AN4566
Application note

How to extend the DAC performance on STM32 MCUs

Introduction

Most of the STM32 microcontrollers embed 12-bit DACs (digital to analog converters), specified to operate at up to 1 Msps (megasamples per second).

Several applications benefit from DACs operating at higher speeds. This document explains how to extend the speed performance of microcontrollers listed in *Table 1* using external operational amplifiers (OpAmps).

The STM32 DAC system is described in *Section 1* of this document, while an application example focusing on 5 Msps sine wave generation is presented in *Section 2*.

Table 1. Applicable products

Type	Product series
Microcontrollers	STM32F0 series
	STM32F1 series
	STM32F2 series
	STM32F3 series
	STM32F4 series
	STM32F7 series
	STM32G0 series
	STM32G4 series
	STM32H5 series
	STM32H7 series
	STM32L0 series
	STM32L1 series
	STM32L4 series
	STM32L4+ series
	STM32L5 series
	STM32U0 series
	STM32U3 series
	STM32U5 series

Contents

List of tables

List of figures

1 The STM32 DAC system

1.1 DAC equivalent circuit

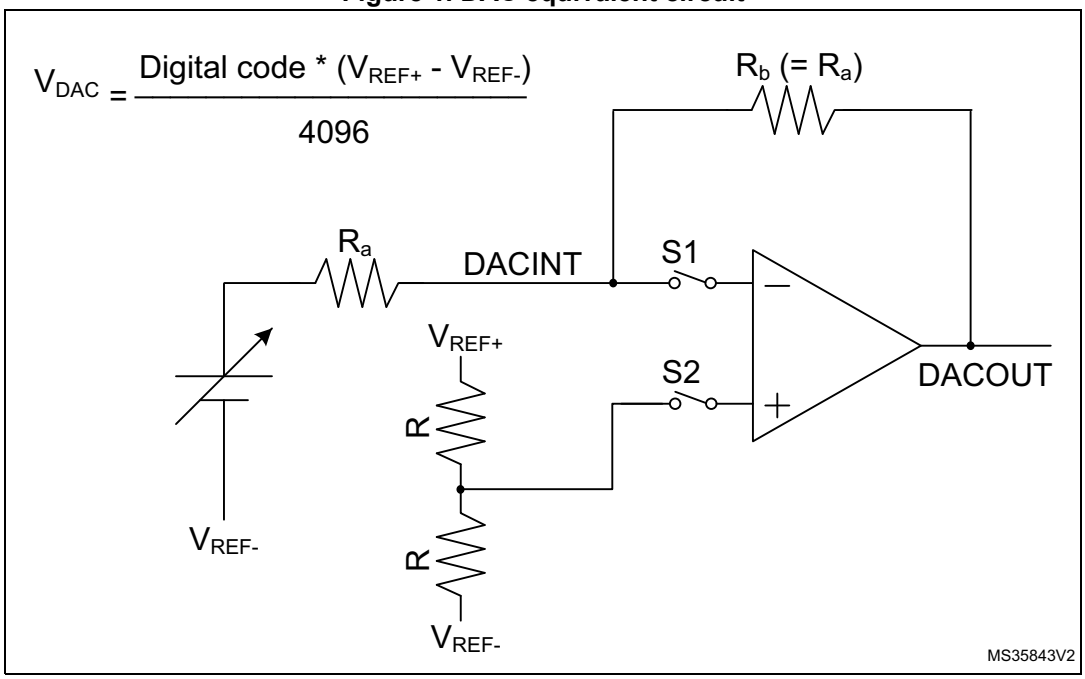
STM32 microcontrollers, based on Arm®(a) cores, use DACs to transform digital data into analog signals. The DAC can be modeled as a digitally controlled voltage source and an output impedance, as shown in *Figure 1*.

The output impedance of the DAC is constant, independent from the digital input signal.

When the output buffer is OFF, DACINT and DACOUT are connected through the resistor R_b, hence the output impedance of the DAC is $R_a + R_b$ ($R_b = R_a$), and $R_{DAC} = 2 * R_a$ (S1 and S2 switches are open).

When the buffer is enabled, the OpAmp is configured as an inverting amplifier with $A_v = -1$, and the output impedance is almost zero thanks to the feedback loop.

