

# Анализ инженерных данных в MATLAB

Проанализируем сигналы со спутника

## Содержание

Загрузка данных.....	1
Предобработка сигналов.....	2
Визуализация сигнала.....	6
Анализ сигнала.....	7

## Загрузка данных

Считаем сигналы из Excel-файла

```
S = readtable('signals.xlsx');
summary(S)
```

Variables:

Time: 17525×1 datetime

Values:

Min	15.08.2017	19:37:26.759
Median	15.08.2017	20:15:11.007
Max	15.08.2017	20:53:01.362

Ch1: 17525×1 double

Values:

Min	-481
Median	-5
Max	480
NumMissing	1

Ch2: 17525×1 double

Values:

Min	-98
Median	-4
Max	148
NumMissing	15

Ch3: 17525×1 double

Values:

Min	-1500
Median	-50
Max	1000
NumMissing	10

Ch4: 17525×1 double

Values:

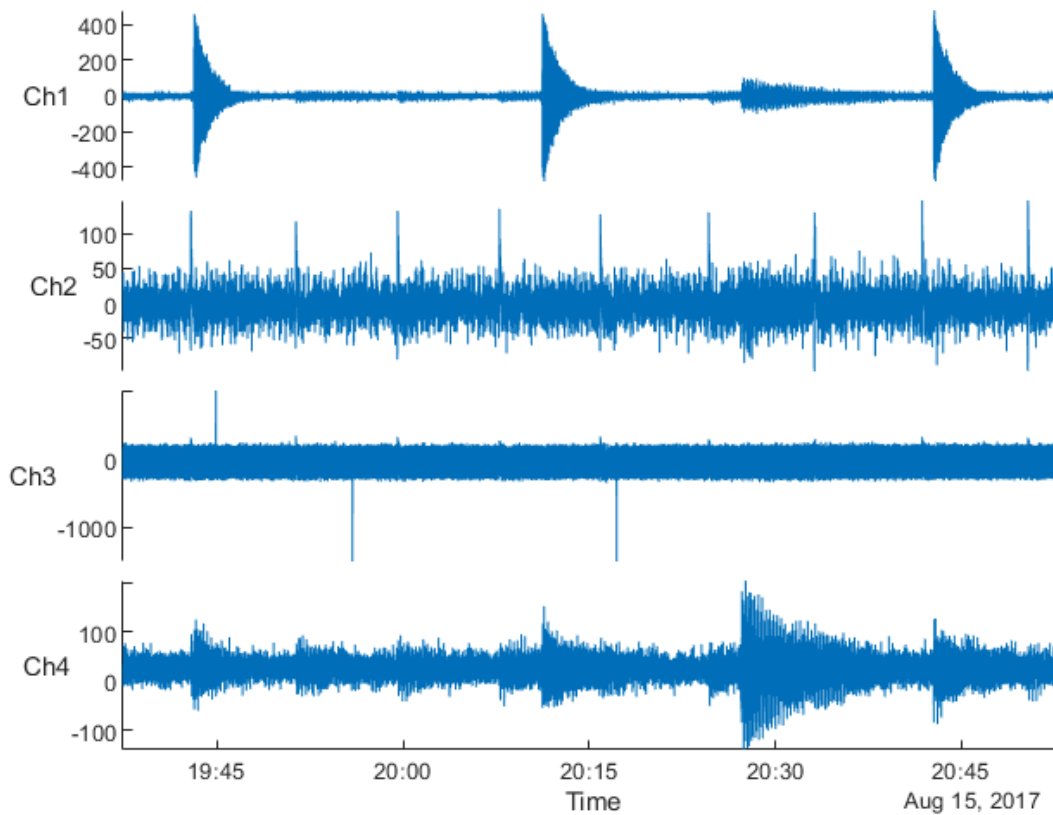
Min	-139
Median	25
Max	204
NumMissing	4

Переведем данные в формат Timetable

