Tampere University

Ida Wik

FREQUENCY AND PULSE GENERATION FEATURES IN A MULTIFUNCTIONAL FIELD CALIBRATOR

Master's Thesis
Faculty of Information Technology and Communication Sciences
Examiners: Professor Karri Palovuori and University Lecturer Erja Sipilä
September 2022

ABSTRACT

Ida Wik: frequency and pulse generation features in a multifunctional field calibrator
Master's Thesis
Tampere University
Master's programme in electrical engineering
September 2022

The aim of the Thesis was to investigate improvements that could be made for frequency and pulse generation features of a next-generation multifunctional field calibrator as well as to suggests how the found improvements could be implemented. The improvement investigation was done by reviewing the frequency and pulse generation specifications of multifunctional calibrators that were on the market during the writing process of the Thesis. In addition to that, a customer needs analysis was performed by interviewing experts, and by analyzing customers' feedback. Based on the results of the investigation, it can be concluded that the frequency and amplitude range and resolution of the current solution by Beamex is competitive and do not require alternation. However, the selection of generatable waveforms could be improved by adding a sine wave generation possibility into the frequency generation function. The current solution is only capable of generating symmetric and positive square waves. Furthermore, some requests for dual pulse generation were found during the investigation.

The main focus in the solution design process was the sine wave generation because the dual pulse generation can be utilized easily if the next-generation multifunctional field calibrator has a modular structure. In that case, the number of frequency and pulse generation channels in the calibrator can be increased by adding multiple frequency and pulse generation modules into the calibrator. On the other hand, adding a sine wave generation option to the system is more complicated. Two possible solution suggestions for sine wave generation were designed and evaluated in the present thesis. One solution is based on direct digital synthesis and another one on usage of timer, registers, and direct memory access feature of a microcontroller. In theory, both of the solution suggestions should be able to generate square, pulse, and sine waves. However, by evaluating the solution suggestions, it can be said that the option to generate sine waves increases the complexity and cost of the system. In addition to that, the demand for sine wave generation might not be that high. Hence, it should be re-evaluated if it is profitable to add a sine wave option to the frequency generation.

Keywords: signal generation, direct digital synthesis, microcontroller, direct memory access, frequency generation, pulse generation, multifunctional field calibration.

TIIVISTELMÄ

Ida Wik: Taajuus- ja pulssigenerointiominaisuudet kenttäkalibraattorissa
Diplomityö
Tampereen yliopisto
Sähkötekniikan DI-ohjelma
Syyskuu 2022

Tämän diplomityön tarkoitus oli etsiä mahdollisia parannusehdotuksia kenttäkalibraattorin taajuus- ja pulssigenerointiin. Lisäksi työssä tutkittiin, kuinka löydetyt parannusehdotukset pystyttäisiin toteuttamaan Beamexin seuraavan sukupolven kenttäkalibraattorissa. Parannusehdotusten etsinnässä käytettiin kolmea metodia: diplomityön kirjoitusprosessin aikana markkinoilla olevien kalibraattoreiden vertailua, alan ammattilaisten haastattelua, sekä asiakaspalautteen analysointia. Näiden metodien avulla saatiin selville, että Beamexin nykyisen kenttäkalibraattorin taajuus- ja pulssigenerointiominaisuuden ulostulon taajuus- ja amplitudialue, sekä niiden resoluutiot ovat sopivia ja kilpailukykyisiä. Sen sijaan parannuskohdat löytyivät generoitavista aaltomuodoista, nykyinen kenttäkalibraattori kykenee generoimaan vain symmetristä ja positiivista kanttiaaltoa, mutta tutkimuksen perustella asiakkailla voisi olla kysyntää myös siniaallon generoinnille, sekä mahdollisuudelle generoida kahta eri pulssijaksoa samanaikaisesti kahdella eri kanavalla.

Työssä keskityttiin suurimmaksi osaksi tutkimaan, kuinka siniaallon generointi voitaisiin lisätä taajuusgenerointiominaisuuteen. Sillä käyttämällä modulaarista rakennetta seuraavan sukupolven kenttäkalibraattorissa, useampikanavainen signaalin generointi olisi yksinkertaisesti toteutettavissa lisäämällä useampia signaalingenerointimoduuleja kalibraattoriin. Sen sijaan siniaallon generointiominaisuuden lisääminen kalibraattoriin on paljon monimutkaisempaa toteuttaa. Tässä työssä esitellään ja analysoidaan kaksi ratkaisua, joilla olisi kanttiaallon lisäksi mahdollista generoida siniaaltoja. Nämä ratkaisut ovat digitaaliseen taajuussynteesin perustuva ratkaisu, sekä mikrokontrollerin kelloa, rekistereitä, ja oikosiirtoa hyödyntävä ratkaisu. Työssä todetaan, että teoriassa molemmat ratkaisut ovat toimivia sekä siniaallon että kanttiaallon generointiin. Kuitenkin ratkaisuja analysoidessa on selvästi nähtävissä, että siniaallon generointimahdollisuuden lisääminen taajuusgenerointiin monimutkaistaisi systeemiä, sekä kasvattaisi sen kustannuksia. Lisäksi asiakkaiden kysyntä siniaallon generoinnille ei todennäköisesti ole niin suuri, että yllä mainittuja siniaallon genreointiin liittyviä ratkaisuja olisi kannattavaa tuottaa. Tätä tulisi kuitenkin vielä uudelleen arvioida tulevaisuudessa.

Avainsanat: signaalin generointi, digitaalinen taajuussynteesi, mikrokontrolleri, oikosiirto, taajuusgenerointi, pulssigenerointi, kenttäkalibraattori.

PREFACE

This Master's Thesis has been done as part of the research and development of the next-generation multifunctional calibrator by Beamex.

First of all, I want to thank Beamex for giving me this opportunity. Also, I want to thank all my colleagues at Beamex for the support and help I received during the writing process of this Thesis. Especially, I want to thank everyone who participated in the interviews, as well as Mats and Markku who helped me to understand some more difficult technical questions which I faced during the designing process. Furthermore, I want to thank Petteri, supervisor of this Thesis from Beamex, for the excellent guidance. Additionally, I thank Karri and Erja, supervisors and examiners of the Thesis, for all the corrections and comments.

Lastly, I want to thank my spouse and my bunny for motivating and supporting me throughout this writing process.

Tampere, 16 September 2022

Ida Wik

CONTENTS

LIST OF FIGURES

LIST OF SYMBOLS AND ABBREVIATIONS

ADC	Analog-to-digital converter
ASRC	Arbitrary sample rate conversion
ATEX	Appareils destinés à être utilisés en atmosphères explosives
DAC	Digital-to-analog converter
DDS	Direct digital synthesis
DFT	Discrete Fourier transformation
DMA	Direct memory access
FFT	Fast Fourier transformation
FET	Field effect transistors
FIR	Finite-impulse response
FPGA	Field-programmable gate array
IC	Integrated circuit
IEEE	Institution of electrical and electronics engineers
IUC	Instrument under calibration
LabVIEW	Laboratory virtual instrument engineering
LSB	Least significant bit
LC	Inductor-capacitor
MCU	Microcontroller
LUT	Look-up table
SMD	Surface-mount device
Op-amp	Operational amplifier
PC	Personal computer
RF	Radio frequency
RC	Resistor-capacitor
ROM	Read only memory
RPM	Revolutions per minute
THD	Total harmonic distortion
VCO	Voltage-controlled oscillator

A	Gain of the amplifier
B	Gain of the feedback circuit
f_{clk}	Clock frequency
f_{in}	Input frequency
f_s	Sampling frequency
f_{out}	Output frequency
i	Ordinal number of the sampling point in a look-up table
k	Number of bits used in ROM
K_m	Frequency bin with the largest magnitude
M	Phase increment word
m	Size of the timer's reload register
n	Number of bits used in phase accumulator
n_c	Number of periods in the sequence
$N_{half\text{-}period}$	Number of counted half-periods
N_{period}	Number of counted periods
n_{tot}	Number of samples of sine wave in on period
n_{sample}	The ordinary number of sine wave sample
r	Desired output frequency resolution
T_M	Mean period
$t_{measurement}$	Measurement time

t_{pulse}	Duration of one clock pulse
t_{sample}	Time between samples
V_i	Input voltage
V_{pp}	Peak-to-peak voltage
V_{th}	Threshold voltage
V_o	Output voltage

1. INTRODUCTION

Accurate and traceable measurements are key components in many of the manufacturers' processes, they verify that the processes are done correctly and within standards which assures safety, as well as the high and uniform quality of the processes and their end products. Regular calibrations of the measurement instruments are required to ensure the accuracy of the measurements. In the field, manufacturers can calibrate their measuring instruments with a multifunctional field calibrator, which can be used to perform calibrations, adjustments, and testing of various kinds of measurement instruments. One of the functions of the multifunctional calibrator is frequency and pulse generation capability. Moreover, frequency generation can be seen as the generation of periodical signals, such as square and sine waves, which amplitude and frequency can be adjusted. Likewise, a pulse generation can be seen as the generation of pulse train which amplitude, duty-cycle, and quantity in the sequence can be adjusted.

The present Thesis is done as a part of the development of a next generation multifunctional field calibrator by Beamex, which is a company that specializes in developing and manufacturing calibrators and calibration software. Moreover, the aim of the Thesis is to investigate how the frequency and pulse generation features could be improved in this next generation calibrator. In order to find the improvements that could be added to the next generation calibrator, the frequency and pulse generation capabilities of multifunctional field calibrators presently on the market are examined. In addition to that, a customer needs analysis about these features is performed, to directly find out in which applications customers of Beamex are using the signal generation features of the calibrator and if there are some gaps in these functions. Furthermore, the internal needs of the company are noted, in the search for improvements. These requirements are for example to have a simple and profitable solution for frequency and pulse generation. In this analysis, also the frequency measuring and pulse counting features of the multifunctional calibrator are taken into account, as they are strongly connected to the pulse and frequency generation. After identifying the improvement possibilities, two possible solution designs are suggested.

The second and third chapter of the Thesis are used to cover the background theory of the subject of the Thesis. More specifically, the second chapter discusses the basics of

calibration as it is the framework of this Thesis. In the third chapter, the theories of both solution suggestions are covered, and related studies are presented and discussed. The fourth chapter includes the plan for improvement investigation and its results. Furthermore, at the end of chapter four, the specification for frequency and pulse generation is presented. After the theory of the solution suggestions are presented and the desired improvements are listed, the designs for solution suggestions are presented, their pros and cons are weighted, also the solutions are compared to each other in the fifth chapter. Additionally, the open-ended questions about the solutions which require further testing are listed in the fifth chapter. Finally, the Thesis is summarized in the final chapter.

2. BASICS OF CALIBRATION

The present chapter is the first part of the theoretical background included in this Thesis. The framework of the Thesis is in the field of calibration. Therefore, it is important that the reader is familiarized with the concept of calibration, which is often misused in the spoken language. After reading through this chapter, the reader understands the basic terminology related to calibration, how to perform a simple calibration, what to do when calibration fails, and what are the common error sources in calibration process.

2.1 Performing a Simple Calibration

Calibration is a process in which the accuracy of a measurement instrument is checked by comparing the instrument's outputs induced by known inputs to a traceable reference standard or device. Accuracy describes how close the measured value is to the reference value. A device that provides the reference standard can be called a calibrator. [1], [2] Usually in the calibration process, the output of the instrument under calibration (IUC) is checked at multiple points to cover the whole calibration range. The calibration range defines, what is the output of the instrument for a certain input. It should not be confused with the instrument range which determines the maximum and minimum values that can be measured by the device.[2]

Let's use an example of calibration of a pressure transmitter to explain the terminology of calibration in more detail. The calibration of a pressure transmitter is a very classical example, which is often used for this purpose, e.g., Cable M. [2] and Kuphaldt T. [3] have used it in their books. In this example, let the instrument range of a pressure transmitter be from 0 bar to 20 bar, and the output range is from 4 mA (milliamps) to 20 mA. Hence, if the input is 0 bar the output is 4 mA, and if the input is 20 bar the output is 20 mA. However, an engineer who installed the pressure transmitter has set the calibration range to be from 0 bar to 10 bar. Therefore, if the input is 0 bar the output is still 4 mA but when the input is 10 bar the output will be 20 mA. With these settings, only the pressure from 0 bar to 10 bar can be measured accurately, although the pressure transmitter could measure pressure up to 20 bar. Next, let's calibrate this pressure transmitter. To cover the whole calibration range three different inputs are used 0, 5, and 10 bar. The reference standard states that the outputs of the IUC for the above inputs should be 4 mA, 12 mA, and 20 mA. After the outputs of the IUC are collected, they can be compared to the corresponding reference values.

2.2 Analyzing Results of Calibration

After learning to perform a simple calibrate, it is time to learn how to analyze the result of the calibration. The collected output values of the IUC are accepted if they are within the tolerance limits of the reference value. Moreover, the tolerance limits used in the calibration, define how much an acceptable output value of the IUC can deviate from the corresponding reference value. Furthermore, the used tolerance level depends on factors such as criticality of the monitored process, capability of the test equipment, unity with the similar instruments, and manufacturer's tolerance specification. For instance, in this example the reference value at 0 bar is 4.0 mA and the tolerance level is ± 0.10 mA. Thus, all the output values between 3.9 mA and 4.1 mA at 0 bar are acceptable. If all the calibration values are within the tolerance limits the calibration is successful, and the device does not need any adjusting to increase the accuracy of it. [3]

The calibration is failed, in other words out of tolerance, if the outputs of the IUC are not within the tolerance limits of the reference values. In this case, the first thing to do, is to make sure that the calibration was executed correctly. After that, one can investigate the suitability of the tolerance level. For example, a question such as "does the tolerance level match with the criticality of the application?" could be asked. In addition to that, one should analyze the calibration uncertainty factors which are elements that affect the accuracy of the measurement result, e.g., environmental factors such as temperature changes, human errors such as incorrectly executed calibration, and the uncertainty of the used reference standard. The uncertainty of reference standard can be evaluated by employing the traceability of the reference standard. It must be bear in mind that, every reference standard needs to be periodically calibrated with a higher-level reference standard, and the higher-level standard needs to be calibrated with an even higher-level reference standard, and so forth. The highest-level reference standard is the international reference standard. Tractability defines this path of calibrations with their uncertainties from the used reference standard back to the international reference standard. [2], [4]

Some common instrument errors, which cause calibration to be out of tolerance, are zero error, span error, and linearization error. These errors can be identified and analyzed by drawing the output values of the IUC to known inputs, and the corresponding reference values to an input-output curve. In zero error, all the measured values are sifted due to a nonzero output response to a zero input. In the case of the pressure transmitter this means that the output is less or more than 4 mA at 0 bar. In figure 1, both cases of the zero error in the input-output curve.