Figure 1. DAC equivalent circuit



1.2 DAC speed on the specification

When the output buffer is enabled on the DAC output, the speed is specified by the output buffer performance. This number is indicated in the $T_{settling}$ or Update rate in the product datasheet.

a. Arm is a registered trademark of Arm Limited (or its subsidiaries) in the US and/or elsewhere.

When the output buffer is disabled, the output signal speed simply follows the RC constant, which is determined by the DAC output impedance R_{DAC} (= 2 * R_a), and the capacitive load on the DACOUT pad.

As an example, the STM32F407 defines the impedance output with buffer off at a maximum value of 15 kΩ. If a 10 pF capacitive load (including the parasitic capacitance of the device on DACOUT pad) is considered, to get ±1 LSB of the final value (from lowest to highest code) we have

$$1 - \frac{1}{2^N} = 1 - e^{T/(CR)}$$

Solving for T gives T = CR * N* ln 2 = 0.693 CR * N = 1.8 µs. In this configuration the conversion time cannot be smaller than 1.8 µs (equivalent to a frequency of 555 kHz).

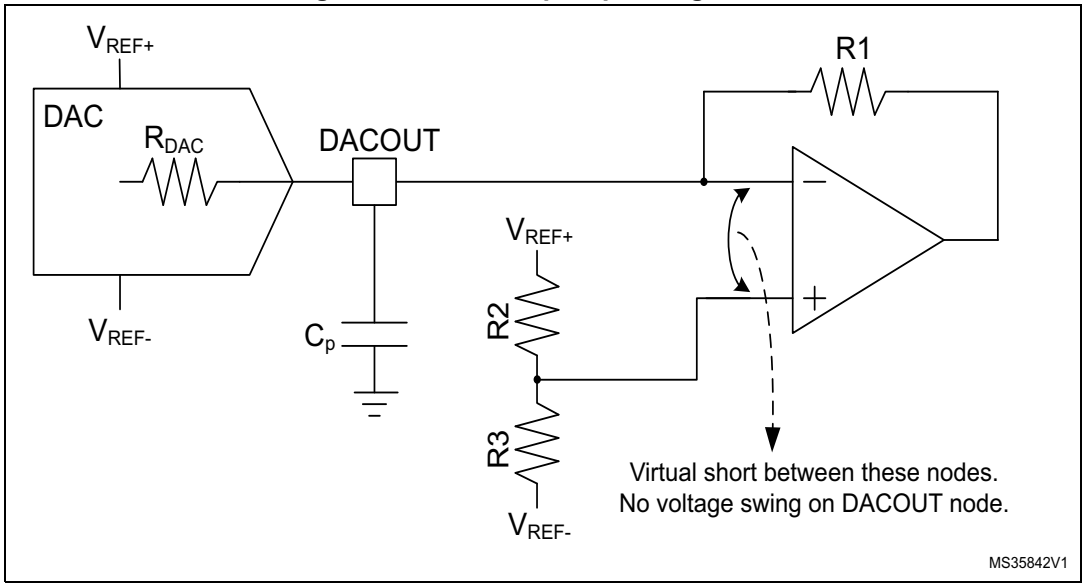
This analysis does not include any effect of the switching speed of the DAC itself and its transient. When using high speed, these factors cannot be ignored, they degrade the performance.

1.3 External OpAmp implementation

As described in *Section 1.2*, the output DAC conversion time is specified by the embedded output buffer when buffer is enabled. When the buffer is disabled, the output impedance and the DACOUT capacitance (C_p) determine the conversion time.

There is a configuration for which it is possible to ignore the DACOUT capacitance. By using the external OpAmp in inverting mode, the DACOUT node voltage is fixed, as shown in *Figure 2*.

Figure 2. External OpAmp configuration



In this configuration there is a minor limitation due to the RC constant, the main limitation are the external OpAmp speed (gain bandwidth and slew rate) and the DAC digital data update rate. There are, however, some disadvantages. The feedback resistor R1 must be equal to the R_{DAC} on chip of the STM32, otherwise it creates a DAC gain error.

Integrated resistors usually feature a rather wide spread on their absolute value, and significant variations over temperature, hence it is necessary to calibrate the gain error (discussed in detail in *Section 2.2.4: Output gain calibration*).

It is also possible to use the external OpAmp in voltage follower mode. This enhances the output bandwidth and slew rate, however the R_{DAC} output impedance and the parasitic capacitor on the DACOUT form an RC filter that limits the speed performance.

For the voltage follower mode, it is not necessary to perform the gain calibration.

