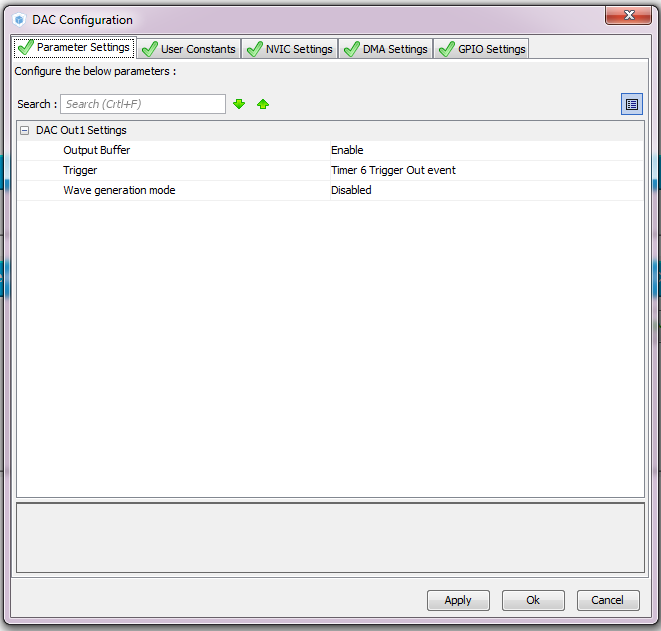
**Lesson 29**

**HAL. DAC. Triangle. DMA**

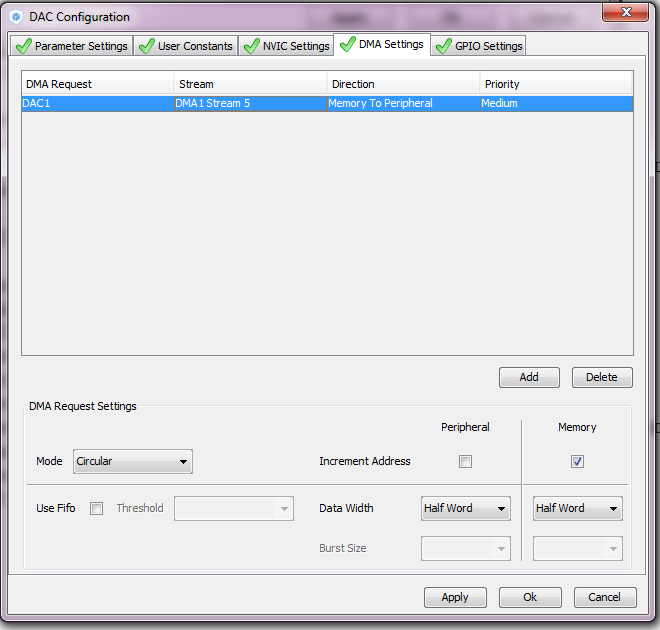
Today we will continue to work with **triangular impulses** , which we have already formed in the [**previous lesson**](http://narodstream.ru/stm-urok-28-hal-dac-triangle/) . We will not only create these pulses with the help of the hardware mechanism built into the DAC, but we will create them using a previously prepared array, and these values will be sent to our converter with the help of **DMA** technology .

Thus we can achieve more flexible results. That is, for example, we can reduce the resolution without changing the amplitude, and leave it at its maximum (4096).

