, {197, 2}, {1, 2}, {196, 2}, {1, 2},
{196, 2}, {1, 2}, {197, 2}, {197, 2}, {1, 2}, {196, 2}, {1, 2}, {199, 2}, {1, 2},
{99, 2}, {1, 2}, {97, 2}, {2, 2}, {197, 2}, {197, 2}, {1, 2}, {197, 2}, {1, 2},
{198, 2}, {2, 2}, {99, 2}, {1, 2}, {100, 2}, {1, 2}, {99, 2}, {1, 2}, {97, 2},
{2, 2}, {198, 2}, {1, 2}, {99, 2}, {1, 2}, {97, 2}, {1, 2}, {197, 2}, {2, 2},
{197, 2}, {199, 2}, {1, 2}, {198, 2}, {1, 2}, {98, 2}, {2, 2}, {198, 2}, {1, 2},
{199, 2}, {1, 2}, {100, 2}, {199, 2}, {1, 2}, {198, 2}, {1, 2}, {100, 2}, {1, 2},
{199, 2}, {1, 2}, {199, 2}, {1, 2}, {100, 2}, {1, 2}, {196, 2}, {1, 2}, {9960, 2}}
```

```
ListPlot[{10^3 First[First[#]], 10^6 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

```
Map[Dimensions, runs = SplitBy[Take[#, {255 000, 355 000}], Last] & /@ digitized, {2}]
```

```
{{{3497, 2}, {365, 2}, {51, 2}, {53, 2}, {362, 2}, {364, 2}, {51, 2}, {53, 2}, {361, 2},
  {364, 2}, {51, 2}, {53, 2}, {362, 2}, {363, 2}, {51, 2}, {53, 2}, {361, 2}, {53, 2},
  {362, 2}, {53, 2}, {361, 2}, {364, 2}, {50, 2}, {54, 2}, {362, 2}, {363, 2},
  {51, 2}, {53, 2}, {361, 2}, {364, 2}, {52, 2}, {54, 2}, {362, 2}, {363, 2}, {51, 2},
  {53, 2}, {3347, 2}, {363, 2}, {51, 2}, {53, 2}, {362, 2}, {364, 2}, {51, 2},
  {53, 2}, {362, 2}, {363, 2}, {51, 2}, {53, 2}, {361, 2}, {364, 2}, {51, 2}, {52, 2},
  {362, 2}, {54, 2}, {361, 2}, {53, 2}, {362, 2}, {363, 2}, {51, 2}, {53, 2}, {362, 2},
  {364, 2}, {51, 2}, {53, 2}, {362, 2}, {363, 2}, {52, 2}, {53, 2}, {361, 2},
  {364, 2}, {50, 2}, {53, 2}, {8329, 2}, {201, 2}, {100, 2}, {201, 2}, {199, 2},
  {102, 2}, {199, 2}, {101, 2}, {97, 2}, {198, 2}, {197, 2}, {199, 2}, {197, 2},
  {199, 2}, {197, 2}, {198, 2}, {198, 2}, {101, 2}, {100, 2}, {101, 2}, {99, 2},
  {101, 2}, {99, 2}, {101, 2}, {97, 2}, {200, 2}, {100, 2}, {98, 2}, {199, 2},
  {101, 2}, {98, 2}, {198, 2}, {196, 2}, {201, 2}, {100, 2}, {101, 2}, {99, 2},
  {102, 2}, {99, 2}, {99, 2}, {198, 2}, {101, 2}, {98, 2}, {199, 2}, {196, 2},
  {200, 2}, {100, 2}, {99, 2}, {197, 2}, {199, 2}, {199, 2}, {101, 2}, {99, 2},
  {102, 2}, {97, 2}, {201, 2}, {99, 2}, {99, 2}, {200, 2}, {101, 2}, {97, 2},
  {199, 2}, {198, 2}, {101, 2}, {99, 2}, {102, 2}, {97, 2}, {198, 2}, {197, 2},
  {201, 2}, {99, 2}, {100, 2}, {197, 2}, {198, 2}, {198, 2}, {199, 2}, {197, 2},
  {199, 2}, {197, 2}, {200, 2}, {197, 2}, {199, 2}, {197, 2}, {198, 2}, {196, 2},
  {199, 2}, {199, 2}, {100, 2}, {100, 2}, {102, 2}, {97, 2}, {200, 2}, {100, 2},
  {99, 2}, {196, 2}, {201, 2}, {99, 2}, {101, 2}, {100, 2}, {99, 2}, {200, 2},
  {101, 2}, {100, 2}, {201, 2}, {99, 2}, {201, 2}, {199, 2}, {101, 2}, {200, 2},
  {201, 2}, {99, 2}, {202, 2}, {199, 2}, {102, 2}, {201, 2}, {201, 2}, {99, 2},
  {201, 2}, {198, 2}, {102, 2}, {198, 2}, {101, 2}, {98, 2}, {199, 2}, {197, 2},
  {199, 2}, {198, 2}, {201, 2}, {99, 2}, {100, 2}, {199, 2}, {101, 2}, {97, 2},
  {199, 2}, {198, 2}, {102, 2}, {98, 2}, {198, 2}, {196, 2}, {199, 2}, {196, 2},
  {199, 2}, {196, 2}, {199, 2}, {197, 2}, {202, 2}, {99, 2}, {102, 2}, {99, 2},
  {102, 2}, {100, 2}, {99, 2}, {198, 2}, {199, 2}, {197, 2}, {198, 2}, {197, 2},
  {199, 2}, {196, 2}, {199, 2}, {196, 2}, {199, 2}, {196, 2}, {201, 2}, {99, 2},
  {99, 2}, {197, 2}, {199, 2}, {196, 2}, {201, 2}, {99, 2}, {102, 2}, {99, 2},
  {100, 2}, {198, 2}, {101, 2}, {97, 2}, {200, 2}, {196, 2}, {201, 2}, {198, 2},
```

```
  {101, 2}, {198, 2}, {201, 2}, {100, 2}, {200, 2}, {199, 2}, {102, 2}, {199, 2},
  {200, 2}, {98, 2}, {198, 2}, {2991, 2}, {201, 2}, {100, 2}, {201, 2}, {199, 2},
  {102, 2}, {199, 2}, {101, 2}, {98, 2}, {199, 2}, {197, 2}, {198, 2}, {197, 2},
  {202, 2}, {99, 2}, {100, 2}, {199, 2}, {102, 2}, {97, 2}, {200, 2}, {199, 2},
  {101, 2}, {97, 2}, {199, 2}, {196, 2}, {198, 2}, {197, 2}, {198, 2}, {197, 2},
  {198, 2}, {197, 2}, {200, 2}, {100, 2}, {102, 2}, {99, 2}, {101, 2}, {99, 2},
  {100, 2}, {197, 2}, {198, 2}, {196, 2}, {199, 2}, {197, 2}, {199, 2}, {197, 2},
  {198, 2}, {196, 2}, {199, 2}, {196, 2}, {201, 2}, {100, 2}, {99, 2}, {196, 2},
  {198, 2}, {197, 2}, {201, 2}, {98, 2}, {102, 2}, {98, 2}, {99, 2}, {199, 2},