1.4 Digital data update rate

The STM32 DAC output data need to be written to the DAC holding register (DHR), then the data is moved to the DAC output register (DOR) for the conversion.

Generally, the data are saved in a RAM, and the CPU is in charge of the transferring the data from RAM to DAC.

When using the DMA, the overall performance of the system is increased by freeing up the core: data go from memory to DAC by DMA, without need for any actions by the CPU. This keeps CPU resources free for other operations.

The trigger of the DAC conversion can be done by the software, external triggers, or by the timers. For the high speed conversion cases, it is recommended to use the timer trigger in combination with the data transfer done by the DMA.

The transfer speed from memory to the DAC is limited by several factors, among them:

- the clock cycle of the APB or of the AHB (DAC clock)
- the DMA transfer cycle from memory to the DAC (includes the AHB to APB bridge)
- the trigger mechanism itself.

The DAC on STM32F407x microcontrollers is running on the APB1:

- three cycles after the trigger, DHR data is moved to the DOR register
- at the same time a DMA request is generated from DAC to DMA
- DMA transfer takes at least one APB clock cycle.

So a total of four APB clock cycles is needed to update the DOR data. As APB1 maximum clock is 42 MHz (for ST32F407x), 10.5 Msps is the maximum update rate for the DAC output register when timer trigger and the DMA are used for the data update.

The minimum transfer clock cycle by DMA to the DAC is not the same for all STM32 microcontrollers, because of the different bus configuration.

Table 2 shows the maximum sampling rate for different STM32 products.

Table 2. Maximum sampling time

Product	Maximum bus speed	DAC maximum sampling rate
STM32F0 series	48 MHz	4.8 Msps
STM32F100xx	24 MHz	2.4 Msps
STM32F101xx STM32F103xx STM32F105xx STM32F107xx	36 MHz	4.5 Msps
STM32F2 series	30 MHz	7.5 Msps
STM32F3 series	36 MHz	4.5 Msps
STM32F40x STM32F41x	42 MHz	10.5 Msps
STM32F42x	45 MHz	11.25 Msps
STM32F7 series	54 MHz	13.5 Msps
STM32G0 series	64 MHz	8.0 Msps
STM32G4 series	170 MHz	28.8 Msps 30.9 Msps (DMA double data mode)
STM32H72x STM32H73x	137.5 MHz	51.1 Msps
STM32H5 series	250 MHz	41.6 Msps
STM32H74x STM32H75x	120 MHz	40.8 Msps
STM32H7Ax STM32H7Bx	140 MHz	49.2 Msps
STM32L0 series	32 MHz	4.0 Msps
STM32L1 series	32 MHz	3.2 Msps
STM32L4 series	80 MHz	10 Msps
STM32L4+ series	120 MHz	12 Msps
STM32L5 series	110 MHz	11 Msps
STM32U0 series	56 MHz	7 Msps
STM32U3 series	96 MHz	16 Msps
STM32U5 series	160 MHz	16 Msps

Note: The values reported in Table 2 have been measured on the bench, when the bus is not used by any other system. In real applications some margin is needed.

1.4.1 DMA double data mode

Some DACs on STM32 MCUs support DMA double data mode. When the DMA controller is used in Normal mode, only 8- or 12-bit data are transfered by a DMA request. As the

STM32 MCUs can be accessed with 32-bit data bus, DMA double data mode transfers two 12-bit sets data at once, thus requiring lower bus occupation compared to the normal mode.

1.5 Summary

By using an external high speed OpAmp, it is possible to extend the speed performance of the DACs by more than 1 Msps. See *Section 2* for an example showing how to use this technique on STM32 products.

2 Example

The example of the DAC used at high speed is based on STM32F407. It shows how to generate a 200 kHz sine wave by the DAC operating at 5 Msps.

2.1 External OpAmp choice

As indicated before, the external OpAmp defines the DAC total performance.

To choose the OpAmp, the following parameters must be considered.

- slew rate
- gain bandwidth (GBW)
- open loop gain
- supply voltage range
- output voltage swing performance
- input common mode voltage range
- minimum stable gain.

For the lowest to highest code transient on 5 Msps case with V_{REF} voltage 3.3 V, the OpAmp must have a slew rate higher than $3.3 * 5 * 10^6 = 16.5$ V/µs.

If STM32 DAC operates at 3.3 V, it is possible to use the OpAmp 3.3 V supply. It is also possible to consider another analog supply rail, this is the option used in the example.