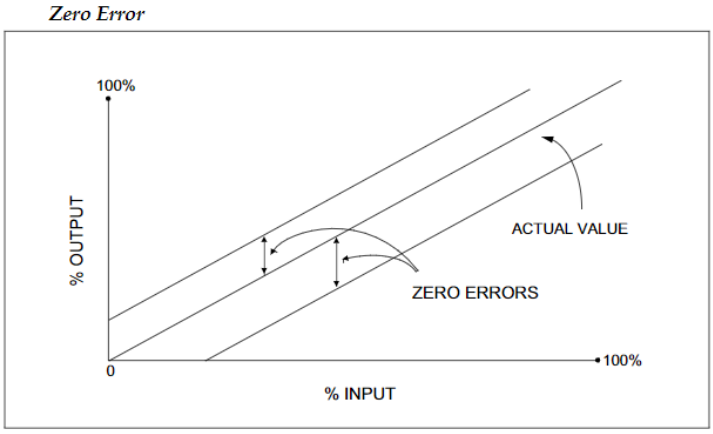

Figure 1. *Both cases of the zero error drew into the input-output curve* [2].

A span error is an error in the slope of the input-output function. Hence, the error is magnified as the output and input values increase. The Span error of the pressure transmitter could be seen as follows, the output is 4 mA at 0 bar but at 5 bar the output is more or less than 12 mA, and at 10 bar the input is less than 20 mA or is not measured, as the maximum output has been reached before the maximum input value. In Figure 2 both cases of the span error are presented.

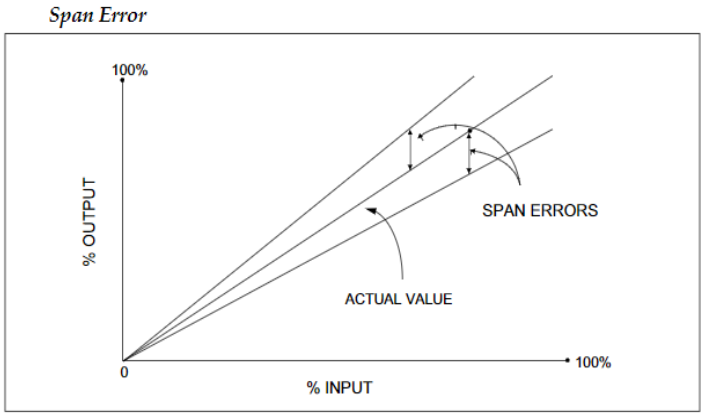

Figure 2. *Both cases of the span error drew into the input-output curve* [2].

A linearization error is seen as nonlinearity of the input-output-curve. In the case of the pressure transmitter, the output for 0 bar is 4 mA and the output for 10 bar is 20 mA, but the output for 5 bar is less or more than 12 mA. Both cases of the linearization error are illustrated in Figure 3.

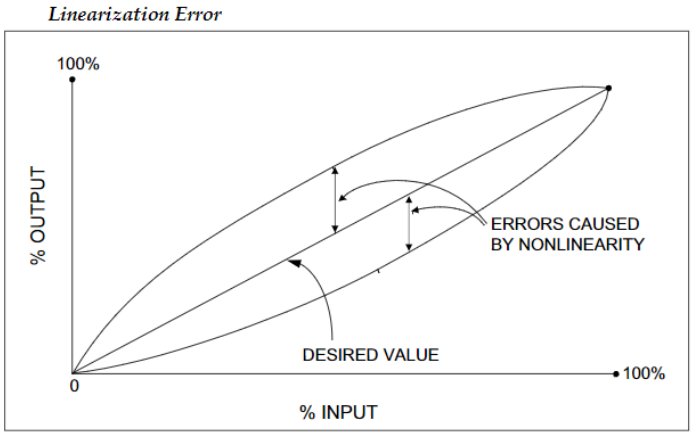

Figure 3. *Both cases of the linearization error drew in the input-output curve* [2].

All the mentioned errors can be corrected by adjusting the IUC. The zero error can be corrected by producing a parallel shift of the input-output curve. By adjusting the slope of the input-output function, the span error is fixed. Linearity adjustment can use to correct the linearization error. However, in case of large linearization error, the only option is to replace the instrument. [2]

2.3 Different Types of Calibration

In this subchapter, the questions about how and when to calibrate are discussed. Calibration needs to be repeated occasionally, because factors such as environment, drift, and process changes influence the measurement accuracy of the instrument. Suitable calibration intervals can be designed based on aspects such as criticality of the monitored process, manufacturer's recommendations, standards, intervals in similar instruments, and the previous calibrations of the instrument. [1],[4]

How the calibration is performed depends on the instrument and its location. An instrument can be linear, non-linear, or discrete. The earlier mentioned pressure transmitter is an example of calibration of a linear instrument. Calibration of a non-linear instrument, such as a square root characterizer, is more complicated than calibration of a linear instrument as more calibration points are required. Often nonlinear instruments have their own manufacturer's recommended calibration procedures. For discrete instruments, e.g., process switch, the purpose of calibration is to check that the state of instrument is changed at the correct input value. Moreover, the location of the instrument influences the calibration procedure, which can be dived into field and bench calibration based on the location in which the calibration is performed. Field calibration is performed at the site in which the instrument has been installed, which allows calibrating at the instrument operating environment. When the instrument is removed from its installation

place to be calibrated or a new instrument is calibrated before installation the procedure is referred to as bench calibration. [2], [3]

Furthermore, calibration can be done as a loop calibration or individual instrument calibration. In the individual instrument calibration, the input and output of the IUC are disconnected from the systems it operates, and the input is connected to a source that produces controlled known values. The individual instrument calibration allows adjusting of the specific instrument. However, it does not consider the tolerance of the whole loop, for that loop calibration, is needed. As the name implies, in the loop calibration, IUC kept connected to the other components in the loop. By doing so, the tolerance of the whole loop in the calibration can be analyzed. If adjustments are made after loop calibration, they will made for the whole loop. [2], [3]

3. BASICS OF SIGNAL GENERATION

The present chapter is dived into two subchapters: analog and digital signal generation. The fully analog signal generation methods are based on oscillators and are too preliminary to be employed as a signal generator in a multifunctional calibrator. However, they can be used to discuss some of the main problems that typically occur during the designing process of a signal generator. Some common problems are such as narrow output frequency range, as well as instability of the output frequency and amplitude. Most of the problems can be overcome by utilizing digital signal generation instead of analog signal generation. However, the conversion from digital to analog brings its own challenges, which are discussed in the second subchapter, which focuses on digital signal generation. Moreover, the second subchapter covers the operation principles of two digital signal generation methods which are later suggested as solutions for the current problem.

3.1 Analog Signal Generation

The traditional signal generators are mainly based on oscillators, which are circuits that consist of an amplifier and a feedback loop. The output of an oscillator is an oscillating signal, which can be either a sine, a square, or a triangular wave, depending on the oscillator type. Furthermore, oscillators can be divided into two categories; harmonic oscillators and relaxation oscillators which are presented in separate sub subchapters.

3.1.1 Harmonic Oscillators

Harmonic oscillators are sinusoidal signal generators, which types, and working principles are discussed in this sub subchapter. The basic operating principles of harmonic oscillators can be understood by viewing the harmonic oscillator as an amplifier with a feedback loop, and by applying the Barkhausen criterion which states that the output of a feedback amplifier circuit, presented in Figure 4, is sinusoidal if the gain of the full circuit is one and the phase shift between input voltage V_i and output voltage V_o is zero. The gain of the full circuit, which is also referred to as loop gain, consists of the gain of the amplifier A and the gain of the feedback circuit β. Moreover, if the loop gain is less than one the output is damped, and if the loop gain is much more than one the output signal is no more sinusoidal. The frequency at which the oscillator oscillates is

called as oscillator's resonance frequency and it is determined by the feedback loop as it induces a phase shift to the V_o in order to match it with the phase of V_i. [5], [6]

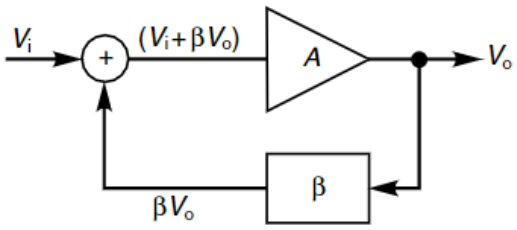

Figure 4. *A feedback amplifier circuit* [6].

Moreover, the harmonic oscillators can be divided into subcategories, and in this section, three common types of harmonic oscillators are presented. The first presented harmonic oscillator type is an oscillator whose feedback loop is a resistance-capacitor (RC) circuit, or an inductor-capacitor (LC) circuit. The amplifying element of RC or LC circuits can be e.g., an operational amplifier (op-amp), or a field-effect transistor (FET). Some well-known examples of this type of harmonic oscillator are a Wien bridge oscillator which is based on an RC circuit, and a Colpitts oscillator which is based on an LC circuit. As discussed earlier, the values of feedback loop components determine the resonance frequency of the oscillator. Therefore, components of the feedback loop must be altered, if it is desired to change the output frequency of the RC or LC circuit oscillator. In addition to being highly impractical to change the component values manually, the resistors and reactive components of the feedback loop cause instability of the output frequency and amplitude as the component values may vary due to the temperature, and input voltage changes. Furthermore, there can be small inaccuracies in the component values leading to an inaccurate output frequency. Moreover, the component value variations change the feedback loop gain to be less or more than one which causes the output signal to be damped or lose its shape. [5], [6]

The next presented harmonic oscillator type is a voltage-controlled oscillator (VOC) which provides quicker and easier alternation of the output frequency than the oscillators based on an RC or LC circuit. As the name suggests, the output frequency of VCO is controlled by the input voltage. In theory, all the oscillators based on RC and LC circuits could be turned into VCOs by replacing a capacitor in the feedback network with a voltage-controlled capacitor whose capacitance is determined by the input voltage. Therefore, by changing the input voltage, the output frequency is altered.

The third oscillator type is crystal oscillators, which have a very stable resonance frequency, unlike the LC or RC-circuit oscillators. In crystal oscillators, the feedback loop is a piezoelectric circuit that consists of the piezoelectric material such as Quartz crystal. Moreover, by applying an alternating electrical field across the crystal, it begins to

oscillate at a frequency determined by the size of the crystal. This phenomenon is called the inverse piezoelectric effect. The output frequency of crystal oscillator is very stable because the size of the crystal determines the resonance frequency of the crystal oscillator. There are crystals available with resonance frequencies from 10 kHz to 100 MHz. In addition to the size of the crystal, small changes in the resonance frequency can be done by adding a tuning capacitor in series with the crystal. Similar to the oscillators based on an RC or LC circuit the output frequency of the crystal oscillator cannot be quickly changed. Hence, the crystal oscillator by itself is not a suitable solution for the signal generator which is used as test equipment. However, it can be used as a reference frequency for frequency synthesis. [5], [6]

3.1.2 Relaxation Oscillators

In this section, the second type of oscillator which is the relaxation oscillator is discussed. In addition to that, this section presents simple methods, how one can transform a square wave into a sine, or a triangular wave. The benefits of the relaxation oscillators are the possibility to generate different waveforms simultaneously, easy alternation of the output frequency, and good amplitude stability because the Barkhausen criterion is not implied. Also, a stable output frequency can be provided by using a cheap and simple integrated circuits (IC) such as the 555 timer in the oscillator circuit. However, most of the basic relaxation oscillators cannot generate frequencies below a few hertz, which is desirable in the case of the Thesis.

Relaxation oscillators are oscillators, which oscillation is based on charging and discharging of a capacitor. In these oscillators, a capacitor is charged through a resistor, and the voltage across the capacitor is monitored by a comparator which compares this voltage to a threshold voltage. The value of the threshold voltage is defined by the comparator's output voltage and voltage divider. When the voltage across the capacitor reaches the threshold voltage, the output voltage of the comparator, (which is also the output voltage of the whole circuit), becomes reversed. This leads to the threshold voltage becoming reversed, and the capacitor begins to discharge in order to reach the new threshold voltage. When it reaches the new threshold voltage, the output voltage of the comparator is reversed again, and a new cycle of charging and discharging of the capacitor begins. Figure 5 describes the relationship between the voltage across the capacitor and the output voltage of the comparator. The output frequency can be controlled by the charging current of the capacitor, as it determines the speed of the capacitor charging. [6], [7]

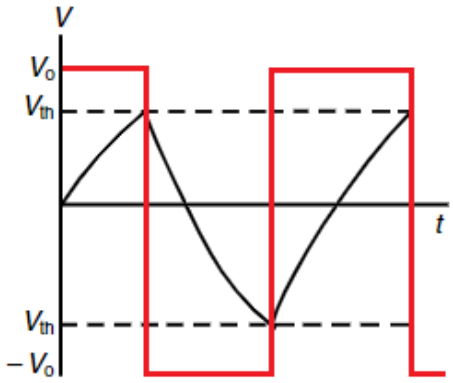

Figure 5. *The relationship between the voltage across the capacitor (black) and the output voltage of the comparator (red). The V_o is the output voltage of the comparator, and the V_{th} is the threshold voltage.* [edited from [6]].