  {101, 2}, {98, 2}, {199, 2}, {196, 2}, {201, 2}, {199, 2}, {102, 2}, {199, 2},
  {202, 2}, {99, 2}, {201, 2}, {198, 2}, {102, 2}, {199, 2}, {200, 2}, {100, 2},
  {197, 2}, {2990, 2}, {202, 2}, {98, 2}, {201, 2}, {199, 2}, {101, 2}, {199, 2},
  {102, 2}, {98, 2}, {199, 2}, {196, 2}, {199, 2}, {197, 2}, {202, 2}, {99, 2},
  {100, 2}, {199, 2}, {102, 2}, {97, 2}, {199, 2}, {200, 2}, {101, 2}, {98, 2},
  {198, 2}, {197, 2}, {199, 2}, {196, 2}, {199, 2}, {196, 2}, {199, 2}, {196, 2},
  {202, 2}, {99, 2}, {102, 2}, {99, 2}, {102, 2}, {98, 2}, {99, 2}, {196, 2},
  {199, 2}, {198, 2}, {198, 2}, {197, 2}, {199, 2}, {196, 2}, {198, 2}, {197, 2},
  {198, 2}, {196, 2}, {201, 2}, {99, 2}, {100, 2}, {197, 2}, {198, 2}, {197, 2},
  {201, 2}, {99, 2}, {102, 2}, {99, 2}, {100, 2}, {198, 2}, {101, 2}, {97, 2},
  {200, 2}, {197, 2}, {200, 2}, {198, 2}, {101, 2}, {198, 2}, {201, 2}, {100, 2},
  {200, 2}, {198, 2}, {102, 2}, {199, 2}, {201, 2}, {100, 2}, {198, 2}, {9960, 2}},
 {{3498, 2}, {363, 2}, {52, 2}, {51, 2}, {365, 2}, {361, 2}, {54, 2}, {51, 2},
  {363, 2}, {362, 2}, {52, 2}, {51, 2}, {365, 2}, {361, 2}, {53, 2}, {51, 2}, {363, 2},
  {51, 2}, {363, 2}, {51, 2}, {364, 2}, {361, 2}, {53, 2}, {52, 2}, {363, 2}, {362, 2},
  {53, 2}, {50, 2}, {364, 2}, {362, 2}, {53, 2}, {52, 2}, {365, 2}, {361, 2}, {53, 2},
  {51, 2}, {3348, 2}, {362, 2}, {53, 2}, {50, 2}, {364, 2}, {363, 2}, {52, 2},
  {51, 2}, {365, 2}, {361, 2}, {53, 2}, {51, 2}, {363, 2}, {362, 2}, {52, 2}, {51, 2},
  {364, 2}, {52, 2}, {363, 2}, {51, 2}, {363, 2}, {362, 2}, {53, 2}, {51, 2}, {364, 2},
  {361, 2}, {53, 2}, {52, 2}, {363, 2}, {362, 2}, {54, 2}, {51, 2}, {363, 2},
  {362, 2}, {52, 2}, {51, 2}, {8331, 2}, {199, 2}, {102, 2}, {199, 2}, {201, 2},
  {99, 2}, {201, 2}, {100, 2}, {98, 2}, {197, 2}, {199, 2}, {197, 2}, {198, 2},
  {198, 2}, {199, 2}, {196, 2}, {200, 2}, {99, 2}, {101, 2}, {100, 2}, {101, 2},
  {99, 2}, {101, 2}, {98, 2}, {100, 2}, {198, 2}, {102, 2}, {96, 2}, {201, 2},
  {99, 2}, {99, 2}, {197, 2}, {198, 2}, {199, 2}, {102, 2}, {98, 2}, {102, 2},
  {100, 2}, {101, 2}, {97, 2}, {200, 2}, {99, 2}, {99, 2}, {197, 2}, {198, 2},
  {199, 2}, {101, 2}, {98, 2}, {199, 2}, {196, 2}, {202, 2}, {99, 2}, {101, 2},
  {99, 2}, {100, 2}, {199, 2}, {101, 2}, {97, 2}, {201, 2}, {100, 2}, {98, 2},
  {198, 2}, {200, 2}, {98, 2}, {102, 2}, {100, 2}, {99, 2}, {196, 2}, {199, 2},
  {199, 2}, {101, 2}, {98, 2}, {199, 2}, {196, 2}, {199, 2}, {198, 2}, {199, 2},
  {197, 2}, {199, 2}, {197, 2}, {199, 2}, {198, 2}, {199, 2}, {196, 2}, {198, 2},
  {196, 2}, {201, 2}, {99, 2}, {102, 2}, {99, 2}, {100, 2}, {198, 2}, {102, 2},
  {96, 2}, {199, 2}, {199, 2}, {100, 2}, {100, 2}, {102, 2}, {97, 2}, {201, 2},
  {100, 2}, {102, 2}, {199, 2}, {100, 2}, {199, 2}, {202, 2}, {99, 2}, {201, 2},
  {200, 2}, {101, 2}, {199, 2}, {202, 2}, {99, 2}, {203, 2}, {200, 2}, {100, 2},
  {199, 2}, {201, 2}, {99, 2}, {200, 2}, {100, 2}, {99, 2}, {198, 2}, {199, 2},
  {197, 2}, {199, 2}, {199, 2}, {102, 2}, {98, 2}, {201, 2}, {99, 2}, {99, 2},
  {197, 2}, {200, 2}, {100, 2}, {99, 2}, {196, 2}, {199, 2}, {196, 2}, {199, 2},
  {196, 2}, {199, 2}, {197, 2}, {199, 2}, {199, 2}, {102, 2}, {100, 2}, {101, 2},
  {100, 2}, {101, 2}, {98, 2}, {199, 2}, {197, 2}, {199, 2}, {197, 2}, {199, 2},
```

```
{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}, {198, 2},
{197, 2}, {199, 2}, {197, 2}, {199, 2}, {197, 2}, {198, 2}, {196, 2}, {198, 2},
{197, 2}, {198, 2}, {199, 2}, {101, 2}, {97, 2}, {199, 2}, {197, 2}, {199, 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}, {200, 2}, {100, 2}, {201, 2}, {199, 2}, {102, 2}, {196, 2}, {9961, 2}}}
```

```
Length /@ runs
```

```
{421, 421}
```

```
GraphicsGrid[
 {ListPlot[{10^3 First[First[#]], 10^6 timestep Length[#]} & /@ #, Frame → True,
    FrameLabel → {"Time [ms]", "Run length [μs]"}]} & /@ runs]
```

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

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

```
GraphicsGrid[
 {ListPlot[Take[#, {268 000, 278 000}], Frame → True, Joined → True, FrameLabel →
     {"Time [s]", ""}, GridLines → {Range[-.15, .15, 1.*^-4], None}]} & /@ digitized]
```