It is recommended to have at least two times of sampling speed of the gain bandwidth, so, for 5 Msps, GBW must be wider than 10 MHz.

To keep good DAC linearity, the open loop gain must be higher than 60 dB.

If the output voltage must be near the supply voltage, the output voltage swing of the OpAmp must preferably be rail to rail. Otherwise, if the voltage swing is near to the supply or ground rail, the signal is saturated and this results in distortion.

Even the OpAmp negative input is fixed at the reference voltage level, it is necessary to verify that the input common voltage range covers the reference voltage level with a margin.

The used OpAmp gain is about -1, so the OpAmp must be stable at this gain.

By considering the above criteria, LMH6645/6646/6647 from Texas Instruments fit the requirements:

- slew rate: 22 V/µs
- gain band width: 55 MHz
- open loop gain: 87 dB
- supply voltage range: 2.5 to 12 V
- input common mode voltage 0.3 V beyond rails
- output voltage swing 20 mV from rails
- stable from gain +1.

2.2 Software implementation

For this example the STM32F407 is powered with a 3.3 V supply.

2.2.1 Digital sine waveform pattern preparation

As described in AN3126 "*Audio and waveform generation using the DAC in STM32 products*", available on www.st.com, a sine wave pattern must be prepared according to the following formula

$$Y_{SineDigital}(x) = \left(\sin\left(2\Pi \cdot \frac{x}{n_S}\right) + 1\right) \cdot \frac{0xFFF + 1}{2}$$

Digital inputs are converted to output voltages by linear conversion between 0 and V_{REF+}.

The analog output voltage on each DAC channel pin is determined as:

$$DAC_{Output} = V_{REF} \cdot \frac{DOR}{DAC\_MaxDigitalValue}$$

So the analog sine waveform can be determined by the following equation

$$Y_{SineAnalog}(x) = 3.3Volt \cdot Y_{SineDigital}(x) / 0xFFF$$

The table can be saved in the memory and transferred by DMA. The transfer is triggered by the same timer that triggers the DAC.

2.2.2 Setting the sine waveform frequency

To set the frequency of the sine wave signal, it is necessary to set the frequency of the timer trigger output. The frequency of the produced sine wave is

$$f_{Sinewave} = f_{TimerTRGO} / n_S$$

If TIMx_TRGO is 5 MHz (n_S = 25), then the frequency of the DAC sine wave is 200 kHz.

To have the exact frequency on the output, the system clock must be adjusted, so that the timer can generate exactly 5 MHz.

In STM32F407, some timers can run with a clock frequency twice the one of the APB1 clock, so the resolution is two times better than APB1 clock. However, the DAC captures the trigger signal by APB1 clock, so the DAC timing cannot be better than APB1 clock.

For example, if the timer is programmed with 25 clock cycles (corresponding to 12.5 cycles of the APB1 clock), the DAC trigger occurs 12 times, then 13 times, alternately. So one APB1 clock results in jitter on every sampling period.

Here is the example of the clock setting:

- System clock source = PLL (HSE)
- SYSCLK (Hz) = 160000000
- HCLK (Hz) = 160000000
- AHB prescaler = 1
- APB1 prescaler = 4
- APB2 prescaler = 2
- HSE frequency (Hz) = 8000000
- PLL_M = 8
- PLL_N = 320
- PLL_P = 2
- PLL_Q = 7

TIM6 has been used for the trigger.

With this configuration, 80 MHz is the timer clock, so, to get 5 MHz trigger, the prescaler has been set to 1 (PSC = 0) and the counter to 16 (CNT = 15).

2.2.3 Offset calibration

The use of an external OpAmp introduces additional offsets, among them the one of the OpAmp itself, and the one coming from the external V_{REF} resistor ladder.

To do the calibration, it is necessary to connect the output of the OpAmp to one of the available ADC channels of the STM32 microcontroller.

The procedure to calibrate the offset is the following one (see *Table 3*):

1. Set up the DAC DOR as 2047.
2. Measure the OpAmp output by the ADC.
3. Set up the DAC DOR of the ADC result of last measurement (in this case 2065).
4. Verify the result with the ADC (in this case, 2048, still 1 LSB offset).

Table 3. Example of the offset calibration measurement

DAC DOR	ADC result value
2047	2065
2065	2048

2.2.4 Output gain calibration

As indicated before, the output gain is defined by the ratio of the DAC output impedance and the feedback resistance of the external OpAmp.