Figure 6 illustrates a simple relaxation oscillator circuit in which the comparator is based on an op-amp, and two resistors (R_A and R_B). The threshold voltage is defined by the output voltage of the op-amp and the potential divider formed by R_A and R_B. The capacitor (C) is charged via the resistor (R). This type of oscillator circuit can be referred to as an astable multi-vibrator or a square wave generator. [6], [7]

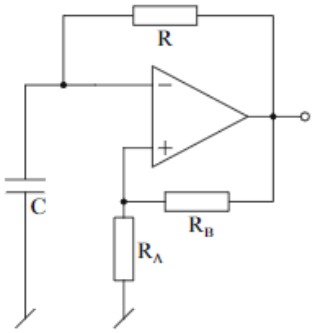

Figure 6. *An example circuit of an astable multi-vibrator* [7].

A triangular wave can be obtained from the astable multivibrator described above, by connecting the output of the op-amp of the circuit to an integrator. An integrator is an op-amp whose output is the integral of the input voltage. Thus, a nonlinear increase or decrease in the input leads to a linear increase or decrease in the output. [5]–[7] Hence, if the input of the integrator is a square wave the output will be a triangular wave. In addition to relaxation oscillators, this technique could be applied to other signal generators whose output is a square wave to generate triangular waves. Furthermore, a sine wave can be produced from a triangular wave by a sine function network, which utilizes diodes' electrical characteristics of becoming slowly conductive. [5] Another option is to use lowpass filtering to produce a sine from square wave by removing harmonic components of the square wave. Both forementioned methods for converting

other waveforms are quite robust and are not effective for input signals with wide range of amplitudes and frequencies.

3.2 Digital Signal Generation

This subchapter presents more advanced methods for signal generation that are based on generating a signal digitally and then converting it to an analog signal. Circuits that consist of both analog and digital parts can be referred to as mixed-signal circuits. The benefits of implementing a signal generator with a mixed-signal circuit instead of with a fully analog circuit are a smaller size, higher output frequency accuracy, and improved output characteristics such as wider output frequency range. Moreover, two methods employing mixed-signal circuits for signal generation are discussed in this subchapter. These methods are a method based on a microcontroller (MCU) and a method based on a direct digital synthesis (DDS) chip. The main focus is on the DDS as there is more scientific discussion around it. However, before going into details of these methods in sub subchapters 3.2.2 and 3.2.3, a component that is often the main limiting factor in both of the methods is discussed in sub subchapter 3.2.1. This component is a digital-to-analog converter (DAC) which is used to convert a digital signal into an analog signal.

3.2.1 Digital-to-Analog Converter

It does not matter how pure, accurate, and easily adjustable output signal it is possible to be generated digitally if it cannot be converted to an analog signal without compromising the characteristics of the signal. Therefore, the DAC is a critical component in the mixed-signal signal generators, and the characteristics of the DAC that affect the final output signal must be understood before discussing the digital signal generation methods. The DACs can be separated into two classes based on the output of the DAC which can be current or voltage. However, in the discussion of the basic characteristics and working principles of DAC, the quantity of the output does not matter, hence, in this section only voltage output is discussed.

3.2.1.1 Sampling Rate and Resolution

The influence of resolution and sampling rate of a DAC to the output signal of a signal generator are considered in this section. The resolution of a DAC is expressed as bits, e.g., a 16-bit DAC can generate 2^{16} different output values between the maximum and minimum output which are defined by the reference voltage of the DAC. In other words, the resolution of the DAC defines the number of different output values that can be used to represent a cycle of waveform. Moreover, the sampling rate of the DAC affects the number of samples that can be employed to represent a cycle of a waveform, because

the sampling rate of the DAC defines the number of sampling points that can be generated in a second. Therefore, the sampling rate of the DAC sets the maximum output frequency which is half of the maximum sampling rate of the DAC. [5], [8] Sampling rate of the DAC can be also referred as DAC speed.

In addition to the DAC speed, the generated waveform sets further demands for the sampling rate and resolution, which can be understood by comparing the generation of square and sine wave to each other. The frequency and alternation of the amplitude of a square wave can be accurately represented with only two different values, high and low. Hence, square waves can be easily generated even with low resolution and sampling rate, as the DAC must only alternate between two different output values. On the other hand, only the frequency of a sine wave can be accurately represented with two sampling points per cycle. To represent the amplitude profile of the sine wave, more sampling points per cycle are required, in fact, a pure analog sine signal has an infinite number of amplitude values between its lowest and highest points.

In order to increase the number of sampling points per cycle of a sine wave, a study by Wang L. et al. [9] employed a method called a time-interleaving DAC. In which four DACs are used to generate one period of a sine wave. The first DAC generates a part of the sine wave that is from the beginning to quarter period, and the second DAC from quarter period to half period, and so forth. By applying this technique, one period of the sine wave has four times more voltage levels than a sine wave produced by a single DAC. However, the benefit of employing multiple DACs depends on the desired output frequency as the number of useable voltage levels decreases as the output frequency increases. Hence, the benefit of using multiple DACs increases as the output frequency increases. However, the noise increases as the number of employed DACs increases. Therefore, at low frequencies the usage of multiple DACs can decrease the quality of the output signal, due to the increased noise. [9]

One possible technique that can be used to maintain the quality of the sine wave through the whole output frequency range is to employ different sampling frequencies for different output frequencies. In other words, use lower sampling frequency to produce lower output frequencies, and so forth. A method, proposed by Wang L. and Zhao Y. [10] widened the output frequency range by utilizing an arbitrary sample rate conversion (ASRC) to change the sampling frequency of the DAC according to the output frequency. The ASCR algorithms employ integration oversampling and continuous-time resampling filters to resample a signal [11]. In the study of Wang L. and Zhao Y. [10] the system was successfully implemented with a field-programmable gate array (FPGA), and they suggest that based on the results of the experiment their method is effective.

On the other hand, even if the highest possible resolution and sampling rate are applied in the sine wave generation, the output signal will still have the ‘stepped’ waveform appearance rather than continuous one. This is due to the sample-and-hold technique which DAC employs in the conversion process. In this technique the DAC holds the output at certain level for certain amount of time. This is not an issue in a square and pulse wave generation but becomes a problem in a sine wave generation as it induces high frequency components into the output signal. These components are called sin(x)/x roll of images, and their location in the frequency domain depends on the desired output frequency and the sampling frequency. The first image appears at sampling frequency deducted by the desired output frequency, hence by applying a lowpass filter with cutoff at the frequency of the first sin(x)/x roll image the output sine wave can be ‘smoothed’. [5], [8], [12]

3.2.1.2. Dynamic and Statical Errors of DAC

In addition to sampling rate and resolution of the DAC, the errors of the DAC affect the quality of the output signal. These errors can be divided into static and dynamic errors, and these errors are well-known and defined in datasheets of DACs. The statical characteristics of DAC that should be noted in the designing process of the signal generator are offset, gain error, and non-linearity. All of them can be obtained by comparing the ideal and actual transfer functions to each other. The transfer function describes the relationship between inputs and outputs of a system. The gain of the DAC can be seen as a slope of the transfer function. Therefore, the gain error is the difference between the slope of the ideal and actual transfer function. The offset error is the difference between a single point in the ideal and actual transfer function. [13] The gain and offset error of the DAC influence the amplitude accuracy of the output signal. The non-linearity of the DAC causes harmonic distortion in the signal. Distortion is the difference between an input waveform and an output waveform, and harmonics are frequencies that are integer multiples of the fundamental frequency. For example, if the fundamental frequency is 50 Hz, the 2^{nd} harmonics is 100 Hz, the 3^{rd} harmonics is 150 Hz, and so forth. With these definitions in mind, harmonic distortion can describe to be the amount of distortion in the output signal caused by the harmonics in the output signal. Hence, the output signal has the frequency components of the input signal, but also the harmonics of the input frequency components which induces the distortion. As the location of the harmonic components depends on the fundamental frequency, it is difficult to filter them out as the cutoff frequency should be adjusted every time the output frequency changes. Hence, it is easier to choose a DAC with good linear properties, than to try to filter out the harmonics. In addition to choosing DAC with good linear properties,

harmonic cancelation techniques can be employed to further decrease the harmonic components of the output signal. Total harmonic distortion (THD) is a percentage that defines the total amount of harmonic distortion in the output signal. Lower the percentage, less harmonic distortion in the signal. [5], [7], [13]

In the literature, different variations of harmonic cancelation techniques have been proposed. One popular harmonic cancelation technique that is described at least in studies [14]–[16] is to add periodical signal to its time-shifted and scaled versions with correct scaling factors that define the order harmonics that are canceled out. Malloug H. et al. [14] suggest that this technique can be implemented by generating phase-shifted digital square waves which are then combined and scaled to a sinusoidal signal by a summing network. For example, the study by Malloug H. et al. [16] applied phase-shift register to generate five time-shifted square waves and combined them with current-steering DAC with an output filter. The issue with this harmonic cancelation technique arises from the required high accuracy of the scaling factors. Malloug H. et al. [14] reported that even small errors in the scaling ratios between different phase-shifted signals reduce the effectiveness of this technique.

Dynamic errors of DACs are discussed in this paragraph and following paragraph. A slew rate of DAC is the maximum rate at which output of the DAC can be changed. Issue with slew rate is that if the slew rates of the increase and decrease are different, spurious harmonics (spurs) that are non-signal components that exists only in one frequency are induced to the output signal. The slew rate of increase is illustrated in Figure 7, also settling time is presented in that Figure. The settling time describes the time it takes for the output of the DAC to stabilities after it has been changed. It can be expressed as the time it takes to stabilize to a static value within a defined error band. For example, if the settling time is 1 µs to ± ½ least significant bit (LSB) it takes 1 µs to stabilize to a static value with ± ½ LSB error band. [13], [17] To have highly stable output frequency, the settling time should be as short as possible and the error band as small as possible. Moreover, the settling time influences the maximum DAC speed.

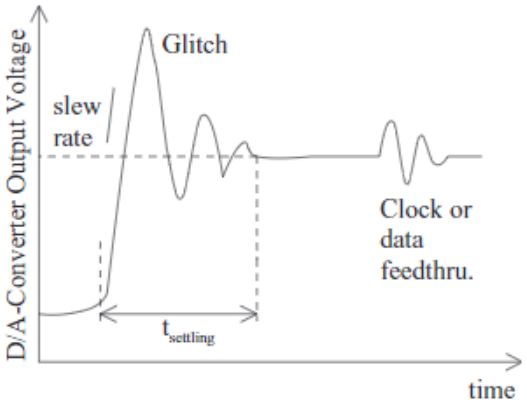

Figure 7. *Slew rate, glitch, settling time ($t_{settling}$), and clock or data feedthrough illustrated in a voltage–time curve of a DAC's output* [17].

Furthermore, during the settling time, glitches can occur at the output, especially when there is a large change in the output. Glitches can be negative or positive, in Figure 7 positive glitch occurs during the settling time. The Glitch energy, also referred to as glitch impulse is the total area under the glitch in a voltage-time curve. The size of the glitches increases as the output frequency rises because there are fewer output samples of the DAC that can be used to present the digital input as an analog output. Study by Richen J. et al [18] propose three types of DAC architectures that reduce the size of the glitches. The dynamic characteristics of DAC that influence the quality of the output signal are slew rate, glitch energy, and settling time. The three types are a voltage-mode, a current-mode, and a current-steering DACs which all input gray code instead of the natural binary code. In gray code, the difference between two successive bits is only one bit, e.g., in 4-bit natural binary code one is 0001 and two is 0010, but in gray code in one is 0001 and two is 0011. According to the Richen J. et al [18] by replacing the natural binary code input of DAC by gray code input the size of glitches are reduced. The reason for this, is that in gray code transient state of zero which occur switching the input values can be avoided, unlike in natural binary code. [18] In Figure 7, the clock or data feedthrough is also illustrated which is an interference in the DAC output signal caused by clock or/and data signals coupling to the DAC's output. Proper layout and fabrication methods can eliminate clock and data feedthrough. [13], [17]

3.2.2 Microcontroller Based Method

A simple method for square wave generation with a MCU is to employ one of its timers to generate square waves at the desired frequency from the clock frequency. This technique can be explained by going through the operation principles of timer. A basic timer contains two registers; a current and reload register. The current register keeps track of time by counting the input clock pulses of the clock signal, and the reload register

determines the length of the intervals between the output interrupts. Moreover, in a countdown timer, the current register counts from the value of the reload register to zero, and in a count-up counter from zero to the value of the reload register. [19] When the counter reaches zero or the value of the reload register, the state of the output changes. Hence, by controlling the value of the reload register, one can control the frequency of the output square wave of timer as the value of the reload register determines the number of input clock pulses that a half cycle of the output square wave consists of. Furthermore, the output frequency range of the timer depends on the value of the reload register and the input clock frequency. Therefore, the output frequency can be calculated by the following equation

$$f_{out} = \frac{f_{clk}}{m} \quad [20](1)$$

in which f_{out} is the output frequency, f_{clk} is the clock frequency, and m is the value of the reload register. The maximum value of the reload register depends on the size of the timer, e.g., in 16-bit timer it is 2^{16}-1. The value for the m can be calculated by dividing the half period of the wanted output frequency by the period of the clock pulse and rounding it up to the closest integer. The duty cycle of the square wave can be altered to obtain pulse waves by separately defining the time for the high and low time. Furthermore, the value of the reload registers must be updated after every timer overflow, to set the length of the next half cycle.