Medium length data packets

mfx signal

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

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

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

Long data packet

```
GraphicsGrid[
 {ListPlot[Take[#, {282 000, 316 000}], Frame → True, Joined → True, FrameLabel →
     {"Time [s]", ""}, GridLines → {Range[-.15, .15, 1.*^-4], None}]} & /@ digitized]
```

Run length decoding

bit frame (100μs) per sampling step

$$\frac{\text{timestep}}{10^{-4}}$$

0.005

Get run lengths of full traces

```
Length /@ (fullruns = SplitBy[#, Last] & /@ digitized)
```

{2312, 2312}

Run lengths in bit frames

$$\text{rlbits} = \text{Map}\left[\frac{\text{timestep}}{10^{-4}}\ \text{Length}[\#]\ \&,\ \text{fullruns},\ \{2\}\right]$$

{{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, 0.51, 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, 0.5, 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, 0.505, 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, 0.99, 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,
 0.98, 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, 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, 0.5, 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, 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, 1.815,
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, 0.995, 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, 0.505, 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, 0.995, 0.51, 0.49, 1.005, 0.495, 0.5,
0.985, 1.005, 0.5, 0.495, 0.985, 0.995, 0.985, 0.99, 0.98, 0.995, 0.985, 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,
0.985, 0.995, 0.99, 0.51, 0.5, 0.505, 0.485, 1.005, 0.495, 0.495, 0.985,
0.995, 0.995, 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,

```
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, 0.985, 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, 0.99, 0.505,
0.485, 0.995, 0.99, 0.505, 0.5, 0.505, 0.49, 1.005, 0.495, 0.5, 0.995, 0.51,
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, 0.995, 0.98,
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, 0.985, 0.995, 0.985,
0.995, 0.995, 0.5, 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}}
```

Difference in run lengths of opposite polarities in sampling steps

```
ListPlot[MapThread[(#1 - #2)/(timestep/10^-4) &, rlbits], Frame → True, PlotRange → All]
```

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

```
rlruns = SplitBy[#, # > 2 &] & /@ rlbits
```

```
{{{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, 0.51, 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, 0.5, 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, 0.505, 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, 0.99,
 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, 0.98, 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, 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, 0.5, 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,
```

```
 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, 1.815, 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, 0.995, 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, 0.505, 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, 0.995, 0.51, 0.49, 1.005, 0.495, 0.5, 0.985, 1.005, 0.5, 0.495,
 0.985, 0.995, 0.985, 0.99, 0.98, 0.995, 0.985, 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, 0.985, 0.995, 0.99,
 0.51, 0.5, 0.505, 0.485, 1.005, 0.495, 0.495, 0.985, 0.995, 0.995, 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, 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, 0.985, 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, 0.99, 0.505, 0.485,
 0.995, 0.99, 0.505, 0.5, 0.505, 0.49, 1.005, 0.495, 0.5, 0.995, 0.51, 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, 0.995, 0.98, 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, 0.985, 0.995, 0.985, 0.995, 0.995,
    0.5, 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}}}
```

```
Dimensions[rlruns]
```

```
{2, 53}
```

```
Map[Length, rlruns, {2}]
```

```
{{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, 77, 1, 35, 1, 35, 1, 191, 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, 77, 1, 35, 1, 35, 1, 191, 1, 76}}
```

Length of packet separators in 100μs bit frames

```
TableForm[Transpose[Select[#, Length[#] ≤ 1 &] & /@ rlruns]]
```

```
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

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

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 25μs time unit multiples