The output gain calibration must be performed during the initialization of chip, and every time the temperature changes significantly (say more than 10 °C). Temperature changes can be detected by the on-chip temperature sensor.

To do the calibration, connect the output of the OpAmp to one of the available STM32 ADC channels.

To calibrate the gain (see *Table 4*) go through the following steps:

1. Set up DAC DOR as 1023
2. Measure the OpAmp output by the ADC
3. Set up DAC DOR as 3071
4. Measure the OpAmp output by the ADC.

Table 4. Example of the calibration measurement

DAC DOR	ADC result value
1023	3135
3071	983

So the amplifier has a gain of 1.0508, obtained as (3135 - 983) / 2048.

This result can be used in the equation shown in *Section 2.2.1*. It is recommended to have some margin (say 100 mV) for each of the supply rail and ground rails. The digital code swing must be less than 200 mV from the supply, and also use the gain calibration factor

$$Y_{SineDigital}(x) = \frac{3.1}{3.3 \cdot 1.0508} \cdot \left(\sin\left(2\Pi \cdot \frac{x}{n_S}\right) + 1\right) \cdot \frac{0xFFF + 1}{2} + 18$$

By using the above equation, *Table 5* can be generated.

Table 5. Example of digital sample values

Sample	Digital sample value $Y_{SineDigital}$ (x)
0	2066
1	2521
2	2948
3	3319
4	3612
5	3807
6	3893
7	3864
8	3723
9	3477
10	3142
11	2740
12	2295
13	1837
14	1392
15	990
16	655

Table 5. Example of digital sample values (continued)

Sample	Digital sample value $Y_{SineDigital}$ (x)
17	409
18	268
19	239
20	325
21	520
22	813
23	1184
24	1611

Note: The output signal is inverted if compared to the digital code, because of the inverting amplifier stage of the external OpAmp.

2.3 Hardware implementation

As described in *Section 2.1*, an external component has been chosen.

The actual circuit is shown in *Figure 3*, the component values are listed in *Table 6*.

R1, chosen as typical output DAC impedance, is 12.5 kΩ (for other devices, consult the electrical specification in the datasheet). C1 is added to avoid overshoot on the output signal.