```
S = table2timetable(S);
```

Визуализируем данные таблицы

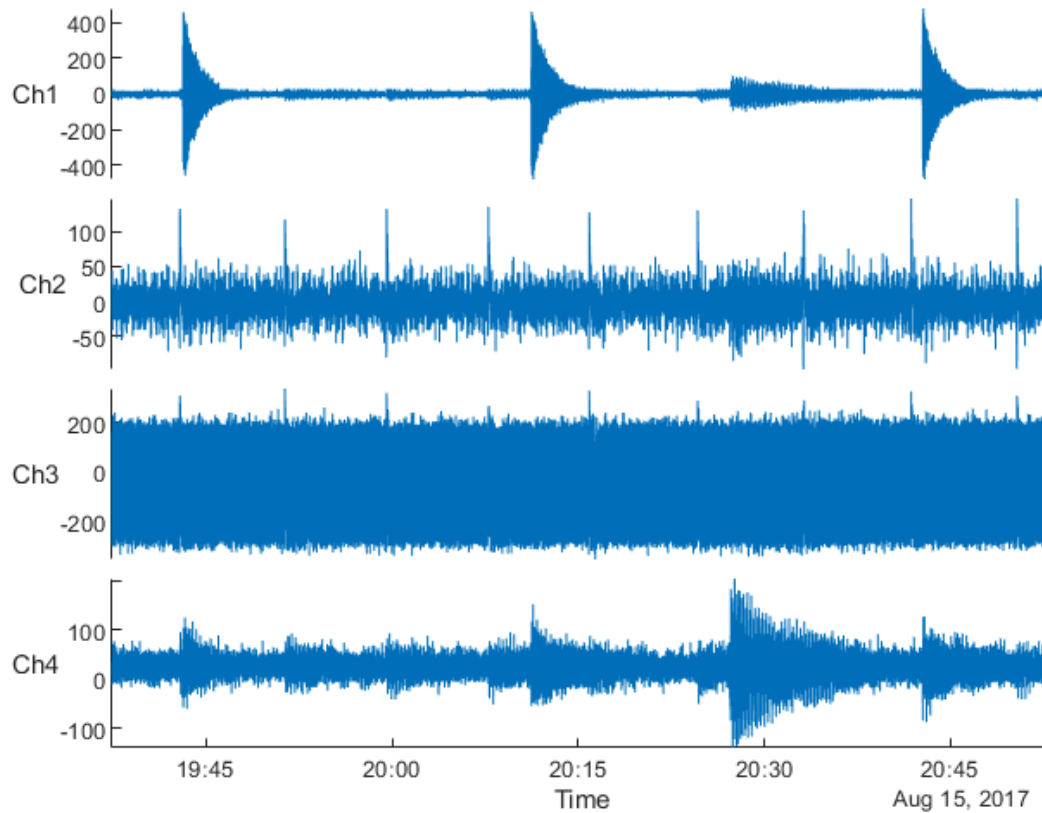
```
stackedplot(S);
```



## Предобработка сигналов

На графике сигнала Ch3 отчетливо видно несколько выбросов. Заполним их интерполированными значениями

```
S.Ch3 = filloutliers(S.Ch3, 'linear');  
stackedplot(S);
```



Попробуем посчитать стандартное отклонение сигнала

```
devs = std(S{:, :})
```

```
devs = 1×4
      NaN      NaN      NaN      NaN
```

Пропущенные значения (NaN) мешают работать с сигналами

Заменяем пропущенные значения

```
S = fillmissing(S, 'linear');
```

Теперь можно посчитать статистические характеристики сигналов

```
means = mean(S{:, :})
```

```
means = 1×4
-1.7816 -2.8401 -47.4544 25.1878
```

```
devs = std(S{:, :})
```

```
devs = 1×4
```

