

# Измерение параметров механической колебательной системы

Ревков Сергей

МГУ имени М.В. Ломоносова

# Постановка задачи

С помощью встроенного датчика угловой скорости устройство должно измерять период колебаний математического маятника (в качестве маятника выступает плата, подвешенная на относительно длинном проводе). На основе полученного периода колебаний и известного значения ускорения свободного падения должна определяться длина подвеса маятника. Результат измерений должен выводиться на дисплей.

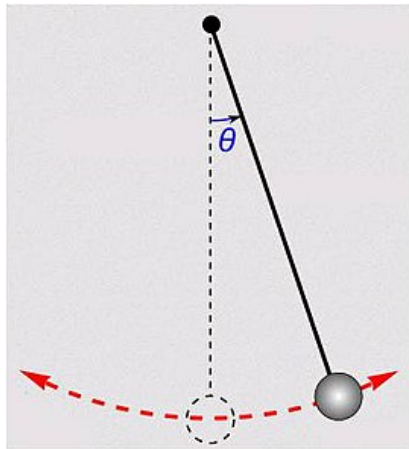
# Математический маятник

Математический маятник — осциллятор, представляющий собой механическую систему, состоящую из материальной точки на конце невесомой нерастяжимой нити или лёгкого стержня и находящуюся в однородном поле сил тяготения. Период малых собственных колебаний маятника длины  $L$ , подвешенного в поле тяжести, равен

$$T = 2\pi\sqrt{\frac{L}{g}}$$

где  $L$  - длина маятника,  $g$  - ускорение свободного падения.

# Математический маятник

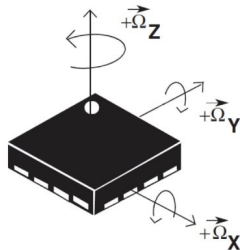


# Гироскоп (отладочная плата)



## ST L3GD20

- ▶ Three-axis angular rate sensor.
- ▶ Full scale of  $\pm 250/\pm 500/\pm 2000$  dps
- ▶ I2C/SPI digital output interface.
- ▶ Integrated low- and high-pass filters with user selectable bandwidth.
- ▶ Embedded temperature sensor.
- ▶ Embedded FIFO buffers.



# Гироскоп (характеристики)

**Table 4. Mechanical characteristics<sup>(1)</sup>**

Symbol	Parameter	Test condition	Min.	Typ. <sup>(2)</sup>	Max.	Unit
FS	Measurement range	User-selectable		±250		dps
				±500		
				±2000		
So	Sensitivity	FS = 250 dps		8.75		mdps/digit
		FS = 500 dps		17.50		
		FS = 2000 dps		70		
SoDr	Sensitivity change vs. temperature	From -40 °C to +85 °C		±2		%
DVoff	Digital zero-rate level	FS = 250 dps		±10		dps
		FS = 500 dps		±15		
		FS = 2000 dps		±75		
OffDr	Zero-rate level change vs. temperature	FS = 250 dps		±0.03		dps/°C
		FS = 2000 dps		±0.04		dps/°C
NL	Non linearity	Best fit straight line		0.2		% FS
Rn	Rate noise density			0.03		dps/(√Hz)
ODR	Digital output data rate			95/190/ 380/760		Hz
Top	Operating temperature range		-40		+85	°C

# Гироскоп (характеристики)

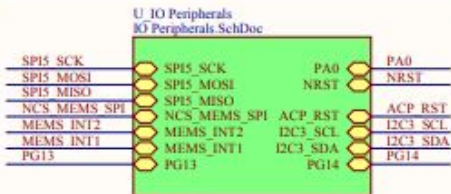
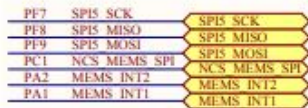
**Table 7. SPI slave timing values**

Symbol	Parameter	Value <sup>(1)</sup>		Unit
		Min	Max	
tc(SPC)	SPI clock cycle	100		ns
fc(SPC)	SPI clock frequency		10	MHz
tsu(CS)	CS setup time	5		ns
th(CS)	CS hold time	8		
tsu(SI)	SDI input setup time	5		
th(SI)	SDI input hold time	15		
tv(SO)	SDO valid output time		50	
th(SO)	SDO output hold time	6		
tdis(SO)	SDO output disable time		50	

1. Values are guaranteed at a 10 MHz clock frequency for SPI with both 4 and 3 wires, based on characterization results; not tested in production.