Figure 3. Circuit implementation

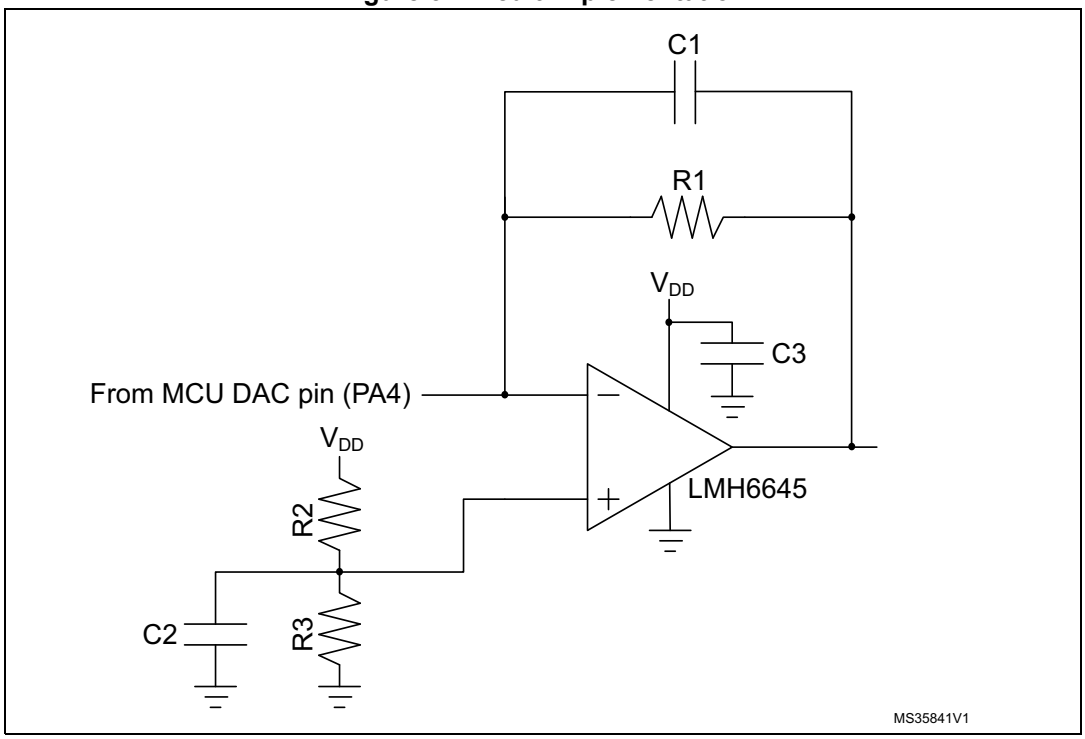


Table 6. Component values

Type	Component	Value
Resistor	R1	12 kΩ
	R2	10 kΩ
	R3	10 kΩ
Capacitor	C1	5 pF
	C2	100 nF
	C3	100 nF

3 Measurements

The measurements have been done on a STM32F4DISCOVERY board with the configuration shown in *Figure 3*.

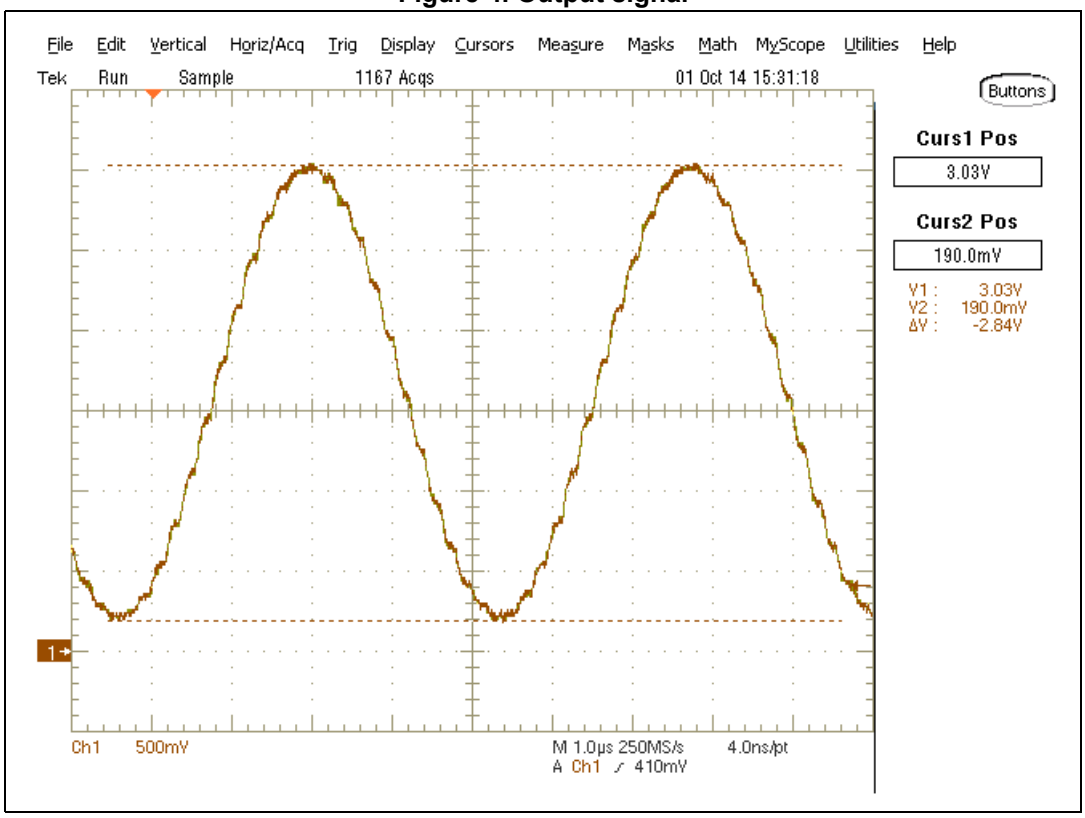
3.1 Board modification

STM32F407 DAC1 output is assigned to PA4, which, in turn, is connected to the on-board audio codec through a 100 kΩ resistor to GND. To remove this effect, the R48 (0 Ω) resistor has been removed from the board.