52.8533    20.2171    174.8782    26.8842

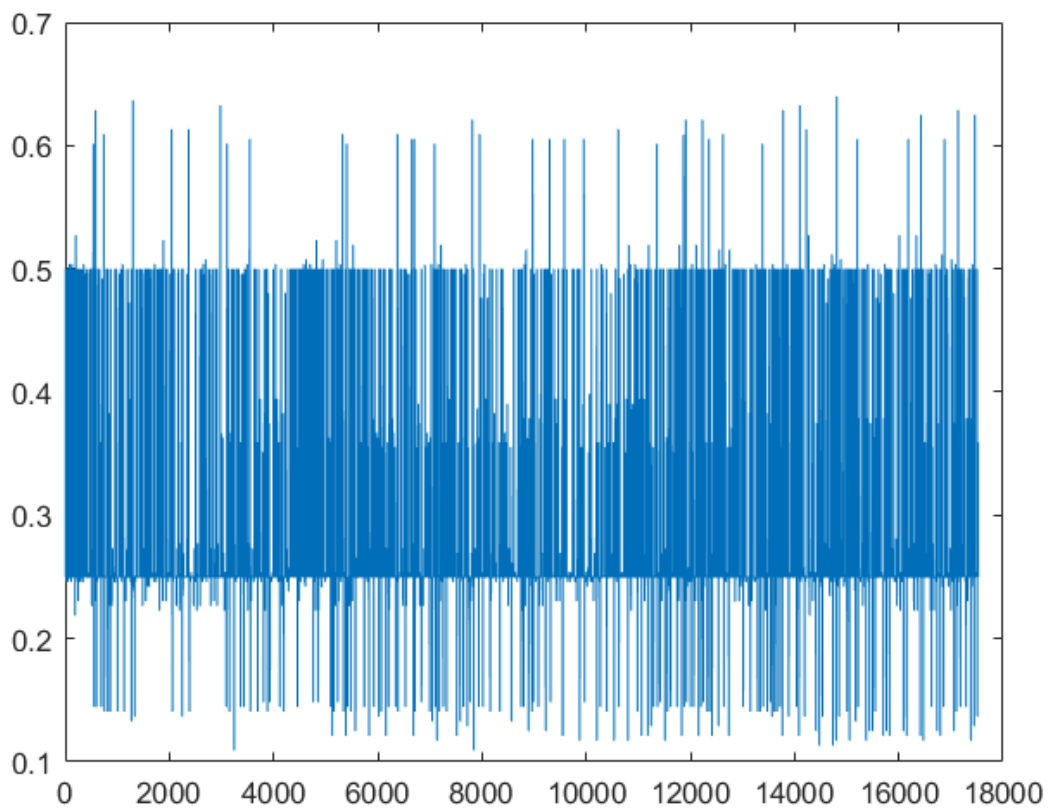
Проверим, что такт, с которым сняты сигналы, постоянен

```
isregular(S)
```

```
ans = logical  
      0
```

Найдем величину такта

```
step = seconds(diff(S.Time));  
figure  
plot(step)
```