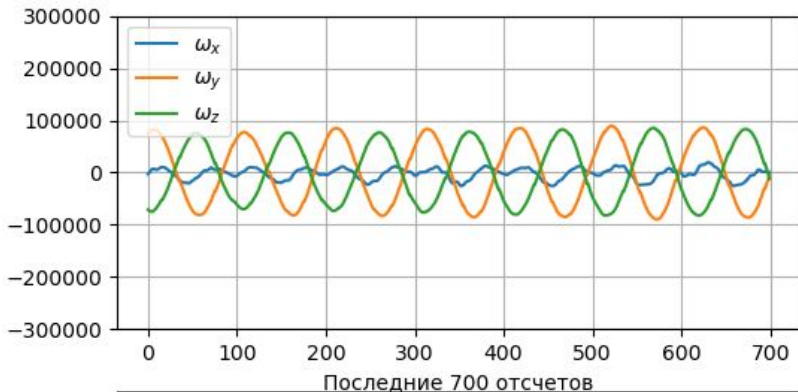
# Схема алгоритма



# Схема алгоритма

```
// Периферия
LL_APB2_GRP1_EnableClock(LL_APB2_GRP1_PERIPH_USART1);
LL_APB1_GRP1_EnableClock(LL_APB1_GRP1_PERIPH_TIM2);
// Инициализация
HAL_Init();
SystemClock_Config();
InitializeRCC_GPIO();
ConfigurationUSART();
ConfigurationTIM();
BSP_GYRO_Init();
BSP_LCD_Init();
// Получение данных
BSP_GYRO_GetXYZ(omega_xyz);
// Фильтрация
MEDIANFILTER_Init(&medianFilter);
// Период, длина, вывод на экран
period();
length();
display();
BSP_LCD_DisplayStringAtLine(row_d++, (unsigned char *)str);
```