3.2 Measurement results

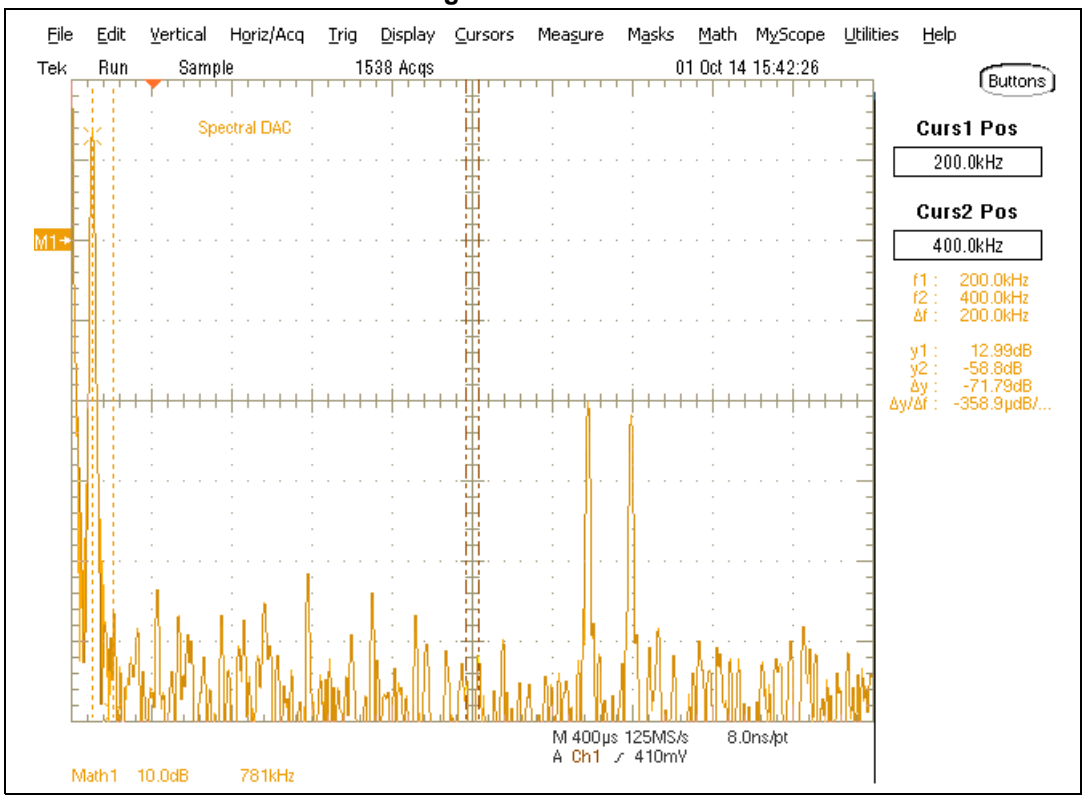
The output signal is shown in *Figure 4*, while *Figure 5* is the corresponding FFT analysis.

Figure 4. Output signal



Output swing is not equal to 3.1 V_{pp}, as sampling time is not aligned with the peak of the sine wave signal.

Figure 5. FFT result



The second and third harmonics are around the noise level.

4 Conclusion

The DAC used by STM32F4 microcontrollers has been characterized up to 1 Msps. By using a high speed external OpAmp, it can operate up to 5 Msps.

Additional remarks:

- by using high speed sampling rate, it is possible to reduce the anti-aliasing filter order
- by using the on chip ADC, it is possible to calibrate the output swing and the offset.

5 Revision history

Table 7. Document revision history

Date	Revision	Changes
03-Nov-2014	1	Initial release.
02-Aug-2015	2	Added STM32L4 series in *Table 1: Applicable products* and in *Table 2: Maximum sampling time*.
19-Sep-2019	3	Document scope extended to STM32L4+, STM32L5, STM32H7, STM32G0 and STM32G4 series, hence updated *Table 1: Applicable products* and *Table 2: Maximum sampling time*. Updated *Section 1.1: DAC equivalent circuit*, *Section 1.3: External OpAmp implementation* and *Section 2.2.4: Output gain calibration*. Added *Section 1.4.1: DMA double data mode*. Updated *Figure 1: DAC equivalent circuit*. Minor text edits across the whole document.
03-Oct-2022	4	Document scope extended to STM32U5 series. Updated *Table 1: Applicable products* and *Table 2: Maximum sampling time*. Minor text edits across the whole document.
09-Jan-2025	5	Document scope extended to STM32H5, STM32U0, and STM32U3 series, hence updated *Table 1: Applicable products* and *Table 2: Maximum sampling time*. Minor text edits across the whole document.