Moreover, the resolution of the output frequency is the length of one clock pulse which is determined by the input clock frequency. Thus, with a higher clock frequency, a higher resolution can be achieved. However, as the clock frequency increases, also the minimum output frequency increases. Hence, the used clock frequency is often compromise between the output frequency range and the resolution. By increasing the size of the timer, it is possible to expand the output frequency range. One way to increase the timer size is to use a prescaler, which allows user to define the number of clock pulses that are counted as a one pulse by the timer. For example, it can be set that the value of the current register changes every fourth clock pulse.

Furthermore, the generated square wave can be used in the generation of other waveforms, such as a sine wave, by using it as a sampling frequency which determines the speed at which sampling points of a sine wave are transferred to a DAC. In this technique the sampling points of the desired waveform, which is a sine wave in this case, are stored in a look-up table (LUT). The values for the sampling points can be calculated by using following equation

$$y(n_{sample}) = \sin\left(n_{sample} * \frac{2\pi}{n_{tot}}\right) \quad (2)$$

in which n_{tot} is the total number of samples in one period and n_{sample} is the number of samples that is calculated. Hence, n_{sample} gets values from 0 to n_{tot} if the calculated sine wave is symmetric. Furthermore, the result can be multiplied to manipulate the amplitude of the wave. The same look-up table can be used to generate all the generatable output frequencies as the sampling points in the LUT only consists of the y-coordinate of the sine wave samples in the time-domain. The x-coordinate in the time-domain, which is the location of samples in time, is defined by the speed at which the samples are transferred from LUT to DAC and that speed is determined by the sampling frequency. [21]–[23]

Furthermore, the sampling points can be transported from LUT to DAC by utilizing a direct memory access (DMA), which is a system that can be used to transfer data between different memory regions. It consists of a control block and multiple channels. The DMA controller has a control register that includes information about the transfer, such as the length of the transferred data. Additionally, it has source and destination registers which defines from where-to-where the data is transferred. The benefit of using the DMA for data transfer between memory regions is that it requires minimal main processor unit (CPU) intervention, hence, freeing it for other tasks. Moreover, the CPU is only required for initializing the DMA controller registers. After that, the actual transferring can be triggered by other events such as timer interrupts. Another, benefit of the DMA is that it can be used to transfer data at a rate close to clock frequency. [21]–[23]

To conclude, the method can be described as follows; a DMA is configurated to transfer data from a LUT to a DAC, after configuration, every time a timer triggers a DMA request a sample from the LUT is transferred to a DAC. Therefore, the rate at which the requests are triggered defines the frequency of the output signal of DAC. As discussed above the output frequency of a timer can be controlled, therefore, the output frequency can be controlled by controlling the timer's output frequency. Moreover, the following equation can be driven for the output frequency

$$f_{out} = \frac{f_{clk}}{divider} * n_{samples} \quad (3)$$

in which the *divider* is the value of the clock cycles one half cycle of timer output consists of.

The method of employing timer to generate square and pulse waves, as well as usage of DMA and look-up table to generate different waveforms is quite simple, well-known, and is widely covered by the application notes of MCU manufacturers and blog posts of electronic enthusiasts. However, during the literature search of the present Thesis no article presenting solution that combines both methods in a way described above were found. Also, there is hardly any scientific papers post on the either of the methods. Also, component limitations such as the clock frequency and the DAC speed limits the output frequency and resolution and the number of sampling points used. Trying to overcome these restrictions can quickly make the system much more complex, which is further discussed in subchapter 5.2.

3.2.3 Direct Digital Synthesis Based Method

In the present subchapter the basic working principles of DDS are discussed. In addition to that, a short history of DDS and its overall advantages and disadvantages are presented. DDS is not a new invention, first time it was presented by Tierney J., Bader C., and Gold B. in the year of 1971 [24]. However, back then, IC technology was not advanced enough for the DDS. For example, it was difficult and expensive to implement and had high power consumption. However, after that huge steps have been taken in IC technology. Although, the resolution and speed of the digital-to-analog converter (DAC) are still the major factors that limit the capabilities of the DDS. Regardless of that, when compared to other frequency synthesis methods, DDS has a high output frequency resolution (up to μHz), rapid frequency switching, low phase noise, and high stability. Furthermore, being a digital signal processing method gives DDS benefits such as easy and fast modulation of output phase, frequency and amplitude, small size, and freedom from component variation problems. [17], [25], [26] The theoretical aspects of DDS discussed in the section are mainly based on the Doctoral Thesis of Vankka J. [17] and study of Qi J. et al. [27] report.

3.2.2.1 Working Principle of a Conventional Direct Digital Synthesizer

The four building blocks of a basic DDS are a phase accumulator, a phase to amplitude converter, a DAC, and a filter. Figure 8 is a block diagram of DDS in which the connection between the bulling blocks can be seen, and it also illustrates the output of each block.

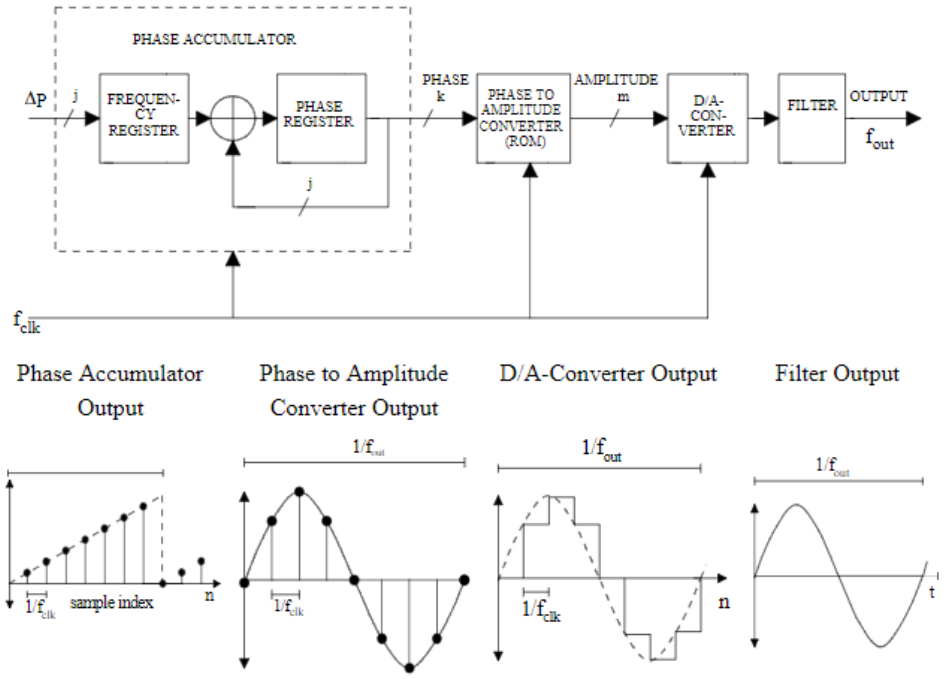

Figure 8. *A block diagram of a conventional DDS* [17]

The first block in Figure 8 is a phase accumulator that consists of an n-bit frequency register, an n-bit full adder, and a phase register. The operation principle of the phase accumulator can be explained by using a phase wheel which is illustrated in Figure 9. In the Figure, a vector revolves around the wheel, like clock hands in a clock and very point on the wheel corresponds to a different point on a sine wave. These points are represented as yellow dots in Figure 9. The rotation of the vector around the phase wheel at constant speed generates sine waves in which one cycle is one period of a sine wave. In the phase accumulator, the current phase value is stored in the phase register, and it is the current location of the vector at the phase wheel. At each clock pulse the phase increment word *M*, which is held in the frequency register, is added to the previous phase value by the full adder. This causes the vector to move on the phase wheel. This addition process is repeated until the vector has rotated around the whole phase wheel that causes the phase accumulator to overflow. The rate of the overflow defines the output frequency. Moreover, the value of *M* defines the number of points of the wheel that crossed over at each clock pulse. A large *M* means that more points are jumped over at every clock cycle, hence fewer clock cycles are required for the overflow, and faster one cycle of a sine wave is generated. Thus, by adjusting the value of *M*, the output frequency can be changed to a desired value.

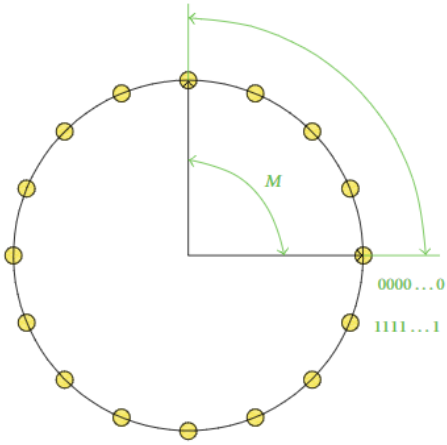

Figure 9. *The operation principle of phase accumulator can be explained by using a phase wheel* [27].

The second block in Figure 8 is a phase to amplitude converter which is in the conventional DDS is read only memory (ROM) which holds samples of a sine wave in a look-up table. This look-up table is used to convert the phase information received from the phase accumulator to amplitude values of a sine wave. The amplitude information, which is held in the phase to amplitude converter, is then transferred to the DAC that generates quantized analog sine wave based on the amplitude values. The filter presented in Figure 8 can be a simple low-pass filter that removes the high frequency sampling components from the signal. [17], [27]

DDS being a digital signal processing system has benefit of easy modulation of phase, amplitude and frequency. These three modulations are applied to the signal under generation at different locations of the DDS process. The frequency modulation is applied before the phase accumulator block. As discussed above the output frequency can be modulated by adjusting the phase increment word M. The phase modulation is applied on the output of the phase accumulator, between phase accumulator and phase to amplitude converter block. The amplitude modulation is carried out by adjusting the binary amplitude information of the phase amplitude converter's output before it enters to the DAC. In addition to that, another way to implement amplitude control, is to utilize a voltage-controlled DAC which analog output amplitude can be adjusted by varying the control voltage. [17]

3.2.2.2 Output Signal Charachteristics of Direct Digital Synthesizer

In this section theory of DDS, and studies that have designed DDS-based signal generators are used to discuss the following topics: output frequency resolution and range, output amplitude range, generatable waveforms, and pulse generation. Let's begging by analyzing the output frequency. The output frequency resolution depends on

the clock frequency f_{clk} and the number of bits used in phase accumulator n which is usually referred as tuning word. The equation for output frequency resolution

$$Frequency\ resolution = \frac{f_{clk}}{2^n} \tag{4}$$

which is true if the phase increment word is one. The output frequency accuracy of the DDS system is determined by the *M* as it determines the number of sampling points of the phase wheel employed in the generation process. In other words, the *M* determines, how many sampling points of the phase wheel are jumped over at every increment. As discussed earlier, to generate higher frequencies, the value of *M* must be increased. Furthermore, the value of the *M* affects the amount of harmonic distortion in the output. Therefore, the THD in the output signal is different for different frequencies.

Output frequency f_{out} can be calculated by following equation

$$f_{out} = \frac{Mf_{clk}}{2^n} \tag{5}$$

which only true if the output frequency is less or half of the clock frequency. The theoretical maximum output frequency is half of the clock frequency. The theoretical minimum frequency can be calculated by setting the value of M to be one in the equation 5. However, these are only theoretical values, factors such as noise factors and component limitations restrict the upper and lower limits of the output frequency range. [17], [27] For example, the signal generator designed by Qi J. et al. [27] has output frequency range from 0.1 Hz to 12.5 MHz with 0.05…0.1 Hz resolution. The upper output frequency limit is based on the theoretical maximum output frequency. However, according to the study the actual maximum output frequency is around 10 kHz, after it the output amplitude begins to decline due to component limitations. The signal generator designed by Wey T. S. et al [28] suffered from similar problem, however they were able to generate frequencies up to 1 MHz with an appropriate amplitude accuracy. Although, the clock frequency of the used DDS chip employed in this study was 125 MHz which sets the theoretical maximum frequency to be 62.5 MHz, which is much more than the obtained 1 MHz. Moreover, Nurul A. et al. [29] employed the same DDS chip AD9850 than Wey T. S. et al [28] in their implementation and came to same conclusion, as the output frequency increases, the output amplitude decreases. In other words, the maximum output frequency limitations do not come from the operation principles of the DDS, rather from component limitations. In the Wey T. S. et al [28] implementation the restricting component was the DAC of the DDS chip. It could be assumed that this was also the reason in Nurul A. A. M. N. et al. [29] case. In the design proposed by Qi J. et al. [27], the limitations of the multiplier were the reason for amplitude decline after 10

kHz. Therefore, it can be assumed that maximum output frequency of DDS based signal generator is lower than half of its clock frequency. Based on the literature search of capabilities of DDS done in this Thesis, it was difficult to approximate the minimum output frequency, because most studies were more interested in the maximum than the minimum output frequency. In all of the reviewed studies the minimum output frequency was either 0 Hz, 0.1 Hz, or it was not mentioned [27]–[33].

The operation principles of the DDS do not set the output amplitude range. It depends on factors such as the used DAC and the amplification circuit. The output amplitude range of reviewed studies variates a lot. The output amplitude range of Qi J. et al. [27] signal generators is from 0 to 4.5 V. Also, negative amplitudes can be generated, a signal generator designed by Zhenyu Z. et al. [32] generates amplitudes from -4 V to 4 V. They employed digital potentiometer to change the unipolar input signal to a bipolar output signal which allows the generation of both negative and positive amplitude values. Likewise, higher output amplitudes can be generated, e.g., Chen X. and Chen J. [30] implementation was able to reach 15 V peak amplitude.

Waveforms that can be generated with DDS technology depend on the characteristics of the DDS chip. For example, DDS chip AD9833 employed in study by Zhenyu Z. et al. [32] could generate a triangular, a square, or a sine wave. Another option is to use the DDS chip to only generate one type of waveform, and then by signal conditioning generate the other waveforms from it. This method is used in signal generator design by Qi J. et al. [27], in which the DDS chip is used to generate sine waveform and an additional circuit with a Schmitt trigger is used to transform the generated sine wave to a square wave if wanted. Neither of the above-mentioned studies discussed why they choose to directly use or not use the DDS chip to generate different waveforms.