# Гироскоп (данные)

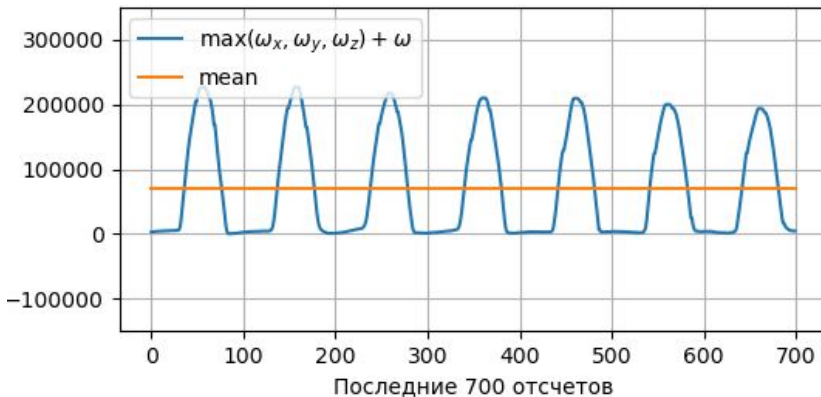


# Алгоритм

$$\text{Из } T = 2\pi\sqrt{\frac{L}{g}} \Rightarrow L = \frac{gT^2}{4\pi^2}.$$

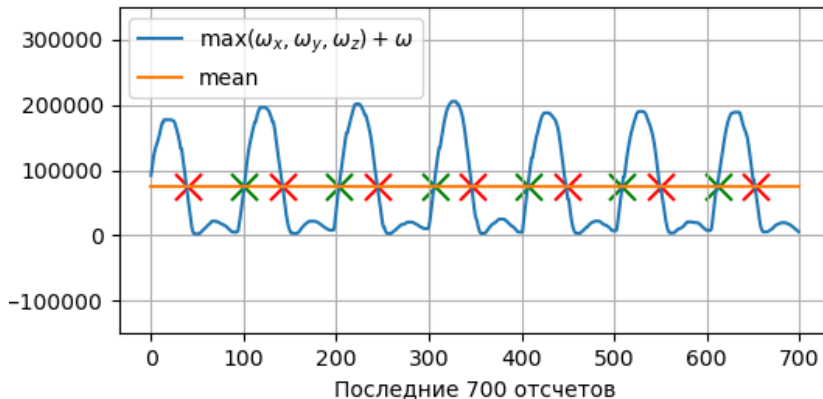
$g$  и  $\pi$  нам известно, остается найти  $T$ .

Предлагается следующий алгоритм. Сложим  $\max(\omega_x, \omega_y, \omega_z, )$  и  $\omega$  и в полученном массиве найдем среднее.



# Алгоритм

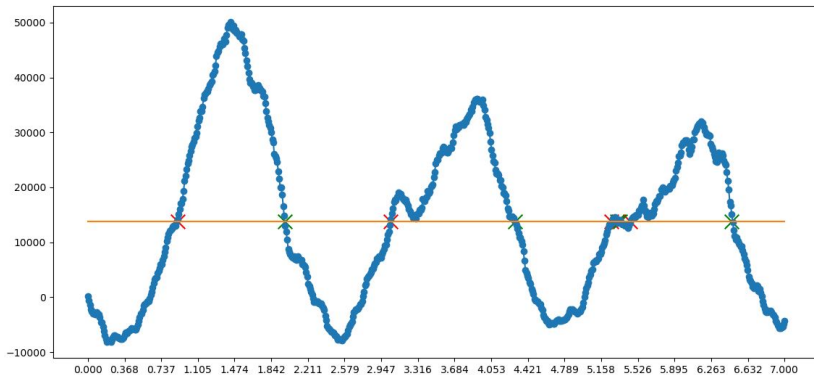
Далее, считаем сколько раз наш график пересекает среднее. Причем считаем два варианта - первый когда очередная точка стала меньше среднего и второй вариант, когда больше. Получается два массива  $T_1$  - красные точки и  $T_2$  - зелёные.



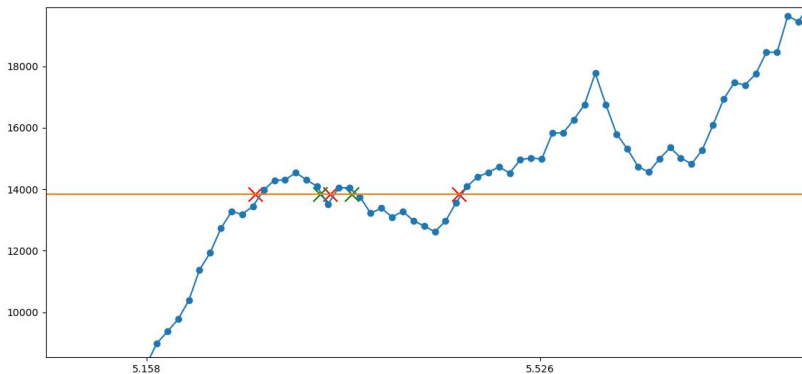
Затем находим среднее по красным и зелёным точкам -  $T_{1mean}$  и  $T_{2mean}$ . И нужный нам период рассчитываем так:

$$T = \frac{T_{1mean} + T_{2mean}}{2}$$

# Фильтрация

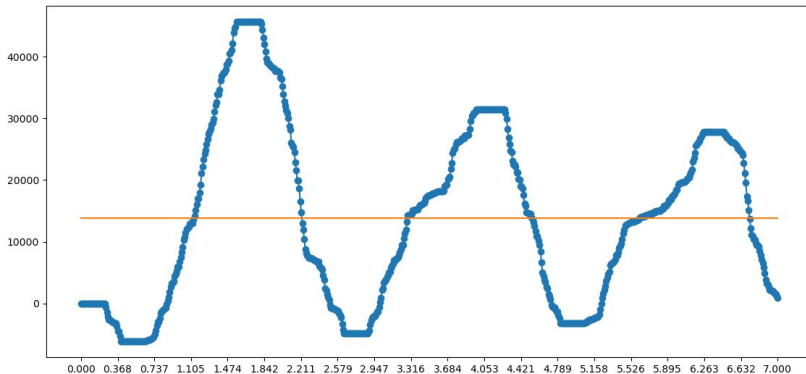


# Фильтрация

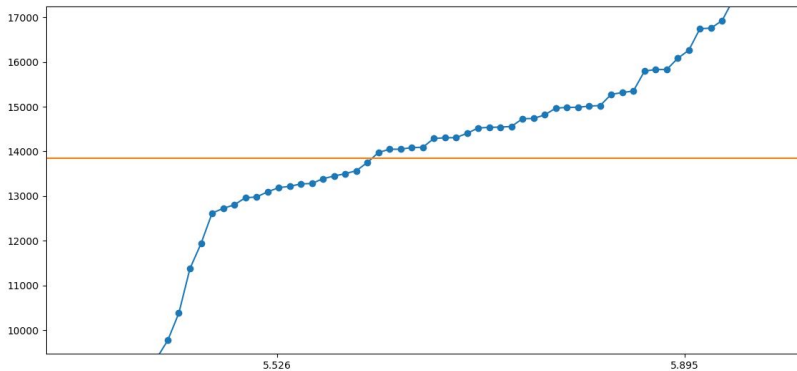




# Фильтрация








# Фильтрация







# Медианный фильтр

Медианный фильтр — один из видов цифровых фильтров, широко используемый в цифровой обработке сигналов и изображений для уменьшения уровня шума. Значения отсчётов внутри окна фильтра сортируются в порядке возрастания (убывания); и значение, находящееся в середине упорядоченного списка, поступает на выход фильтра.(wikipedia)

До фильтрации	2	3	80	6	2	3			
Окно 3									
0	2	3	=>	2					
	2	3	80	=>	3				
		3	80	6					
		3	6	80	=>	6			
			80	6	2				
			2	6	80	=>	6		
				6	2	3			
				2	3	6	=>	3	
					2	3	0		
					0	2	3	=>	2
После фильтрации	2	3	6	6	3	2			

-  [mikheev.hopto.org/stm32](http://mikheev.hopto.org/stm32) Материалы курса «Программирование микроконтроллеров» , Москва, 2023-2024.
-  Datasheets STM32F429, L3GD20
-  Reference manual STM32F429
-  <https://microtechnics.ru/stm32f3-spi-i-giroskop-l3gd20/> STM32 и гироскоп L3GD20. Часть 1. Настройка и обмен данными.
-  <https://microtechnics.ru/stm32-i-giroskop-l3gd20-chast-2/> STM32 и гироскоп L3GD20. Часть 2. Определение положения платы.

-  Interfacing L3GD20 MEMS Gyroscope using SPI  
<https://www.youtube.com/watch?v=iAzqwXTmGcM>
-  STM32 SPI Interrupt Tutorial: Setup And Usage With Registers <https://www.youtube.com/watch?v=RBXQLPGr7Q>
-  <https://github.com/adem-alnajjar/Gyroscope-L3GD20-STM32/>
-  MedianFilter  
<https://github.com/accabog/MedianFilter/tree/master>



Median Filter <https://en.wikipedia.org/wiki/Medianfilter>



ArduinoRealTimePlot

<https://github.com/WaveShapePlay/ArduinoRealTimePlot/>

Спасибо за внимание!