```
RoundRunTime = Round[4 #] &
```

```
Round[4 #1] &
```

```
MatrixForm[
 Transpose[mmruns = RoundRunTime[Select[#, Length[#] == 35 &] & /@ packetruns]]]
```

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

$$\left|\begin{pmatrix}1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\end{pmatrix}\begin{pmatrix}1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\end{pmatrix}\right|$$

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

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

$$\left|\begin{array}{cc}
\begin{pmatrix}1\\1\\7\\1\\7\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\end{pmatrix} & \begin{pmatrix}1\\1\\7\\1\\7\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\end{pmatrix} \\
\begin{pmatrix}7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\1\\7\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\end{pmatrix} & \begin{pmatrix}7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\1\\7\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\end{pmatrix} \\
\begin{pmatrix}7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\end{pmatrix} & \begin{pmatrix}7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\end{pmatrix}
\end{array}\right|$$

$$\left|\begin{pmatrix}7\\1\\1\\7\\7\\1\\1\\7\\1\\7\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\end{pmatrix}\begin{pmatrix}7\\1\\1\\7\\7\\1\\1\\7\\1\\7\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\end{pmatrix}\right|$$

$$\left|\begin{pmatrix}7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\1\\7\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\end{pmatrix}\begin{pmatrix}7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\1\\7\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\end{pmatrix}\right|$$

$$\left|\begin{pmatrix}7\\1\\1\end{pmatrix}\begin{pmatrix}7\\1\\1\end{pmatrix}\right|$$

$$\left|\begin{array}{cc}
\begin{pmatrix}7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\1\\7\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\end{pmatrix} & \begin{pmatrix}7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\1\\7\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\end{pmatrix} \\
\begin{pmatrix}7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\1\\7\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\end{pmatrix} & \begin{pmatrix}7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\1\\7\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\end{pmatrix}
\end{array}\right|$$

$$\left|\begin{array}{cc}
\begin{pmatrix}1\\1\end{pmatrix} & \begin{pmatrix}1\\1\end{pmatrix}\\
\begin{pmatrix}7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\1\\7\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\1\\7\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7 & \begin{pmatrix}7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\1\\7\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\1\\7\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\end{pmatrix}\\
\begin{pmatrix}7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\1\\7\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\1\\7\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7 & \begin{pmatrix}7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\1\\7\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\\1\\7\\1\\7\\7\\1\\1\\7\\7\\1\\1\\7\end{pmatrix}
\end{array}\right|$$

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

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

$$\begin{pmatrix} \begin{pmatrix} 4 \\ 2 \\ 4 \\ 4 \\ 2 \\ 4 \\ 2 \\ 2 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 2 \\ 2 \\ 4 \\ 2 \\ 2 \\ 4 \\ 4 \\ 2 \\ 2 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 2 \\ 2 \\ 2 \\ 2 \\ 2 \\ 2 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 2 \\ 2 \\ 4 \\ 4 \\ 4 \end{pmatrix} & \begin{pmatrix} 4 \\ 2 \\ 4 \\ 4 \\ 2 \\ 4 \\ 2 \\ 2 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 2 \\ 2 \\ 4 \\ 2 \\ 2 \\ 4 \\ 4 \\ 2 \\ 2 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 2 \\ 2 \\ 2 \\ 2 \\ 2 \\ 2 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 2 \\ 2 \\ 4 \\ 4 \\ 4 \end{pmatrix} \end{pmatrix}$$

$$\left|\begin{pmatrix}4\\2\\2\\2\\2\\4\\2\\2\\4\\4\\4\\4\\2\\4\\4\\2\\4\\4\\2\\4\\4\\2\end{pmatrix}\begin{pmatrix}4\\2\\2\\2\\2\\4\\2\\2\\4\\4\\4\\4\\2\\4\\4\\2\\4\\4\\2\\4\\4\\2\end{pmatrix}\right|$$

$$\left|\begin{pmatrix}4\\2\\4\\4\\2\\4\\2\\2\\4\\4\\4\\4\\4\\2\\2\\4\\2\\2\\4\\4\\2\\2\\4\\4\\4\\4\\4\\4\\4\\4\\4\\2\\2\\2\\2\\2\\2\\4\\4\\4\\4\\4\end{pmatrix}\begin{pmatrix}4\\2\\4\\4\\2\\4\\2\\2\\4\\4\\4\\4\\4\\2\\2\\4\\2\\2\\4\\4\\2\\2\\4\\4\\4\\4\\4\\4\\4\\4\\4\\2\\2\\2\\2\\2\\2\\4\\4\\4\\4\\4\end{pmatrix}\right|$$

$$\left|\begin{pmatrix}4\\4\\4\\4\\4\\4\\4\\2\\2\\4\\4\\4\\4\\2\\2\\2\\2\\4\\2\\2\\4\\4\\4\\4\\2\\4\\4\\2\\4\\4\\2\\4\\4\\2\\4\end{pmatrix}\begin{pmatrix}4\\4\\4\\4\\4\\4\\4\\2\\2\\4\\4\\4\\4\\2\\2\\2\\2\\4\\2\\2\\4\\4\\4\\4\\2\\4\\4\\2\\4\\4\\2\\4\\4\\2\\4\end{pmatrix}\right|$$

$$\left|\begin{pmatrix}4\\2\\4\\4\\2\\4\\2\\2\\4\\4\\4\\4\\4\\2\\2\\4\\2\\2\\4\\4\\2\\2\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\end{pmatrix}\begin{pmatrix}4\\2\\4\\4\\2\\4\\2\\2\\4\\4\\4\\4\\4\\2\\2\\4\\2\\2\\4\\4\\2\\2\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\end{pmatrix}\right|$$

$$\left|\begin{pmatrix}4\\2\\2\\2\\2\\2\\2\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\2\\2\\4\\4\\4\\4\\2\\2\\2\\2\\4\\2\\2\\4\\4\\4\\4\\2\\4\\4\\2\\4\\4\\2\\4\end{pmatrix}\begin{pmatrix}4\\2\\2\\2\\2\\2\\2\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\2\\2\\4\\4\\4\\4\\2\\2\\2\\2\\4\\2\\2\\4\\4\\4\\4\\2\\4\\4\\2\\4\\4\\2\\4\end{pmatrix}\right|$$

$$\left|\begin{pmatrix}4\\2\\4\\4\\2\\4\\2\\2\\4\\4\\4\\4\\4\\2\\2\\4\\2\\2\end{pmatrix}\begin{pmatrix}4\\2\\4\\4\\2\\4\\2\\2\\4\\4\\4\\4\\4\\2\\2\\4\\2\\2\end{pmatrix}\right|$$

$$\left|\begin{array}{cc}
\begin{pmatrix}4\\4\\2\\2\\4\\4\\4\\4\\4\\4\\4\\4\\4\\2\\2\\2\\2\\2\\2\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\2\\2\\4\\4\\4\\4\\2\\2\\4\\2\\2\\4\\4\\4\\4\\2\\4\\4\\2\\4\\4\\2\\4\end{pmatrix} & \begin{pmatrix}4\\4\\2\\2\\4\\4\\4\\4\\4\\4\\4\\4\\4\\2\\2\\2\\2\\2\\2\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\2\\2\\4\\4\\4\\4\\2\\2\\4\\2\\2\\4\\4\\4\\4\\2\\4\\4\\2\\4\\4\\2\\4\end{pmatrix} \\
\begin{pmatrix}4\\2\\4\\4\\2\\4\end{pmatrix} & \begin{pmatrix}4\\2\\4\\4\\2\\4\end{pmatrix}
\end{array}\right|$$

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

$$\left|\begin{array}{cc}
\begin{pmatrix} 4 \\ 2 \\ 4 \\ 4 \\ 2 \\ 4 \end{pmatrix} & \begin{pmatrix} 4 \\ 2 \\ 4 \\ 4 \\ 2 \\ 4 \end{pmatrix} \\
\begin{pmatrix} 4 \\ 2 \\ 4 \\ 4 \\ 2 \\ 4 \\ 2 \\ 2 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 2 \\ 2 \\ 4 \\ 2 \\ 2 \\ 4 \\ 4 \\ 2 \\ 2 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 2 \\ 2 \\ 2 \\ 2 \\ 2 \\ 2 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 2 \\ 2 \\ 4 \\ 4 \\ 4 \\ 4 \\ 2 \\ 2 \\ 2 \\ 2 \end{pmatrix} & \begin{pmatrix} 4 \\ 2 \\ 4 \\ 4 \\ 2 \\ 4 \\ 2 \\ 2 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 2 \\ 2 \\ 4 \\ 2 \\ 2 \\ 4 \\ 4 \\ 2 \\ 2 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 2 \\ 2 \\ 2 \\ 2 \\ 2 \\ 2 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 2 \\ 2 \\ 4 \\ 4 \\ 4 \\ 4 \\ 2 \\ 2 \\ 2 \\ 2 \end{pmatrix}
\end{array}\right|$$

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

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

$$\left|\begin{pmatrix} 4 \\ 4 \\ 4 \\ 2 \\ 2 \\ 4 \\ 4 \\ 4 \\ 4 \\ 2 \\ 2 \\ 2 \\ 2 \\ 4 \\ 2 \\ 2 \\ 4 \\ 4 \\ 4 \\ 4 \\ 2 \\ 4 \\ 4 \\ 2 \\ 4 \\ 4 \\ 2 \\ 4 \\ 4 \\ 2 \\ 4 \end{pmatrix} \begin{pmatrix} 4 \\ 4 \\ 4 \\ 2 \\ 2 \\ 4 \\ 4 \\ 4 \\ 4 \\ 2 \\ 2 \\ 2 \\ 2 \\ 4 \\ 2 \\ 2 \\ 4 \\ 4 \\ 4 \\ 4 \\ 2 \\ 4 \\ 4 \\ 2 \\ 4 \\ 4 \\ 2 \\ 4 \\ 4 \\ 2 \\ 4 \end{pmatrix}\right|$$

$$\left|\begin{pmatrix} 4 \\ 2 \\ 4 \\ 4 \\ 2 \\ 4 \\ 2 \\ 2 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 2 \\ 2 \\ 4 \\ 2 \\ 2 \\ 4 \\ 4 \\ 2 \\ 2 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 2 \\ 2 \\ 2 \\ 2 \end{pmatrix} \begin{pmatrix} 4 \\ 2 \\ 4 \\ 4 \\ 2 \\ 4 \\ 2 \\ 2 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 2 \\ 2 \\ 4 \\ 2 \\ 2 \\ 4 \\ 4 \\ 2 \\ 2 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 2 \\ 2 \\ 2 \\ 2 \end{pmatrix}\right|$$

$$\left|\begin{pmatrix} 2 \\ 2 \\ 2 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 2 \\ 2 \\ 4 \\ 4 \\ 4 \\ 4 \\ 2 \\ 2 \\ 2 \\ 2 \\ 4 \\ 2 \\ 2 \\ 4 \\ 4 \\ 4 \\ 4 \\ 2 \\ 4 \\ 4 \\ 2 \\ 4 \\ 4 \\ 2 \\ 4 \end{pmatrix} \begin{pmatrix} 2 \\ 2 \\ 2 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 2 \\ 2 \\ 4 \\ 4 \\ 4 \\ 4 \\ 2 \\ 2 \\ 2 \\ 2 \\ 4 \\ 2 \\ 2 \\ 4 \\ 4 \\ 4 \\ 4 \\ 2 \\ 4 \\ 4 \\ 2 \\ 4 \\ 4 \\ 2 \\ 4 \end{pmatrix}\right|$$

$$\left|\begin{pmatrix} 4 \\ 2 \\ 4 \\ 4 \\ 2 \\ 4 \\ 2 \\ 2 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 2 \\ 2 \\ 4 \\ 2 \\ 2 \\ 4 \\ 4 \\ 2 \\ 2 \\ 4 \end{pmatrix} \begin{pmatrix} 4 \\ 2 \\ 4 \\ 4 \\ 2 \\ 4 \\ 2 \\ 2 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 2 \\ 2 \\ 4 \\ 2 \\ 2 \\ 4 \\ 4 \\ 2 \\ 2 \\ 4 \end{pmatrix}\right|$$

$$\left|\begin{pmatrix}4\\4\\4\\4\\4\\4\\4\\4\\4\\2\\2\\2\\2\\2\\2\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\2\\2\\4\\4\\4\\4\\2\\2\\2\\2\\4\\2\\2\\4\\4\\4\\4\\2\\4\\4\\2\\4\\4\\2\\4\\4\\2\\4\end{pmatrix}\begin{pmatrix}4\\4\\4\\4\\4\\4\\4\\4\\4\\2\\2\\2\\2\\2\\2\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\2\\2\\4\\4\\4\\4\\2\\2\\2\\2\\4\\2\\2\\4\\4\\4\\4\\2\\4\\4\\2\\4\\4\\2\\4\\4\\2\\4\end{pmatrix}\right|$$

$$\left|\begin{pmatrix}4\\2\\4\\4\\2\\4\\2\\2\\4\\4\end{pmatrix}\begin{pmatrix}4\\2\\4\\4\\2\\4\\2\\2\\4\\4\end{pmatrix}\right|$$

	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	
	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	
	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}}
```