In terms of the pulse generation, during the literature search of the Thesis, no such DDS-based signal generator that is able to do both generation of periodical and pulse signal was found. Although square wave generation was possible in most of the implementations, a user could not adjust the duty cycle which is the essential difference between square and pulse wave generation. However, few papers presenting DDS-based pulse generators were found. However, these circuits cannot be used to generate periodical signal, if the square wave which can be viewed as a pulse wave with a 50 % duty cycle is not counted [34], [35]. Suchenek M. and Starecki T. [34] used programmable counters to adjust the duty cycle and period of a square wave. The design includes three counters implemented on a field-programmable gate array (FPGA). The circuit had two outputs, one for pulse trains, and one for trigger pulses. One of the three counters is used to control the number of pulses generated, and another one to modulate

the duty cycle of generated pulses. The third counter is used as a trigger delay counter, this counter produces a pulse that ends after it reaches a specified value. Specifications for the generatable pulse waves weren't well specified in the paper. Other than concluding that the pulses with accurate timing and quasi-continuous tuning could be generated at below 1 Hz to 620 MHz. The output amplitude modulation was not discussed in the text. Sharm A. et al. [35] proposed an even more advanced pulse generator in which a user could, in addition to duty cycle modulation, adjust the rising and falling edges. In their design they used mathematical methods to convert the phase accumulators phase signal to pulse wave with desired duty cycle and rise and fall time. This technique replaced the look-up table of the phase to amplitude block. Although, no DDS-based system that could generate both periodical signals and pulse trains was found. It does not mean that it is not possible to design a DDS-based system that can do both of those things. For example, the system could have two separate circuits with their own DDS chips, one for the generation of periodical signals and one for the pulse generation.

3.2.2.3 Noise properties of Direct Digital sythesis

In addition to spur induced by DAC of the DDS-based signal generator, the DDS technology itself induces some supers due to phase truncation error, and the amplitude quantization error. Furthermore, the phase noise properties of DDS are shortly discussed in this section. To understand truncation error, one must first understand truncation which is a quantization method in which all the bits that are less significant that the LSB are ignored aka "truncated". Truncation is used for the output of phase accumulator before ROM receives it. The reason for the truncation is to reduce the size of the look-up table. However, this causes phase truncation error which leads to spurs in the output signal, which are referred as phase truncation spurs. The factors that affect phase truncation error are the number of phase accumulator bits n, the number of bits used ROM k, and the size of the phase increment word M. The phase truncation does not occur if the greatest common divisor of the M and 2^n equals to or is larger than 2^{n-k}. [17] [13] Another source for spurs is the amplitude quantization error that happens in the ROM. The amplitude quantization error occurs due to the fact the ROM tries to transform phase information from the accumulator to a continuous sine wave by using discrete samples. Generally, the effect of amplitude quantization error on the spurs of the output signal is much smaller than the effect of DAC errors, and the phase truncation. [27]

4. NEEDS AND REQUIREMENTS FOR THE SIGNAL GENERATION

The aim of the present chapter is to define requirements for the frequency and pulse generation of a multifunctional calibrator. As the frequency measuring and pulse counting are strongly connected to these generation functions, the requirements for those are also investigated in the present chapter. Moreover, the requirements are defined by using two different methods. The first method is to examine the capabilities of current multifunctional calibrators to generate and measure pulse and frequency. The second method is to conduct a customer needs analysis by interviewing experts and reviewing customer feedback gathered in the quality feedback database of Beamex. The final specification suggestion is formed by combining the information and specification suggestion gathered by using these two methods. The final specification suggestion is presented in subchapter 4.3.

4.1 Current Solutions

The first method used to define the requirements is to review the specifications of solutions that are currently on the markets. Moreover, solution refers to techniques that are presently available in multifunctional calibrators for signal generation, frequency measurement, and pulse counting. The solution ideas suggested in this Thesis should be able to compete with solutions that are currently on the markets. Frequency and pulse measurement and generation features of multifunctional calibrator MC6 are used as Beamex's current solution in this review. Also, Additel Corporation's ADT227, Fluke's Fluke 745, and Durck's DPI 620 Genii are analyzed. They were recommended to be included to this review by a product manager at Beamex.

From Table 1, it can be seen that the common characteristic in all four calibrators is the frequency range which is from below one hertz up to 50 kHz. MC6 and ADT227 has the best resolution for frequency generation, to compete with them, the resolution of frequency generation must be more than 5-digits. The waveform options in the frequency generation function should be positive and symmetric square wave, as well as sine wave, to cover all popular waveforms. The output amplitude range for positive square waveform needs to be from 0 to around 20 peak-to-peak voltage (V_{pp}). For the symmetric square waveform, the maximum output amplitude could be about half of the maximum output amplitude of the positive square waveform. For the sine waveform maximum output

amplitude could be higher than 20 V_{pp}, and the amplitude accuracy for all waveforms less than 5 % of the V_{pp}. [36]–[40]

Table 1. *Frequency generation specification of calibrators included in this review* [36]–[40].

Device (company)	MC6 (Beamex)	ADT227 (Additel)	Fluke 754 (Fluke)	DPI 620 Genii (Druck)
Frequency range	0.0005...50000 Hz	0.01...50000 Hz	0.01…50000 Hz (Square) 0.1…50000 Hz (Sine)	0…50000 Hz
Resolution	0.000001 (0.0005…0.5 Hz) 0.00001 (0.5…5 Hz) 0.0001 (5…50 Hz) 0.001 (50…500 Hz) 0.01 (500…5000 Hz) 0.1 (5000…50000 Hz)	6-digits	information not available	0.1 (0…1000 Hz) 0.001 (1000…50000 Hz)
Waveforms	Positive and symmetric square	information not available	Symmetric sine and positive square	Square, sine, and triangular
Output amplitude	0…24 V_{pp} (Positive square) 0…6 V_{pp} (Symmetric square)	information not available	0.1…30 V_{pp} (Sine) 0.1…15 V_{pp} (Positive square)	0…20 V_{pp} (Square)
Amplitude accuracy	< 5% V_{pp}	information not available	1% V_{pp} + 75 mV (Square, 0.01…1000 Hz) 10% V_{pp} + 75 mV (Square, 1000…50000 Hz) 3% V_{pp} + 75 mV (Sine)	V_{pp} ±20 mV

Based on the datasheets of the calibrators, it seems that all calibrators except Fluke 754, are capable of pulse generation [36]–[40]. In pulse generation, operator can define the number of pulses generated in certain time frame instead of defining the frequency of the generated signal. Thus, the highest possible resolution is one pulse. The output amplitude ranges for pulse generation are same that output amplitude ranges for square wave in frequency generation. A major finding found in the comparison of pulse generation features of the calibrators was that none of the calibrators were capable of dual pulse generation.

As it can be noted from the Table 2, the frequency measurement specification is similar in all the four calibrators [36]–[40]. Based on the specifications, the frequency measurement function in multifunctional calibrator must be able to measure frequencies

from below 1 Hz to 50 kHz and display the measurement result with more than 5-digits resolution. Additionally, have the minimum amplitude level around 1 V_{pp}.

Table 2. *Frequency measurement specification of calibrators included in this review* [49]–[53].

Device (Company)	MC6 (Beamex)	ADT227 (Additel)	Fluke 754 (Fluke)	DPI 620 Genii (Druck)
Frequency range	0.0027...51000 Hz	0.01...50000 Hz	1…50000 Hz	0…50000 Hz
Resolution	0.000001 (0.0027…0.5 Hz) 0.00001 (0.5…5 Hz) 0.0001 (5…50 Hz) 0.001 (50…500 Hz) 0.01 (500…5000 Hz) 0.1 (5000…51000 Hz)	6-digits	0.01 (1…110 Hz) 0.1 (110.1…1100.0 Hz) 0.001 (1101…11000 Hz) 0.01 (11010…50000 Hz)	0.0001 (0…1000 Hz) 0.00001 (1000…50000 Hz)
Minimum trigger level	1 V_{pp} (< 10000 Hz) 1.2 V_{pp} (10000…50000Hz)	2.5 V	0.3 V_{pp} (1…10000 Hz) 1.4 V_{pp} (1000…30000 Hz) 2.8 V_{pp} (> 30000 Hz)	information not available

Datasheet of every calibrator, except Fluke's, mention ability to count pulses [36]–[40]. After reviewing the pulse counting capabilities of the already mentioned calibrators. It can be stated that, to have competitive pulse counter, both rising and falling edges should be able to be detected and the trigger level should be around or below 2 V_{pp}. Furthermore, pulse counter should be able to count signals that can have frequency up to 50 kHz.

Lastly, it must be noted that there is no evidence of the lower or upper limits of these specifications to be actually required in any field application. The specification suggestions made in this subchapter are solely based on the competition aspect and for the need to keep the specifications similar to the MC6's ones.

4.2 Customer Needs Analysis

The second method used to define the requirements is to conduct a customer needs analysis. The analysis is done by interviewing experts and analyzing issues concerning frequency and pulse generation and measuring listed in the quality feedback database of Beamex. Moreover, total of seven experts were interviewed. Experts were working at Beamex during the interviews. Interviews were executed either via emails or as a online meeting in Teams. In the case of a Teams meeting, the interviewee received the

questions discussed in the interview prior to the meeting. The purpose of the questions was to find, what are the most common applications in which the pulse or frequency features of MC6 are used, and what requirements those applications set to MC6. Moreover, experts were asked to describe the overall feedback which they have received from the customers concerning the pulse and frequency features of MC6. Also, experts' own views were noted. Most experts found the questions to be difficult to answer, as these features are rarely used when compared to other features of MC6. Another factor that makes it tricky to evaluate the overall need for these features, is the fact that they are standard features of MC6.

4.2.1 Validation of Flowmeters

Every expert mentioned validation of flowmeters in the gas and oil industry as major application area for frequency and pulse generation and measuring. Furthermore, the customers operating in other areas don't see these features as important as the customer operating in the gas and oil industry. Therefore, the gas and oil industry's requirements for signal generation, pulse counting, and frequency measuring must be considered in the process of setting the specification for frequency and pulse generation and measuring. In the present Thesis, validation of flowmeters refers to procedures utilized to check the measurement accuracy and correctness of a flowmeter or flow computer. Before going into detail of these procedures and discussing the requirement they set, the basics of flowmeters are discussed.

As the name suggests, flowmeters measure flow. Flow can refer to volumetric flow (volume transported per unit time) or mass flow (mass transported per unit time). The flow can be defined as a flow rate, which can be presented as liters per minute (*l*/min). In addition to the flow rate, flowmeters can present the total amount of substance passed over the sensor in a given time frame. This feature of a flowmeter is referred to as a totalizer. The measured substance can be gas, vapor, liquid, or slurry. Different type of flowmeters utilizes different operation principles to measure the flow. For the purpose of the Thesis, only flowmeters that can be simulated by either pulse train or with other periodic signal are considered. In these types of flowmeters, flow is converted to an electrical signal, in most cases to a pulse train. The number of pulses in a time frame depends on the flow rate. A simple example of this type of flowmeter is a turbine flowmeter, in which a free-spinning turbine is rotated by flowing substance. The velocity of the rotating turbine is directly proportional to the flow velocity of the substance. The velocity of the rotating turbine is converted to a pulse train by using magnetic pickup which is then converted to flow rate and total flow. [3], [41] It must be bear in mind, that

the flow is complex variable to measure due to multiple factors effect on it. These factors are such as viscosity of the measured fluid, pressure, temperature, and shapes of the flow guiding system. To increase the accuracy of the flow measurement, a flow computer can be added to the flow measuring system. The flow computer corrects the flowmeters measurement result by taking into consideration other characteristic of the measured substance e.g., temperature and pressure of it, in the calculation of flow. Thus, in addition to a flowmeter, other sensors are connected to the flow computer to collect the information needed for the calculation of flow. [41] When validating a flow computer all the sensor outputs used to achieve the flow measurement must be considered.

The validation of flowmeters in gas and oil industry is especially important in custody transfer points. In custody transfer points, the ownership of the gas or oil exchanges. Based on the measurement of the flowmeter, both parties of the transaction know the amount of substance exchanged. [42] An inaccurate flowmeter measurement can cause the customer to pay too little or too much. Furthermore, the security level of the pulse data transmission system of a flowmeter affects the validation process. There are five different security levels for pulse data transmission systems of flowmeter which are defined by ISO 6551 standard. The three highest levels include dual pulse data transmission lines, which improves the error monitoring capability of the system, because the pulse trains from transmission lines are compared to each other and if the number of pulses is not the same in both signals, an alarm is given. The lower levels have only one transmission line. The standard does not define which level of security should be used in certain applications. [43]

According to the experts, in the flowmeter validation process the component that converts the flow to an electrical signal, e.g., turbine in turbine flowmeter, is replaced by a pulse or a frequency generator. In the case of validating a totalizer which has a low-security level, a known frequency is generated to the computational unit of the flowmeter for a certain amount of time. The output of the totalizer depends on the number of pulses it has counted during the generation time. Totalizer's output is compared to a reference value which is the value obtained when all the generated pulses are detected by the flowmeter. Frequency generation can be replaced by pulse generation to avoid timing problems, which occur when timed frequency generation time and actual frequency generation time differ from each other. In pulse generation, the generation time is irrelevant as the duration of generation is defined by the number of pulses generated. The process of validating flow rate is similar to validation of totalizer but only the frequency of the generated signal is important, not the generation time. To validate higher security level systems dual pulse generation is needed. A test in which dual pulse

generation is used to check if added or missing pulses cause measurement uncertainty in the flow totalizer is referend as fidelity check. [44] When dual pulse generation is used, the missing or added pulses caused by an error in the transmission line are seen a difference in the number of pulses in the two lines. In lower security level systems, the error in the transmission line causes the number of pulses counted by the totalizer to differ from the number of generated pulses.