Значение моды такта

```
mode(step)
```

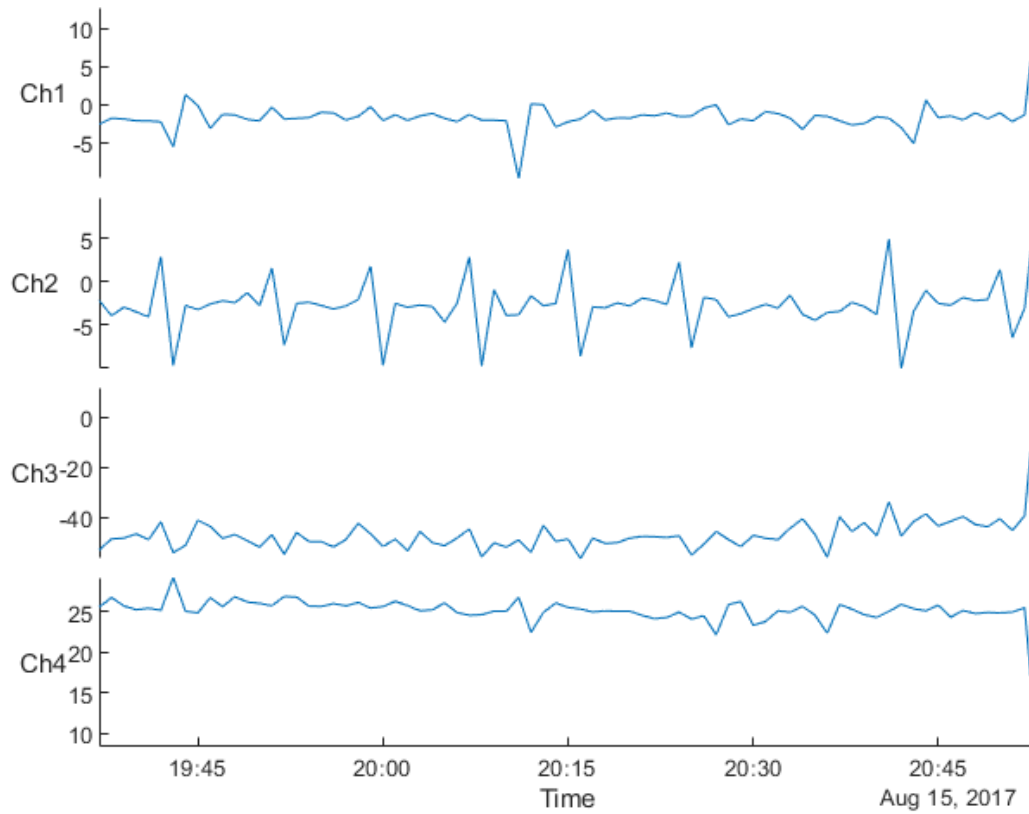
```
ans = 0.2500
```

Передискретизируем сигналы с тактом 0,25 с

```
S = retime(S, 'regular', 'linear', 'TimeStep', seconds(0.25));
```

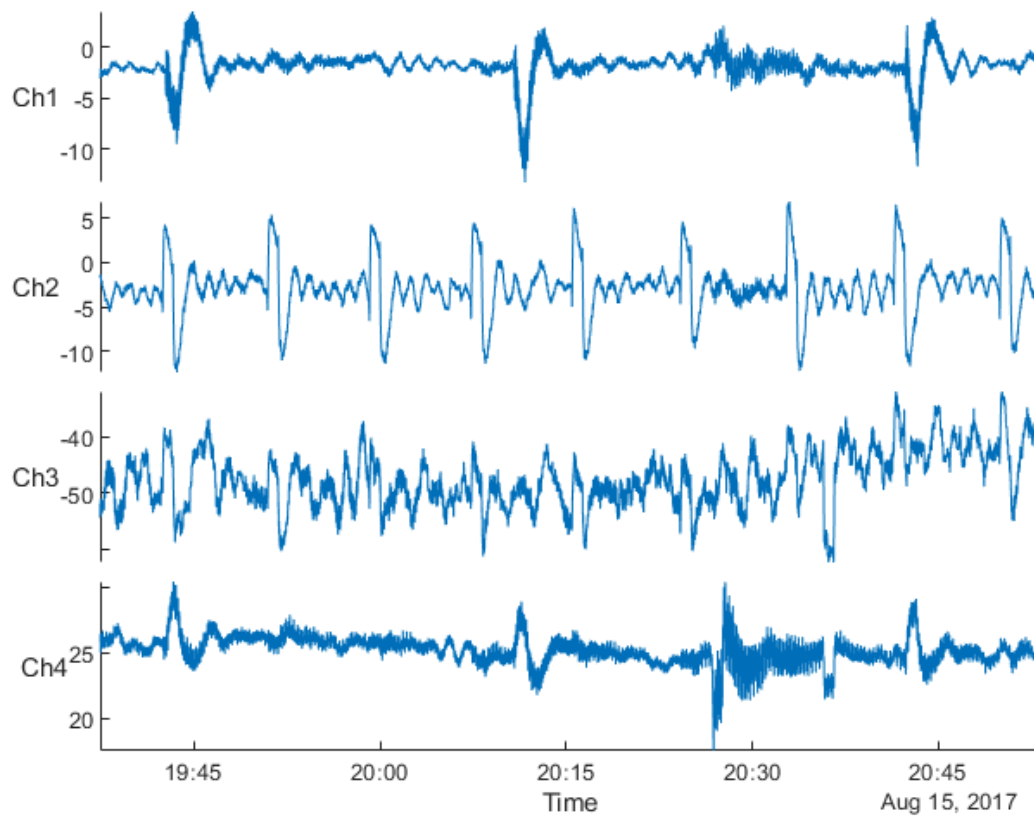
Передискретизировать можно и с осреднением

```
Sm = retime(S, 'minutely', 'mean');  
stackedplot(Sm);
```