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

$$\begin{pmatrix} \begin{pmatrix} 4 \\ 2 \\ 4 \\ 4 \\ 2 \\ 4 \\ 2 \\ 2 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 2 \\ 2 \\ 2 \\ 2 \\ 2 \\ 2 \\ 2 \\ 2 \\ 4 \\ 2 \\ 2 \\ 4 \\ 2 \\ 2 \\ 4 \\ 4 \\ 4 \\ 2 \\ 2 \\ 2 \\ 2 \\ 2 \\ 2 \\ 4 \\ 2 \\ 2 \\ 4 \\ 4 \\ 4 \\ 2 \\ 2 \\ 4 \\ 4 \\ 4 \\ 2 \\ 2 \\ 2 \\ 2 \\ 4 \\ 2 \\ 2 \\ 4 \\ 2 \end{pmatrix} & \begin{pmatrix} 4 \\ 2 \\ 4 \\ 4 \\ 2 \\ 4 \\ 2 \\ 2 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 4 \\ 2 \\ 2 \\ 2 \\ 2 \\ 2 \\ 2 \\ 2 \\ 2 \\ 4 \\ 2 \\ 2 \\ 4 \\ 2 \\ 2 \\ 4 \\ 4 \\ 4 \\ 2 \\ 2 \\ 2 \\ 2 \\ 2 \\ 2 \\ 4 \\ 2 \\ 2 \\ 4 \\ 4 \\ 4 \\ 2 \\ 2 \\ 4 \\ 4 \\ 4 \\ 2 \\ 2 \\ 2 \\ 2 \\ 4 \\ 2 \\ 2 \\ 4 \\ 2 \end{pmatrix} \end{pmatrix}$$

	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	
	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	
	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	
	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	
	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	
	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	
	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
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
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
2	2
2	2
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
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	
	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	
	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	
	2		2	

$$\left|\begin{array}{cc}
\begin{pmatrix}2\\4\\2\\2\\4\\4\\2\\2\\4\\4\\4\\4\\4\\4\\4\\4\\4\\2\\2\\2\\2\\2\\2\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\2\\2\\4\\4\\4\\4\\2\\2\\2\\2\\4\\2\\2\\4\\4\\4\\4\\2\\4\\4\\2\\4\\4\\2\\4\\4\\2\\4\\4\\2\\4\\4\\2\\4\end{pmatrix} & \begin{pmatrix}2\\4\\2\\2\\4\\4\\2\\2\\4\\4\\4\\4\\4\\4\\4\\4\\4\\2\\2\\2\\2\\2\\2\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\2\\2\\4\\4\\4\\4\\2\\2\\2\\2\\4\\2\\2\\4\\4\\4\\4\\2\\4\\4\\2\\4\\4\\2\\4\\4\\2\\4\\4\\2\\4\\4\\2\\4\end{pmatrix} \\
\begin{pmatrix}4\\2\end{pmatrix} & \begin{pmatrix}4\\2\end{pmatrix}
\end{array}\right|$$

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

$$\left|\begin{pmatrix}4\\4\\2\\2\\4\\4\\4\\4\\4\\4\\4\\4\\4\\2\\2\\2\\2\\2\\2\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\2\\2\\4\\4\\4\\4\\2\\2\\2\\2\\4\\2\\2\\4\\4\\4\\4\\2\\4\\4\\2\\4\\4\\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		
	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		
	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		
	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
	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
	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
	2		2
	2		2
	4		4
	2		2
	2		2
	4		4
	4		4
	2		2
	2		2

$$\left|\begin{pmatrix}4\\4\\4\\4\\4\\4\\4\\4\\4\\2\\2\\2\\2\\2\\2\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\2\\2\\4\\4\\4\\4\\2\\2\\2\\2\\4\\2\\2\\4\\4\\4\\4\\2\\4\\4\\2\\4\\4\\2\\4\\4\\2\\4\end{pmatrix}\begin{pmatrix}4\\4\\4\\4\\4\\4\\4\\4\\4\\2\\2\\2\\2\\2\\2\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\4\\2\\2\\4\\4\\4\\4\\2\\2\\2\\2\\4\\2\\2\\4\\4\\4\\4\\2\\4\\4\\2\\4\\4\\2\\4\\4\\2\\4\end{pmatrix}\right|$$

$$\left|\begin{pmatrix}4\\2\\4\\4\\2\\4\\2\\2\\4\\4\end{pmatrix}\begin{pmatrix}4\\2\\4\\4\\2\\4\\2\\2\\4\\4\end{pmatrix}\right|$$