It is difficult to set requirements for the signal generator needed in flowmeter validation because the input signal of the flowmeter's transmitter or flow computer variates a lot between devices. Furthermore, generated signal depends on if it is used for testing flow rate or flow totalizer, and the ISO 6551 security level of the system under the test. Following specification for a signal generator that can be used to validate flowmeters was derived by asking experts opinions, reading datasheets of some popular flow computers, and using the results of the questionnaire regarding frequency generation conducted in the year of 2020 [45]. Participants of the questionnaire were Beamex's salesmen. The signal generator should be able to generate at least pulse waves with adjustable duty cycle. The user should have an option to define the generated signal by its frequency or by the number of pulses generated with a certain duty cycle. In the case which signal is characterized by its frequency, it would be handy if user could also define the generation time. This feature eliminates the timing problem which was discussed earlier in this chapter. Signal generator's frequency range could be up to 15 kHz. Although, for the most flowmeters, the maximum input frequency is something between 10 kHz and 15 kHz, only frequencies up to few kilohertz are used in simulation of flowmeters. Amplitude range could be from 0 to 10V with 0.1V resolution. Furthermore, the signal generator should be able to generate dual pulse if it is desired to be used validate a flow measurement system with a high ISO 6551 security level. The phase shift between the dual pulse must be able to be controlled. Also, it must be noted that gas and oil customers usually use devices that fulfill the ATEX directive. Devices that follow the ATEX directive are safe to use in a potentially explosive atmosphere. Currently, Beamex has separate versions of their calibrators which fulfill the ATEX directive. For example, there is MC6-Ex which has similar properties than MC6 but fulfills the ATEX criteria. Due to the ATEX, some features are downgraded in MC6-ex, e.g., maximum frequency have been decreased from 50 kHz to 10 kHz in MC6-Ex.

4.2.2 Other Applications

In addition to flowmeter validation, testing of tachometers came up in several interviews. Tachometers are instruments used to measure rotation speed. There are different types

of tachometers, but the validation process is similar, the rotating component is replaced by a frequency generator. The reference revolutions per minute (RPM) value can be calculated from the generated frequency and compared to the RPM value measured by the tachometer under testing. Setting requirements for the frequency generator used in tachometer validation is tricky because there are multiple different types of tachometers. One hertz equals to 60 RPM, thus it could be estimated that frequency range of the generator could from below 1 Hz up to 20 kHz.

Another application related to rotation speed is checking the performance of a speed switch with a frequency generator. The speed switches are used to detect abnormal changes in the rotation speed of a rotating shaft. Rotation of shaft is converted to a pulse train by using e.g., magnetic pickup sensor and fed to the speed switch. If the switch detects abnormal input signal frequency, for example too high frequency caused by too high rotation speed of the shaft, it triggers an alarm. [46] Generation of ramp signal could be useful for this type of test. Also, other types of switches and relays like on/off switches can be tested by a pulse generator.

In addition to applications mentioned above, there were few specific applications requiring sine wave generation mentioned in the Beamex's quality feedback database. To fulfill requirements of the mentioned applications, possibility to generate sine waves with frequency range from 0 to 50 kHz with 10 V_{pp} output amplitude is required. [47], [48] In the interviews, these applications did not come up. However, few experts said that there are a couple of requests for sine wave generation every year. Based on these facts, a possibility to generate sine waves should be added to the future frequency generation solution only if it can be done cost effectively and so that it does not negatively effect on other features.

Experts found it difficult to define specific applications in which frequency measurement and pulse counting features of MC6 are used. Like signal generation, these features are most likely used for testing and troubleshooting systems more often than in the actual calibration procedures. Another factor that makes it hard to pinpoint any specific applications for these measurement features, as well as for the signal generation feature, is the fact that they are basic features in MC6. Therefore, there can be multiple applications for these features which experts are not aware of. One thing that is good to bear in mind, when defining the measurement accuracy for frequency measuring is to understand its operating environment which is at the field, not at the laboratory. Thus, it is not necessary to aim to an accuracy of high-tech oscilloscope designing to be used in a laboratory for research purposes.

4.2.3 Future and Improvement Ideas

Experts were asked to evaluate the future applications for signal generation, frequency measuring, and pulse counting. The overall answer was that the applications will stay the same in the future, and the customers' request for this type of features will remain low. However, as mentioned earlier, these are standard features of MC6. Thus, it is difficult to estimate, the overall request for these features in a calibrator, as clients don't need to especially ask for them.

According to the experts, most of the customers seems to be satisfied with the capabilities of the MC6. However, they had few improvement suggestions, in addition to the possibility to generate sine wave and dual pulse. These suggestions were mostly additional features for the software. The suggestions were: possibility to visually present the measured or generated signal on the display of the calibrator, and timer option for the frequency generation, which could be used in earlier mentioned totalizer validation. Other suggestions heard from the experts or read from the feedback database were possibility to generate ramp-signal, have higher maximum amplitude in frequency generation of the ATEX version of the MC6, and have start/stop-button for the pulse generation.

4.3 Specification Suggestion

This subchapter defines the specification for the frequency and pulse generation and measuring by utilizing the findings of the pervious subchapters of the present chapter. Also, the aspect of signal generation and measuring functions being implemented as a module that is separated from other functions of the multifunctional calibrator is discussed. Moreover, in this viewpoint, these functions for signal generation, frequency measuring, and pulse counting are implemented as a separate circuit that can be added to the multifunctional calibrator only if the customer orders them. This would require the multifunctional calibrator to have a modular structure in which different functions of the calibrator are implemented as modules which are controlled by a main processor. The benefit of the modular structure is that less compromises have to be made in the component selection as components do not have to be suitable for multiple functions. Also, the complexity of the system decreases, and making changes in the system is easier as one component does not affect multiple functions of the calibrator.

Based on the comparison between calibrators on the market made in subchapter 4.1, the specification of the current solution, for signal generation, as well as for frequency measuring, and pulse counting seems to be competitive. Also, the interviewees did believe that the customers' requirements for signal generation, frequency measuring,

and pulse counting will stay similar in the future. Hence, the specification of the current solution (MC6) should act as a base for the specification of frequency and pulse generation and measuring, as it has been proven to be competitive with other multifunctional calibrators and satisfy the needs of most customers.

The selection of generatable waveforms seems to be the main difference between different multifunctional calibrators. In addition to that, most of the few customer feedbacks regarding the signal generation of the current solution were concerning the need for the sine wave generation. Therefore, it should be investigated if it is possible to add sine wave generation to the frequency generation function. The output amplitude and frequency range for the sine wave could be the same as it is for the square wave. Opposite to the sine wave generation, there did not seem to be demand for the generation of the triangular wave. To conclude, the waveform options in the signal generation function should be at least a symmetric and positive square, with an adjustable duty cycle to allow pulse generation. Additionally, a ramp and a sine wave generation should be included in the signal generation function only if it is possible to be done without effecting the capability of the square and pulse wave generation. Furthermore, there seems to be a need for dual pulse generation. One possible application for dual pulse generation is the testing of the high-security level flowmeters. In modular calibrator, the dual pulse generation can be easily implemented by integrating two modules into one calibrator.

Moreover, if signal generation and measuring functions are implemented as an optional module, and not a standard module of the calibrator, it should be noted in the expenses and the workload of the designing process of the module. Based on the interviews the signal generation, frequency measuring, and pulse counting are not the most crucial features of a multifunctional calibrator. Therefore, it could be assumed that not every customer will purchase a calibrator with this module. Hence, design and implementation process of the module should be straightforward and expenses low. Implementation process can be eased by choosing a MCU from the MCU family that is already used in the current solutions. Also, by aiming for highly integrated solution eases the hardware designing process and there are fewer component updates required to be done.

To conclude, the specification of the current solution which is the specification of MC6 is good enough for most of the customers. Hence, the future solution for frequency and pulse generation and measuring should have the same specification for the frequency and amplitude range and resolution. A possibility to generate ramping frequency, sine wavefrom, and dual pulse could be added. On the other hand, if these additional features

cannot be implemented easily and without huge additional costs, it might not be profitable to add them to the future designee of frequency and pulse generation.

5. SOLUTION SUGGESTIONS

In the present chapter, two possible solutions for frequency and pulse generation are presented. Also, the capability of solutions to be employed for pulse counting and frequency measuring is shortly evaluated. The first solution is based on the DDS technology, and the second solution is based on an MCU and involves elements that are already present in the current solution in MC6. The operation principles of both solutions are presented in the theory section of this Thesis. Both solutions are discussed in separate subchapters, which include suggestions on how the solutions could be implemented and list the main features, limiting factors, and issues of the solutions. At the end of the subchapters, the pros and cons of the possible solutions are weighted.

The statements considering the price of the solutions and the quantity of the suitable component options made in this chapter are based on components that were on the market during the writing process of this Thesis. Electronic component distributors' websites such as Digi-Key Electronics and MOUSER electronics, also, the component manufacturers' own websites were used in the search for suitable components, especially ANALOG DEVICES and STMicroelectronics were used.

5.1 Direct Digital Synthesis Based Solution

The first solution suggested is based on a DDS chip. The DDS chips themselves are not usually capable to control the output amplitude, hence, an additional system is required for that. However, before going into details of the output amplitude control, let's consider the characteristics of the output signal that can be altered by the DDS chip. These characteristics are the output frequency and resolution, as well as the output waveform. The output frequency resolution is mainly limited by the clock frequency and the width of the tuning word. Typical tuning word widths of DDS chips are 24, 28, 32, and 48 bits, and the main clock frequency is between a few MHz to GHz. From equation 4, it can be driven that in theory, the output frequency resolution of the DDS chip is the same for the whole output frequency range. The current solution does not have this ability, its frequency resolution decreases as the output frequency increases. Furthermore, by testing different clock frequencies and tuning words in equation 4, it is possible to estimate the requirements for these two parameters, to achieve the desired frequency resolution of 1 µHz, which is the highest output frequency resolution in the current solution. If the tuning word is 32, the clock frequency should be approximately 5 kHz, to obtain the wanted frequency resolution. In addition to low clock frequency being highly

impractical, it also restricts the maximum output frequency to be less than 2.5 kHz due to the Nyquist sampling theorem. Hence, the tuning word size must be increased to 48, with it the output frequency resolution can be kept at 1 µHz with clock frequencies around 400 MHz and under. This reduces the number of the DDS chip options and complicates the generation of different waveforms which is discussed later in this subchapter.

The tuning word and the clock frequency also determine the output frequency range. The maximum output frequency is half of the clock frequency. In practice, due to the required lowpass filtering at the output stage, the maximum output frequency is less than 40% of the clock frequency [17]. As the clock frequency of the DDS chip is always higher than one MHz in other words, the maximum output frequency is always more than 400 kHz, there is no problem in the clock frequency wise to reach the desired maximum output frequency of 50 kHz. However, reaching the required minimum output frequency of 0.005 Hz is more complicated. More specifically, the theoretical minimum output frequency can be obtained by setting the phase increment word to equal one, and tuning word width to be 48-bit, and the clock frequency be below 400 MHz in equation 5. With this method the minimum generatable frequency is around 1 µHz, however, this does not consider noise factors. Moreover, the actual minimum output frequency is difficult to be defined and requires testing. In addition to that, the available scientific papers handling DDS technology do not provide studies analyzing the lowest generatable frequency. The reason for the unclear minimum output frequency might be the fact that DDS technology is primarily utilized in the generation of high frequencies.

Next the possible generatable waveforms are considered. On the market there are DDS chips that are especially designed for waveform generation, such as AD9102 [49]. These chips include a patter memory to store different waveforms. Likewise, they allow the user to create and store new waveforms that can be applied to the DDS output. However, the tuning word width of these type of DDS chips is below 48-bit, hence they cannot be used in this solution. Another option is to analogically convert the sine wave into other waveforms. Most chips with 48-bit tuning word include an internal comparator, hence, by feeding the lowpass filtered sine wave output to the comparator, a square wave output can be achieved. Moreover, if the chip includes an additional DAC, the output of the additional DAC can be used to variate the reference voltage of the comparator to adjust the duty cycle of the square wave to be a pulse wave with a desired duty cycle. Furthermore, an external counter circuit could be employed to track the number of generated pulses and trigger the generation to stop after the predefined number of pulses is reached. Therefore, in the pulse generation mode the user is able to control the number and the duty cycle of the generated pulses. Also, the square wave can be

turned into a triangular wave by using an integrator. However, based on the analysis presented in the chapter 4, need for triangular wave generation seems to be almost nonexistent. On the other hand, the DDS chips often have multiple different modes for ramp signal generation which seems to have some demand from the customers.

In the designing process of the output stage of the DDS-based signal generator, three main points should be kept in mind. The first point is that the high-speed DAC of a DDS chip often has a current output, therefore, a current-to-voltage converter is required. Furthermore, often the output impedance of the DAC is quite high, hence, buffering is required to separate the DDS section from the filtering section, and to lower the output impedance of the DAC for the rest of the circuit. [50] Both of the aforementioned tasks can be done with an op-amp circuit from which there are good examples in previous designs of Beamex. The second point is that a basic DDS system does not include amplitude control or if it is included, it is not enough for this solution. As the desired amplitude range is up to 24 V_{pp} in positive waves, and 6 $V_{pp\ in}$ symmetric waves. However, the voltage compliance range of the high-speed DACs of a DDS chips are often around 1 V and only positive signals can be generated. Therefore, in addition to the amplification circuit, also a circuit that converts the positive signal from the DAC to symmetric is necessary. One option for this is to employ switched capacitor building blocks which consist of a capacitor and two switches. Moreover, two blocks are required, the first is used to capture the potential of the half cycle of the wave, and the second is to adjust the zero-crossing point of the wave by changing the ground potential against which the second capacitor is charged. Page 12 of the datasheet of switched-capacitor instrumentation building block LTC6943 provides a good example circuit of this approach [51]. Furthermore, there should be a circuit that can be used to bypass the switched-capacitor blocks, if a positive signal is desired, instead of the symmetric one. The further amplification of the output signal can be done by a circuit with a programmable gain op-amps, the first one used to compensate the voltage drops caused by the load of the circuit, and the second one to amplify the signal to the desired level. Furthermore, the output stage of the comparator which produces the square wave, also requires an external amplitude control and possibility to convert a positive signal to symmetric. Therefore, switches should be used in the output amplitude modification stage of the sine wave to allow it to be employed for modification of the square and pulse waves amplitude or separate output stage for the comparator should be build.