Осредним сигналы методом скользящего среднего

```
S2 = S;  
S2{:, :} = movmean(S2{:, :}, 200);  
stackedplot(S2);
```



## Визуализация сигнала

Построим график второго сигнала

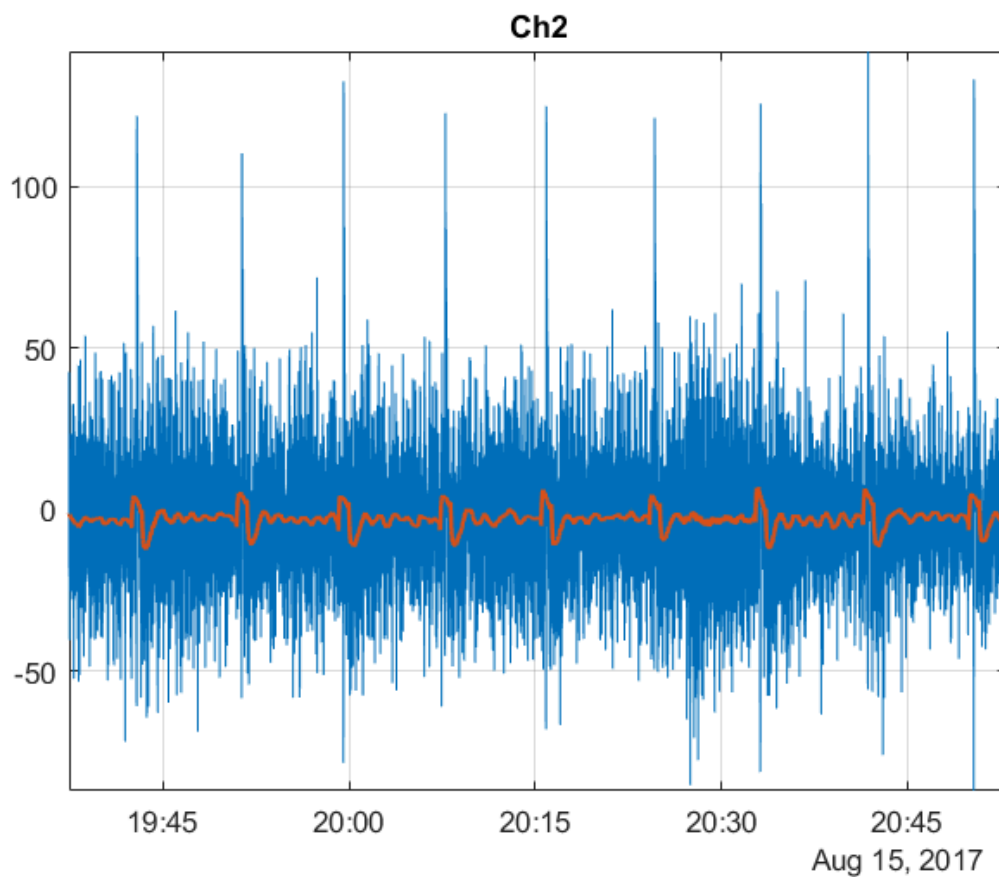
```
figure
plot(S.Time, S.Ch2);
```

Наложим сверху осредненный график

```
hold on
plot(S2.Time, S2.Ch2, 'LineWidth', 1.5);
hold off
```

Настроим график

```
title('Ch2')
grid on
axis tight
```



## Анализ сигнала

(*Signal Processing Toolbox*)

Переведем вектор времени в формат `duration` (длительность)

```
S.Time = S.Time - S.Time(1);
```

Запустим инструмент *Signal Analyzer*

```
signalAnalyzer(S)
```

Pavel Roslovets, [ETMC Exponenta](#) © 2018