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
2	2
2	2
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
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		
	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		
	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		
	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}, {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}, {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}}}
```

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

```
{{{1, 4, 2, 2, 4, 4, 2, 2, 2, 2, 4, 4, 4, 2, 2, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,
   4, 4, 2, 2, 2, 2, 4, 2, 2, 4, 4, 2, 2, 2, 2, 4, 2, 2, 4, 2, 4, 4, 2, 4, 4, 2, 4,
   4, 2, 4, 4, 2, 4, 4, 2, 4, 2, 2, 4, 4, 4, 4, 4, 2, 2, 4, 2, 2, 4, 4, 2, 2, 4, 4,
   4, 4, 4, 4, 4, 4, 4, 2, 2, 2, 2, 2, 2, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 2, 2,
   4, 4, 4, 4, 2, 2, 2, 2, 4, 2, 2, 4, 4, 4, 4, 2, 4, 4, 2, 4, 4, 2, 4, 4, 2, 4},
  {4, 2, 4, 4, 2, 4, 2, 2, 4, 4, 4, 4, 4, 2, 2, 4, 2, 2, 4, 4, 2, 2, 4, 4, 4, 4,
   4, 4, 4, 4, 4, 2, 2, 2, 2, 2, 2, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 2, 2,
   4, 4, 4, 4, 2, 2, 2, 2, 4, 2, 2, 4, 4, 4, 4, 2, 4, 4, 2, 4, 4, 2, 4, 4, 1}},
 {{1, 4, 2, 2, 4, 4, 2, 2, 2, 2, 4, 4, 4, 2, 2, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,
   4, 4, 4, 2, 2, 2, 2, 4, 2, 2, 4, 4, 2, 2, 2, 2, 4, 2, 2, 4, 2, 4, 4, 2, 4, 4, 2,
   4, 4, 2, 4, 4, 2, 4, 4, 2, 4, 2, 2, 4, 4, 4, 4, 4, 2, 2, 4, 2, 2, 4, 4, 2, 2, 4,
   4, 4, 4, 4, 4, 4, 4, 4, 2, 2, 2, 2, 2, 2, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 2,
   2, 4, 4, 4, 4, 2, 2, 2, 2, 4, 2, 2, 4, 4, 4, 4, 2, 4, 4, 2, 4, 4, 2, 4, 4, 2, 4},
  {4, 2, 4, 4, 2, 4, 2, 2, 4, 4, 4, 4, 4, 2, 2, 4, 2, 2, 4, 4, 2, 2, 4, 4, 4, 4,
   4, 4, 4, 4, 4, 2, 2, 2, 2, 2, 2, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 2, 2,
   4, 4, 4, 4, 2, 2, 2, 2, 4, 2, 2, 4, 4, 4, 4, 2, 4, 4, 2, 4, 4, 2, 4, 4, 1}}}
```

```
Function[l, And @@ (l ≠ # & /@ {1, 35, 77, 191})][35]
```

```
False
```

Parse runs

Shotgun mfx parser

```
mfxparser = 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, ⍋]}]]]
 ]
```

```
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

```
NestWhile[mfxparser @@ # &, {mediumruns[[1, 1]], {}}, Length[First[#]] > 0 &]
```

```
{{}, {SYNC, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0,
  0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, SYNC, SYNC}}
```

Shotgun MM parser

```
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, ⚠]}]]
 ]
```

```
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

```
NestWhile[MMparser @@ # &, {mmruns[[1, 1]], {}}, Length[First[#]] > 0 &]
```

```
{{}, {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

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

```
{{{SYNC, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, SYNC, SYNC},
  {SYNC, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, SYNC, SYNC},
  {SYNC, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, SYNC, SYNC},
  {SYNC, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, SYNC, SYNC},
  {SYNC, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, SYNC, SYNC},
  {SYNC, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, SYNC, SYNC},
  {SYNC, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, SYNC, SYNC},
  {SYNC, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, SYNC, SYNC},
  {SYNC, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, SYNC, SYNC},
  {SYNC, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, SYNC, SYNC},
  {SYNC, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, SYNC, SYNC},
  {SYNC, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, SYNC, SYNC}},
 {{SYNC, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, SYNC, SYNC},
  {SYNC, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, SYNC, SYNC},
  {SYNC, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, SYNC, SYNC},
  {SYNC, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, SYNC, SYNC},
  {SYNC, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, SYNC, SYNC},
  {SYNC, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, SYNC, SYNC},
  {SYNC, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, SYNC, SYNC},
  {SYNC, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, SYNC, SYNC},
  {SYNC, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, SYNC, SYNC},
  {SYNC, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, SYNC, SYNC},
  {SYNC, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, SYNC, SYNC},
  {SYNC, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, SYNC, SYNC}}}
```

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

```
{{{⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
   ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
   ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠},
  {⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
```

```
    ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠},
  {⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
    ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
    ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠},
  {⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
    ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
    ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠}}}
```

Parse over long runs

```
longbitsequences = Map[
  Last[NestWhile[mfxparser @@ # &, {#, {}}, Length[First[#]] > 0 &]] &, longruns, {2}]
```

```
{{{SYNC, 1, 0, 0, 0, 0, 0, 0, 0, 0, 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, 0, 0, 0, 1, 0, 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, 1, 1, 1, 0, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, SYNC, SYNC},
  {SYNC, 1, 0, 0, 0, 0, 0, 0, 0, 0, 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, 0, 0, 0, 1, 0, 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, 1, 1, 1, 0, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, SYNC, SYNC},
  {SYNC, 1, 0, 0, 0, 0, 0, 0, 0, 0, 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, 0, 0, 0, 1, 0, 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, 1, 1, 1, 0, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, SYNC, SYNC},
  {SYNC, 1, 0, 0, 0, 0, 0, 0, 0, 0, 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, 0, 0, 0, 1, 0, 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, 1, 1, 1, 0, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, SYNC, SYNC}},
 {{SYNC, 1, 0, 0, 0, 0, 0, 0, 0, 0, 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, 0, 0, 0, 1, 0, 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, 1, 1, 1, 0, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, SYNC, SYNC},
  {SYNC, 1, 0, 0, 0, 0, 0, 0, 0, 0, 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, 0, 0, 0, 1, 0, 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, 1, 1, 1, 0, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, SYNC, SYNC},
  {SYNC, 1, 0, 0, 0, 0, 0, 0, 0, 0, 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, 0, 0, 0, 1, 0, 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, 1, 1, 1, 0, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, SYNC, SYNC},
  {SYNC, 1, 0, 0, 0, 0, 0, 0, 0, 0, 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, 0, 0, 0, 1, 0, 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, 1, 1, 1, 0, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, SYNC, SYNC},
  {SYNC, 1, 0, 0, 0, 0, 0, 0, 0, 0, 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, 0, 0, 0, 1, 0, 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, 1, 1, 1, 0, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, SYNC, SYNC},
  {SYNC, 1, 0, 0, 0, 0, 0, 0, 0, 0, 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, 0, 0, 0, 1, 0, 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, 1, 1, 1, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, SYNC, SYNC},
  {SYNC, 1, 0, 0, 0, 0, 0, 0, 0, 0, 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, 0, 0, 0, 1, 0, 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, 1, 1, 1, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, SYNC, SYNC},
  {SYNC, 1, 0, 0, 0, 0, 0, 0, 0, 0, 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, 0, 0, 0, 1, 0, 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, 1, 1, 1, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, SYNC, SYNC}}}
```

```
Map[Last[NestWhile[MMparser @@ # &, {#, {}}, Length[First[#]] > 0 &]] &, longruns, {2}]
```

```
{{{⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
   ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
   ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
   ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
   ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
   ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
   ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠},
  {⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
   ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
   ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
   ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
   ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
   ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
   ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠},
  {⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
   ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
   ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
   ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
   ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
   ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠},
  {⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
   ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
   ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
   ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
   ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
   ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
   ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
   ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
   ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
```

```
    ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
    ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠}},
  {{⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
    ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
    ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
    ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
    ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
    ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
    ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠},
  {⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
    ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
    ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
    ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
    ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
    ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
    ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠},
  {⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
    ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
    ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
    ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
    ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
    ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
    ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠},
  {⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
    ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
    ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
    ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
    ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
    ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
    ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠}}}
```