The third point is that the sine wave output of the DDS chip requires a lowpass filter at 40% of the clock frequency to filter out the sin(x)/x roll-off images from the output signal. The popular option for filtering sin(x)/x roll-off images is an elliptic filter as it has the

steepest transition region. Its' downside is that there is ripple at the passband, hence, some of the wanted frequencies will be attenuated. On the other hand, Butterworth, and Gaussian family filters have a monotonic response but wider transition band, therefore, the attenuation before the cutoff frequency is minimal but the transition band is much wider than in the elliptic filter. [17] However, this might not be a problem in the present case as the cutoff frequency can be set to be much lower than 40 % of the clock frequency, as the desired maximum output frequency is 50 kHz. Also, the transition band can be narrowed by increasing the order of the filter. However, this causes the size of the filter increase.

In addition to the components required at the output stage, a MCU is needed for the control of the frequency generation in the DDS chip, and for the frequency measuring and pulse counting. Hence, it can be seen that although the DDS technology might make the frequency generation easier on the software side, it most likely increases the complexity of the circuit and the price of the whole system. Also, there are some open-ended questions that need to be studied before the design process can be continued, such as; what is the actual minimum output frequency, and what is the quality of the square wave made from the sine wave by a comparator. The main benefit of the DDS approach is that high frequencies with resolution over µHz can be generated, as well as the possibility to generate sine waves. However, the price of the DDS with high output frequency resolution is over 30 Euros, and the actual customers' demand for the capability to generate sine waves is unclear.

5.2 Microcontroller based solution

The second solution suggestion is based on a microcontroller and on the current solution which uses a timer to generate the square and pulse waves. However, the solution suggestion tries to add a sine generation possibility to this system by employing a look-up table and DMA transfer. The basic operation principles of these techniques are discussed in subchapter 3.2.2.

5.2.1 Square and Pulse Wave Generation

A general-purpose timer of a MCU can be employed to generate different frequencies from one clock frequency. Basically, in this method, the frequency of the output square wave of the timer is defined by the number of input clock pulses the timer waits before it switches its output state. Therefore, by controlling the waiting time, one controls the length of the half cycle of the output square wave which essentially defines the output frequency. Section 3.2.2 already described the basics of this method, and the present

section will further investigate usability of the method for the square and pulse wave generation.

The main issue with this technique is that the output frequency resolution depends on the number of input clock pulses used to define the period of the square wave output of the timer. In other words, as the generated frequency increases, the number of clock pulses in a cycle of the output square wave declines, thus, the resolution decreases. This phenomenon can be seen in the current solution. However, the frequency resolution can be improved by increasing the clock frequency. Moreover, equation 6 can be used to calculate the clock frequency required to obtain a specific resolution at a certain output frequency range.

$$f_{clk} = \frac{1}{\frac{1}{(f_{out} - r)} - \frac{1}{f_{out}}} \tag{6}$$

in which *r* is the desired output frequency resolution as Hertz. As defined in table 1, the current solution is capable of obtaining an output frequency resolution of 0.1 Hz at an output frequency range from 5 kHz to 50 kHz. The clock frequency required for this task can be defined by setting the f_{out} in equation 6 to be 50 kHz, and the *r* to be 0.1 Hz. The value for the required clock frequency becomes 25 GHz, which cannot be generated without some type of a phase-locked loop to amplify the original clock frequency. In addition to increasing the cost of the system, a clock frequency around the GHz range would make it impossible to generate low frequencies due to the limited size of the registers of the timer.

On the other hand, the aforementioned issue is present regardless of the clock frequency. As discussed in section 3.2.2, the size of the registers of the timer, in other words, the timer size defines the maximum number of input clock pulses the timer can wait before overflowing. Therefore, the output frequency range which the timer can generate from the clock frequency is the clock frequency divided by the $2^{\text{size of the timer}}$ to half of the clock frequency. For example, to generate the minimum required output frequency of 0.0005 Hz, with a 16-bit timer, the clock frequency cannot be higher than 32.8 Hz, which would mean that the maximum output frequency would only be 16.4 Hz. However, this problem can be overcome by using multiple timer overflows to define the output frequencies. In practice, this can be done by storing multiple divider values in a table. The term divider is used in this Thesis to describe the value that defines the length of the generated half-cycles as clock pulses. Value for the divider can be calculated by dividing the input clock frequency of the timer by the sampling frequency, which is in the case of square and pulse waves, desired output frequency multiplied by two. If the value

of the divider is larger than the maximum value of the reload register of the timer, the divider value can be implemented over a sequence of divider values. Moreover, the last bit of the divider can be used as a toggle bit which indicates if the timer should change its state as it overflows. In addition to improving the output frequency range, by defining the output frequency with multiple dividers increases, this technique can be used to improve the output frequency resolution. In most frequency measurement devices, the frequency is measured over multiple cycles and the measurement result is the average over the measured cycles. By doing so the resolution of the frequency measurement can be improved. This can be utilized in the frequency generation as well, by generating a sequence of multiple waves consisting of two different frequencies which average into the desired frequency. In practice, this can be done by calculating a sequence that consists of two dividers differencing from each other by a clock pulse.

The first step in the calculation process of the sequence of dividers that produces the desired output frequency is to define the maximum number of periods that the sequence can consist of. The maximum length depends on the gate time of the measuring device, as if the length of the sequence is more than the gate time, the measured frequency will not be the same as the generated, because the measuring device averages over fewer cycles than is used in the generation. The length of the gate time depends on the measurement device and measurement range. Moreover, the maximum sequence length as half-cycles can be obtained by multiplying the gate time with the sampling frequency. However, it could be assumed that it is not desirable to measure frequencies close to 1 Hz over multiple cycles, as the measuring time becomes longer than a second. Moreover, the frequency resolution of these low frequencies is good enough without this method because over million clock pulses is used to define the length of the half-cycles. In the current solution, the gate time utilized in the calculation of sequence length is 200 milliseconds (ms), hence, only frequencies above 5 Hz can be generated over multiple periods. Also, to reduce the required memory, the maximum length for the sequency is limited to 500 cycles. With the above limits, a completable frequency resolution is achieved. [20], [52] As there has not been any demand to widen the frequency range or increase the resolution, the 200 ms and 500 cycles could be used in the calculations done later in this section. Another option could be to give the user the possibility to edit the maximum sequence length, thus, the user could adjust it to be suitable for their application.

The second step is to calculate the dividers, which produces the suitable half-cycle lengths. The divider can be obtained by dividing the clock frequency by the sampling frequency. If the resulting divider is not an integer, two different dividers should define;

one which is equals to the whole part of the non-integer divider, and another which is one larger than the first divider. The ratio of the dividers in the sequence can be counted from the decimals of the non-integer divider as it determines the percentage of larger divider in the sequency. After the sequence of the dividers is formed, they are fed to the timer one by one. Table 3 presents the output frequency resolution improvement that can be achieved by utilizing this method.

Table 3. *A comparison between output frequency resolution generated with one divider value and two divider values. A 10 MHz clock frequency and 16-bit timer were used in the calculations and the limits for the gate time and maximum length of sequence were equal to the limits in the current solution*

Output frequency [Hz]	Resolution with one divider value [Hz]	Resolution with two different divider values [Hz]
0.0005	$2.5*10^{-14}$	$2.5*10^{-14}$
0.5	$2.5*10^{-8}$	$2.5*10^{-8}$
5	$2.5*10^{-6}$	$2.5*10^{-6}$
50	$2.5*10^{-4}$	$2.5*10^{-5}$
500	$2.5*10^{-2}$	$2.5*10^{-4}$
5000	2.5	$0.5*10^{-2}$
50000	251.3	0.5

In Table 3, the output frequency resolution values in the case which above described methos is not employed are calculated by following equation

$$Resolution = \frac{f_{clk}}{(divider*2-1)} - \frac{f_{clk}}{divider*2} \quad (7)$$

and in case which it is employed

$$Resolution = \frac{f_{clk}}{\left(\frac{divider*2*n_c-1}{n_c}\right)} - \frac{f_{clk}}{divider*2} \quad [20](8)$$

in which the n_c is the number of periods in a sequence, and the divider is the average of the dividers in the sequence. Note that the divider is multiplied by two and n_c is the number of cycles instead of half-cycles because the resolution is calculated for the output frequency, not for the sampling frequency. A code employed to calculate the frequency resolutions with the above-described method can be found in Appendix A. The code was created in PyCharm which is a Python-based development environment. The gate time and maximum sequence length were the same as in the current solution, and the used clock frequency was 10 MHz.

When choosing the suitable clock frequency and timer size for this method few factors must be considered. As already mentioned, the size of the divider depends on the size of the registers of the timer. Increasing the timer size will increase the required memory to store the divider sequency as the size of individual divider increases. On the other hand, with larger timer, the length of the divider sequency utilized to generate the low output frequency over multiple clock overflows shortens. Table 4 illustrates the relationship between the timer size and memory required to generate the minimum and maximum frequencies of the current solution.

Table 4. *The relationship between the required memory and timer size. Calculations were made with 10 MHz clock frequency, and the limits for the gate time and maximum length of sequence were equal to the limits in the current solution*

Timer size [bit]	Output frequency [Hz]	Required memory [byte]
16	0.0005	1220704
	50000	2000
24	0.0005	7155
	50000	3000
32	0.0005	40
	50000	4000

Based on table 4, the largest timer should be chosen as it requires the least amount of memory to generate frequencies from 0.0005 to 50 000 Hz.

The employed clock frequency is often a tradeoff between the required memory and the output frequency resolution. Moreover, with a higher clock frequency, a higher output frequency resolution can be obtained with a shorter divider sequence. On the other hand, with the higher clock frequency, generation of frequencies around 1 Hz requires longer divider sequencies as more clock overflow are required to produce a half-cycle of the output square wave. Based on table 4, it can be assumed that the generation of frequencies below and around 1 Hz requires the most amount of memory. Thus, it might be better to set the clock frequency quite low and use longer sequences to generate the high frequencies. Table 5 illustrates, the relationship between the required memory and clock frequency to generate 0.0005 Hz and 50 kHz square wave with at least 0.1 Hz output frequency resolution.

Table 5. *The relationship between the required memory and the clock frequency to generate 0.0005 Hz and 50 kHz square wave with at least 0.1 Hz frequency resolution. The timer size used in the calculations is 32-bit.*

Clock frequency [MHz]	Required memory to generate square wave at 0.0005 Hz [Byte]	Required memory to generate square wave at 50 kHz [Byte]
10	40	14000
20	76	7200
30	112	4800
40	152	3600
50	188	2800
60	224	2400
70	264	2000
80	300	2000
90	336	1600
100	376	1600

Based on table 5 it can be said that after 50 MHz, the decreasing effect that the increasing clock frequency has on the required memory to store the divider sequence is not that significance as before 50 MHz. By utilizing the above done analyses, it seems that the least amount of memory is required to achieve the specification of current solution with a 32-bit timer and 50 MHz clock frequency. Table 6 represents the outcome of frequency generation with this setup.

Table 6. *The outcomes of the square wave generation with a 32-bit timer and 50 MHz clock frequency. The code in appendix A was use in the calculations.*

Desired output frequency [Hz]	Actual output frequency [Hz]	Cycles	Resolution [Hz]	Required memory [Byte]
0.0005	0.0005	1	$5.0*10^{-15}$	192
0.5	0.5	1	$5.0*10^{-9}$	8
5	5	1	$5.0*10^{-7}$	8
50	50	10	$5.0*10^{-6}$	80
500	500	100	$5.0*10^{-5}$	800
5000	5000	450	$1.1*10^{-3}$	3600
50000	50000	450	0.1	3600

The pulse generation can be done with a similar logic to square wave generation; however, the length of the half cycles should be adjusted according to the desired duty cycle. Also, this technique does not define the amplitude of the square and pulse waves. The output amplitude could be controlled by controlling the reference voltage of the DAC. In addition to that, the amplitude can be further adjusted by the additional circuit described in the first solution suggestion. Furthermore, to generate ramp signals there should be twice as much memory as mentioned above because two different divider sequences must be stored, one which is currently generated and the second which is calculated while the first is generated to produce the next frequency.

5.2.2 Sine Wave Generation

In this section, a sine wave generation method that combines the technique described in subchapter 3.2 with the above-presented square wave generation method is presented. The technique described in the above sub subchapter cannot be directly employed in the sine wave generation because in sine wave generation more than two sampling points per cycle are required to generate the pattern of the waveform accurately. However, the above-described square wave generation technique can be used to generate the sampling frequency of the desired sine wave. Moreover, the timer output can be utilized as a DMA request which triggers a transfer of a sampling point of the sine wave from the LUT to the DAC. Subchapter 3.2 already described this technique in which a cycle of the sine wave is presented as a look-up table whose values are transferred to the DAC by utilizing DMA.