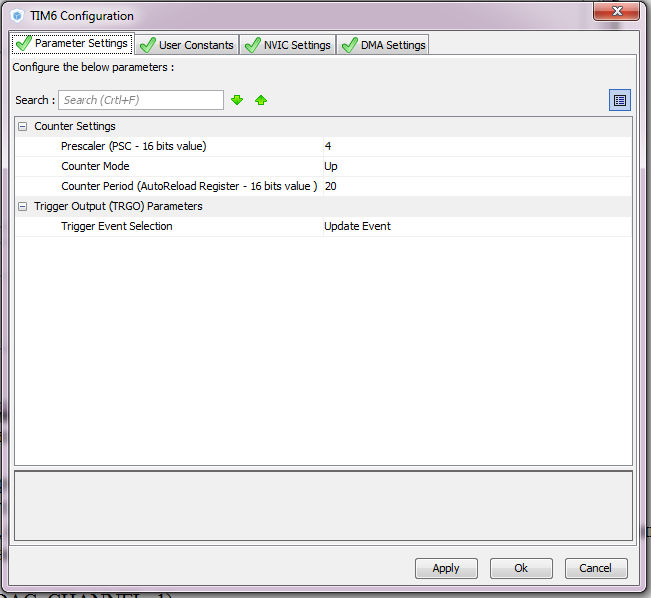
We create the project from the project DAC\_TRIANGLE. Let's call it DAC\_TRIANGLE2. Run the project in the Cube and go directly to Configuration, since everything else we have already configured.Disable hardware triangles in DAC



Let's go to the DMA settings and configure it as follows



In the timer, we make the following changes to the prescaler and period



Generate and run the project. We'll collect the code, set up the programmer and start writing.

Add an array of values ​​to main.c

/ \* USER CODE BEGIN PV \* /

/ \* Private variables ------------------- \* /

**uint16\_t buf\_tr [64] = {**

**127,255,383,511,639,767,895,1023,**

**1151,1279,1407,1535,1663,1791,1919,2047,**

**2175,2303,2431,2559,2687,2815,2943,3071,**

**3199.3327.3455,3583,3711,3839,3967,4095,**

**3967,3839,3711,3583,3455,3327,3199,3071,**

**2943,2815,2687,2559,2431,2303,2175,2047,**

**1919,1791,1663,1535,1407,1279,1151,1023,**

**895,767,639,511,383,255,127.0};**

/ \* USER CODE END PV \* /

Remove the extra variable from the files stm32f4xx\_it.c and main.h

/ \* USER CODE BEGIN 0 \* /

~~extern uint8\_t tim6\_counter;~~

/ \* USER CODE END 0 \* /

#include "stm32f4xx.h"

~~uint8\_t tim6\_counter;~~

Let's replace the function call of the DAC

  / \* USER CODE BEGIN 2 \* /

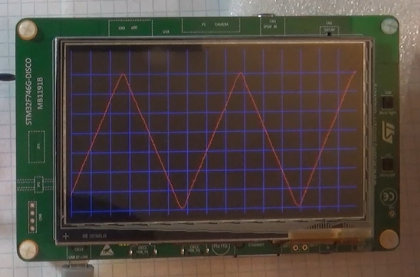
        HAL\_TIM\_Base\_Start (& htim6);

~~HAL\_DAC\_Start (& hdac, DAC\_CHANNEL\_1);~~

**HAL\_DAC\_Start\_DMA (& hdac, DAC\_CHANNEL\_1, (uint32\_t \*) buf\_tr, 64, DAC\_ALIGN\_12B\_R);**

  / \* USER CODE END 2 \* /

We collect, sew, look at our home-made oscilloscope, organized now on the basis of the F746-Discovery board

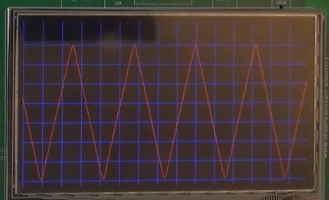


At the moment, we observe oscillations with a frequency of approximately 40-50 kHz.

To make sure that there will be smooth triangles also on another frequency, let's try to change it. For this we do not need to go to the Cube, since we will change this frequency for a while, so we will change it in the timer initialization function  **MX\_TIM6\_Init**

  htim6.Init.Period = **10** ;

The idea is to double the frequency value to see this, collect the project and patch the controller. Watch the result



In the [**next lesson,**](http://narodstream.ru/stm-urok-30-hal-dac-sinus-dma/) we will try to generate something like a sinusoidal oscillation with the help of a DAC. And we will try to measure the frequency of oscillations somehow.