Parse over MM runs

```
Map[Last[NestWhile[mfxparser @@ # &, {#, {}}, Length[First[#]] > 0 &]] &, mmruns, {2}]
```

```
{{{⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
   ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠}, {⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
   ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠},
  {⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
   ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠}, {⚠, 
```

```
mmbitsequences =
 Map[Last[NestWhile[MMparser @@ # &, {#, {}}, Length[First[#]] > 0 &]] &, mmruns, {2}]
```

```
{{{1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0},
  {1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0},
  {1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0},
  {1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0},
  {1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0},
  {1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0},
  {1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0},
  {1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0},
  {1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0},
  {1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0}},
 {{1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0},
  {1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0},
  {1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0},
  {1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0},
  {1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0},
  {1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0},
  {1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0},
  {1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0},
  {1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0},
  {1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0}}}
```

Parse over truncated runs

```
Map[Last[NestWhile[mfxparser @@ # &, {#, {}}, Length[First[#]] > 0 &]] &,
 truncruns, {2}]
```

```
{{{⚠, 0, 1, 0, 0, 1, 1, 0, 0, 0, 1, 0, 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, 1, 1, 1, 0, 0, 0, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, SYNC, SYNC},
  {SYNC, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, SYNC, 0, ⚠, 0, 0, ⚠}},
 {{⚠, 0, 1, 0, 0, 1, 1, 0, 0, 0, 1, 0, 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, 1, 1, 1, 0, 0, 0, 0, 0,
   0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, SYNC, SYNC},
  {SYNC, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, SYNC, 0, ⚠, 0, 0, ⚠}}}
```

```
Map[Last[NestWhile[MMparser @@ # &, {#, {}}, Length[First[#]] > 0 &]] &, truncruns, {2}]
```

```
{{{⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
   ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
   ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
   ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
   ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠},
  {⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
   ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
   ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, 0}},
 {{⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
   ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
   ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
   ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
   ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠},
  {⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
   ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠,
   ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, ⚠, 0}}}
```

Eliminate duplicate sequences

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

```
longbitseq = Flatten[Union[Union /@ longbitsequences], 2]
```

```
{SYNC, 1, 0, 0, 0, 0, 0, 0, 0, 0, 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, 0, 0, 0, 1, 0, 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, 1, 1, 1, 0, 0,
 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, SYNC, SYNC}
```

```
mediumbitseq = Flatten[Union[Union /@ mediumbitsequences], 2]
```

```
{SYNC, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0,
 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, SYNC, SYNC}
```

```
mmbitseq = Flatten[Union[Union /@ mmbitsequences], 2]
```

```
{1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0}
```

Decode MM packets

Motorola format bit

```
mmbit = Function[{in, out}, If[Length[in] < 2, {Drop[in], Append[out, ERROR]},
   {Drop[in, 2], Append[out, Switch[Take[in, 2], {0, 0}, 0, {1, 1}, 1, _, ⚠]]}]]
```

```
Function[{in, out}, If[Length[in] < 2,
  {Drop[in], Append[out, ERROR]}, {Drop[in, 2], Append[out, Switch[Take[in, 2],
     {0, 0}, 0,
     {1, 1}, 1,
     _, ⚠]]}]]
```

Motorola format trit

```
mmtrit = Function[{in, out},
  If[Length[in] < 2, {Drop[in], Append[out, ERROR]}, {Drop[in, 2],
    Append[out, Switch[Take[in, 2], {0, 0}, 0, {1, 1}, 1, {1, 0}, 2, _, ⚠]]}
  ]
 ]
```

```
Function[{in, out}, If[Length[in] < 2,
  {Drop[in], Append[out, ERROR]}, {Drop[in, 2], Append[out, Switch[Take[in, 2],
     {0, 0}, 0,
     {1, 1}, 1,
     {1, 0}, 2,
     _, ⚠]]}]]
```

Motorola format 4-trit address (address 80, all open, is unassigned)

```
mmaddress = Function[{in, out}, If[Length[in] < 8, {Drop[in], Append[out, ERROR]},
   ({#1[[1]], Append[out, {ADDR, FromDigits[Reverse[Last[#1]], 3]}]} &) [
    Nest[mmtrit @@ #1 &, {in, out}, 4]]]]
```

```
Function[{in, out}, If[Length[in] < 8, {Drop[in], Append[out, ERROR]},
  ({#1[[1]], Append[out, {ADDR, FromDigits[Reverse[Last[#1]], 3]}]} &) [
   Nest[mmtrit @@ #1 &, {in, out}, 4]]]]
```

```
mmaddress[mmbitseq, {}]
```

```
{{0, 0, 1, 0, 1, 0, 1, 0, 1, 0}, {{ADDR, 80}}}
```

Motorola format parser

4-trit address, one function (double) bit, 4 velocity (double) bits

```
mm1parse =
 Function[{in, out},
  ({#1, Append[Drop[#2, -4], {SPEED, FromDigits[Reverse[Take[#2, -4]], 2]}]} &) @@
   Nest[mmbit @@ #1 &, ({#1[[1]], Append[Most[#1[[2]]], {FUNC, Last[#1[[2]]]}]} &) [
     mmbit @@ mmaddress[in, out]], 4]]
```

```
Function[{in, out},
 ({#1, Append[Drop[#2, -4], {SPEED, FromDigits[Reverse[Take[#2, -4]], 2]}]} &) @@
  Nest[mmbit @@ #1 &, ({#1[[1]], Append[Most[#1[[2]]], {FUNC, Last[#1[[2]]]}]} &) [
    mmbit @@ mmaddress[in, out]], 4]]
```

MM2 format

```
mm2dataparse = Function[{in, out}, If[8 ≠ Length[in], {in, Append[out, ERROR]},
    Block[{speed = FromDigits[in[[{7, 5, 3, 1}]], 2],
      data = in[[{2, 4, 6, 8}]], tag}, tag = Which[
       data == {1, 0, 1, 0} ∧ speed > 7, {REVERSE},
       data == {1, 0, 1, 1} ∧ speed < 8, {REVERSE},
       data == {0, 1, 0, 1} ∧ speed < 8, {FORWARD},
       data == {0, 1, 0, 0} ∧ speed > 7, {FORWARD},
       True, {DATA, data}];
     {Drop[in, 8], Join[out, {{SPEED, speed}, tag}]}
    ]
   ]
  ]
```

```
Function[{in, out}, If[8 ≠ Length[in], {in, Append[out, ERROR]},
  Block[{speed = FromDigits[in[[{7, 5, 3, 1}]], 2], data = in[[{2, 4, 6, 8}]], tag},
   tag = Which[data == {1, 0, 1, 0} && speed > 7, {REVERSE},
     data == {1, 0, 1, 1} && speed < 8, {REVERSE}, data == {0, 1, 0, 1} && speed < 8,
     {FORWARD}, data == {0, 1, 0, 0} && speed > 7, {FORWARD}, True, {DATA, data}];
   {Drop[in, 8], Join[out, {{SPEED, speed}, tag}]}]]]
```

```
mm2parse =
 Function[{in, out},
  mm2dataparse @@ (({#1[[1]], Append[Most[#1[[2]]], {FUNC, Last[#1[[2]]]}]} &) [
     mmbit @@ mmaddress[in, out]])]
```

```
Function[{in, out}, mm2dataparse @@
  ({#1[[1]], Append[Most[#1[[2]]], {FUNC, Last[#1[[2]]]}]} &) [mmbit @@ mmaddress[in, out]]]
```

Parse Motorola packet

```
Last[mm1parse[mmbitseq, {}]]
```

```
{{ADDR, 80}, {FUNC, 0}, {SPEED, 15 ⚠}}
```

The packet sent by the controller has an illegal address (all open) and illegal (open) velocity information.

```
Last[mm2parse[mmbitseq, {}]]
```

```
{{ADDR, 80}, {FUNC, 0}, {SPEED, 15}, {DATA, {0, 0, 0, 0}}}
```

Parse delimited mfx/DCC packets

Split by SYNC sequences

```
longbits = Select[SplitBy[longbitseq, NumberQ], Union[#] ≠ {SYNC} &]
```

```
{{1, 0, 0, 0, 0, 0, 0, 0, 0, 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, 0, 0, 0, 1,
  0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 1, 1, 0, 1},
 {1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0,
  0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0}}
```

```
mediumbits = Select[SplitBy[mediumbitseq, NumberQ], Union[#] ≠ {SYNC} &]
```

```
{{1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1,
  1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0}}
```

CRC function for mfx

```
mfxCRC = Function[{crc, datum},
  Block[{acc = BitOr[2 crc, datum]}, BitAnd[BitXor[acc, If[acc > 255, 7, 0]], 255]
  ]
 ]
```

```
Function[{crc, datum},
 Block[{acc = BitOr[2 crc, datum]}, BitAnd[BitXor[acc, If[acc > 255, 7, 0]], 255]]]
```

```
Fold[mfxCRC, 127, #] & /@ mediumbits
```

```
{0}
```

```
Fold[mfxCRC, 127, #] & /@ longbits
```

```
{0, 0}
```

```
mediumseries = Drop[#, -8] & /@ mediumbits
```

```
{{1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0,
  0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0}}
```

```
longseries = Drop[#, -8] & /@ longbits
```

```
{{1, 0, 0, 0, 0, 0, 0, 0, 0, 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, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1},
 {1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0,
  1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0}}
```

```
# == First[mediumseries] & /@ longseries
```

```
{False, True}
```

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

Address parser

Parse the different mfx address formats

```
addressanalyzer = Function[{in, out},
  Which[Length[in] ≥ 18 ∧ {1, 1, 1, 1} == Take[in, 4],
   {CMD, Drop[in, 18], Append[out, {A14, FromDigits[Take[in, {5, 18}], 2]}]},
   Length[in] ≥ 15 ∧ {1, 1, 1, 0} == Take[in, 4],
   {CMD, Drop[in, 15], Append[out, {A11, FromDigits[Take[in, {5, 15}], 2]}]},
   Length[in] ≥ 12 ∧ {1, 1, 0} == Take[in, 3],
   {CMD, Drop[in, 12], Append[out, {A9, FromDigits[Take[in, {4, 12}], 2]}]},
   Length[in] ≥ 9 ∧ {1, 0} == Take[in, 2],
   {CMD, Drop[in, 9], Append[out, {A7, FromDigits[Take[in, {3, 9}], 2]}]},
   True, {ERROR, in, Append[out, ERROR]}
  ]
 ]
```

```
Function[{in, out}, Which[Length[in] ≥ 18 && {1, 1, 1, 1} == Take[in, 4],
  {CMD, Drop[in, 18], Append[out, {A14, FromDigits[Take[in, {5, 18}], 2]}]},
  Length[in] ≥ 15 && {1, 1, 1, 0} == Take[in, 4],
  {CMD, Drop[in, 15], Append[out, {A11, FromDigits[Take[in, {5, 15}], 2]}]},
  Length[in] ≥ 12 && {1, 1, 0} == Take[in, 3],
  {CMD, Drop[in, 12], Append[out, {A9, FromDigits[Take[in, {4, 12}], 2]}]},
  Length[in] ≥ 9 && {1, 0} == Take[in, 2],
  {CMD, Drop[in, 9], Append[out, {A7, FromDigits[Take[in, {3, 9}], 2]}]},
  True, {ERROR, in, Append[out, ERROR]}]]
```

```
addressanalyzer[#, {}] & /@ longseries
```

```
{{CMD, {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, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1}, {{A7, 0}}},
 {CMD, {0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0}, {{A7, 5}}}}
```

configuration parser

Parse the subcommands of the mfx configuration command

```
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}}
   ]
  ]
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

```
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]}]]
```


```
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

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

```
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

```
NestWhile[mfxanalyzer @@ # &, {ADDR, #, {}}, Length[#[[2]]] > 0 &] & /@ longseries
```

```
{{CMD, {}, {{A7, 0}, {CENTRALUID, 1195666018, COUNT, 3}}}, {CMD, {},
  {{A7, 5}, {DRV7, FWD, 0}, {FUNC16, {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0}}}}}
```

```
Print[MatrixForm[Last[NestWhile[mfxanalyzer @@ # &,
      {ADDR, #, {}}, Length[#[[2]]] > 0 &]]] & /@ longseries;
```

$$\begin{pmatrix} \{A7, 0\} \\ \{CENTRALUID, 1195666018, COUNT, 3\} \end{pmatrix}$$

$$\begin{pmatrix} \{A7, 5\} \\ \{DRV7, FWD, 0\} \\ \{FUNC16, \{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0\}\} \end{pmatrix}$$