5.2.2.1 System Describtion

Figure 10 is a simple block diagram of the system. From the Figure, it can be seen that two tables must be calculated, before the generation; one which defines the intervals of the DMA request, in other words the sampling frequency, and another one which is a LUT of sine wave. These tables are calculated based on the desired output frequency and the used lowpass filter which can be either filter 1 or 2. Properties of the filters are discussed later in this sub subchapter. First, the number of sampling points in the LUT must be defined. The initial number of sampling points is set to be 10, however, the number is increased until the location of the first sin(x)/x-images of the generated frequency is at the stopband of employed filter. In other words, the sampling frequency is increased by increasing the number of samples in a period of sine wave. After the number of sampling points is adjusted, the sampling frequency can be calculated. Then similar to the square wave generation technique described in the above sub subchapter the sequence of half-cycles is calculated and stored to the interval table to be used to generate the sampling frequency of the sine wave.

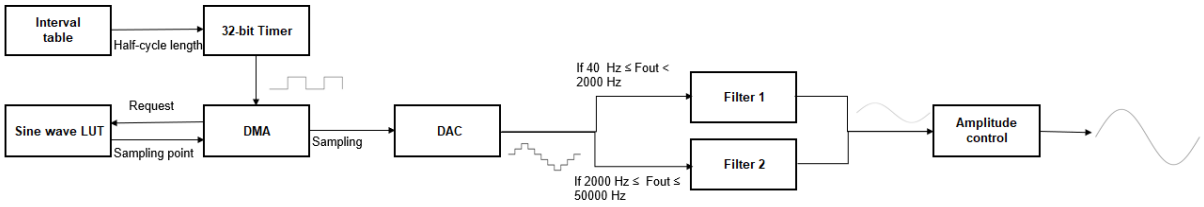

Figure 10. *A simplified block diagram of the system. F_{out} in the diagram is the desired output frequency.*

After both of the tables are defined, the sine wave generation can begin, in which the half-cycle lengths of the square wave sequence stored in the interval table are fed to the timer to control the sampling frequency. When the output of the timer is high, DMA transfers the following sampling point from the LUT to the DAC. Simultaneously, the lowpass filter smooths out the generated sine wave, which amplitude is then adjusted by the amplitude control circuit, which could be similar to the circuit described in subchapter 5.1.

5.2.2.2 Lowpass Filters

To design the lowpass filters 1 and 2, an LT spice simulation was built to simulate the filtering of the DAC output. In addition to that, Analog filter wizard, which is an analog filter designing tool provided by ANALOG DEVICES was employed in the design process of the filter circuits. Four design rules were set for the designed lowpass filters. First, the number of op-amps in the filter circuits must be two or less, thus, the order of filters cannot be more than 4. Second, there should not be more than three filter options. These

two rules were set to keep the size of the system small and price of the system low. The third rule was that the sampling frequency cannot be more than 1 MHz, this condition was set to not exceed the speed limit of common high-speed DACs, which is often a few mega samples per second (Msps). The fourth rule was that at least 10 sampling points per cycle, this was set to obtain the rule 3 and to save memory.

In order to achieve the rules one and two, the minimum output frequency had to be increased from 0.0005 to 40 Hz. The new minimum frequency was chose based on the minimum output frequency for sine wave generation in older solution. With this edition it was possible to filter whole output frequency range with 2 different filters. For output frequencies from 40 Hz to 2 kHz filter 1 is employed, and for frequencies from 2 kHz to 50 kHz filer 2 is applied. The characteristic of the filters and frequencies filtered by them are listed in Table 7.

Table 7. *Properties of designed lowpass filters.*

	Output frequency [Hz]	Passband [Hz]	Stopband (-30 dB) [Hz]	Filter
Filter 1	40…2000	0…2500	6000	4th order Butterworth
Filter 2	2000…50000	0…80000	190000	4th order Butterworth

Figures 11 to 14 are results of the simulations of the filters, the blue signal is the output of the DAC and the green is the filtered signal. Figures 11 and 13 illustrates filtering of the lowest frequency which filters are able to filter successfully, and Figures 12 and 14 the highest frequency that can be filtered without strong attenuation of the signal itself. From the Figures, it can be seen, that both of the filters induce some phase delay on the filtered signal, however, it is not a problem as the initial frequency of the signal is not affected. In addition to filtering, the number of sampling points in a period in different cases can be seen from the Figures.

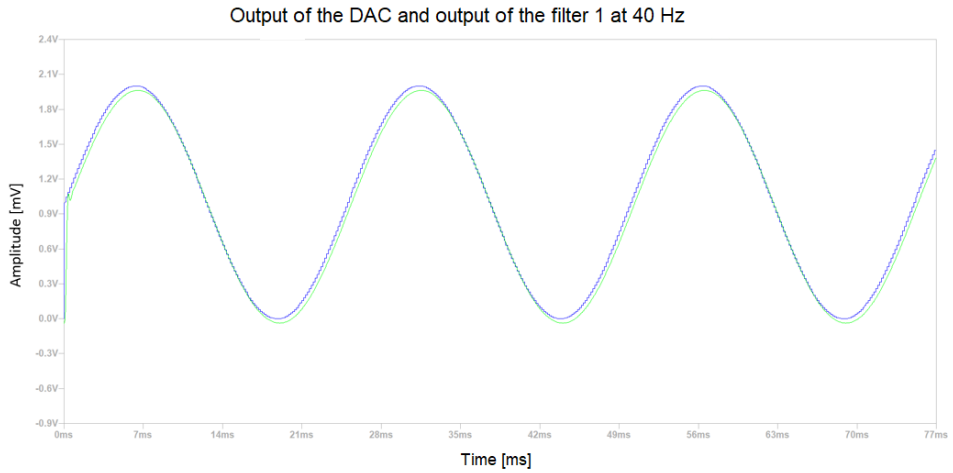

Figure 11. *The blue signal is ouput of the DAC, and the green signal is the output of the filter 1 at 40 Hz.*

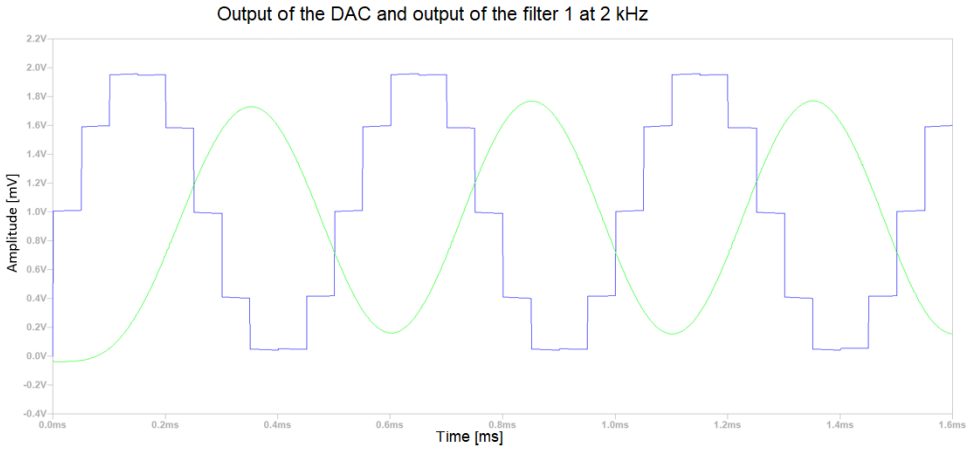

Figure 12. *The blue signals is ouput of the DAC, and the green signal is the output of the filter 1 at 2 kHz.*

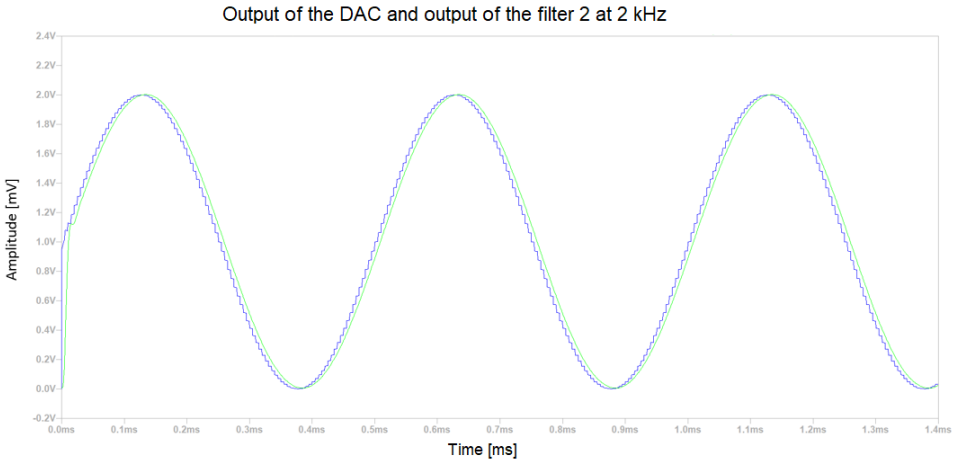

Figure 13. *The blue signals is ouput of the DAC, and the green signal is the output of the filter 2 at 2 kHz.*

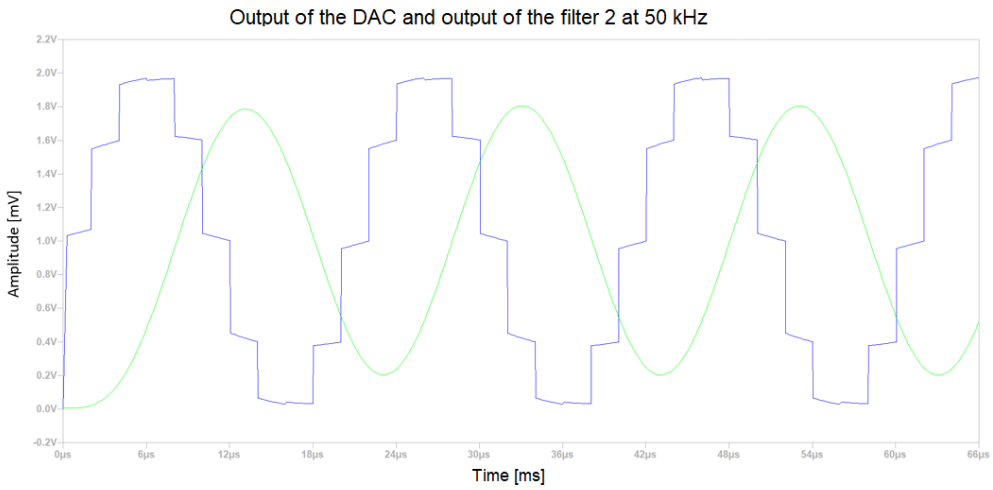

Figure 14. *The blue signals is ouput of the DAC and the green signal is the output of the filter 2 at 50 kHz.*

It must be noted that the filters 1 and 2 are just example filters, which aim is to show that the whole output frequency range can be filtered successfully with only two low order lowpass filters. LT spice simulations for filters can be found in Appendix B and C.

5.2.2.3 Spesification For Sine Wave Genration

Assuming that the same 32-bit timer and register which are used in the square and pulse wave generation can be used in the generation of sampling frequency of sine waves. Therefore, the maximum length for the interval table is 450 half-cycles, which requires around 3600 bytes of memory. The size of the sine wave LUT depends on the sampling points required to achieve the sampling frequency which depends on the cutoff frequency of the employed lowpass filter. More specifically, with filters 1 and 2 the maximum number of sampling points is 151. Furthermore, assuming that the same 32-bit DAC which is used in square wave generation is used in sine wave generation, the maximum size of the LUT becomes 604 bytes. Moreover, if the same 50 MHz clock frequency is used it is possible to achieve following resolution in sine wave generation at output frequency range from 40 Hz to 50 kHz.

Table 8. *The outcomes of sine wave generation. Code in appendix D was used to calculate the resolution and actual output frequency*

Desired output frequency [Hz]	Actual output frequency [Hz]	Resolution [Hz]
40	40.0001	$4.0*10^{-6}$
500	499.9999	$5.0*10^{-5}$
5000	5000.005	$5.0*10^{-4}$
50000	50000	$5.0*10^{-4}$

The resolution in table 8 is calculate by using equation 8, but the number 2 in the formula is replaced by the number of sampling points in one cycle of sine wave. The code used to calculate the values in table 8 can be found in appendix D.

5.2.3 Requirements For the System

In this section, the overall requirements for the microcontroller-based system are listed and discussed. In sub subchapter 5.2.1 it was calculated that the most optimal memory usage is reached with a 32-bit timer and 50 MHz clock frequency are used. Furthermore, the employed DAC must be 32-bit and as defined in sub subchapter 5.2.2, and the DAC speed must be at least 1 Msps. However, the integrated DACs of microcontrollers do not fulfill these requirements, hence an external DAC is required. Moreover, if the internal DAC of the MCU would be used, the size of the DAC should be decreased to 12-bit, as it seems that there are no microcontrollers available which have over 12-bit DAC, and DMA. In addition to requiring an external DAC, the high-speed DACs have current output, therefore the current-to-voltage circuit used in the DDS solution should be utilized also included into the microcontroller-based solution. Furthermore, an amplitude control circuit similar circuit in the DDS solution is most likely required, in order to archive the whole output amplitude range. Additionally, the amplitude control circuit is required to fix the attenuation of the signal induced by the filtering.

At least 8.2 Kbyte of read-write memory that can be accessed by the DMA channel is required if the other specifications of the system are as mentioned-above. 8.2 Kbyte includes memory require to store two sine look-up tables with the maximum amount of sampling points, and two square wave half-cycle sequence tables with the maximum number of half-cycles. The reason for double tables is the generation of ramp signal in which the tables for next output frequency increment must be calculate while simultaneously generating the previous frequency.

5.3 Comparison Between the Solutions

In the comparison of the DDS-based and MCU-based solutions to each other, the following aspects are considered; complexity and price of the solution, the amount of pre-studying needed, and how the workload of the developing process is divided between hardware and software.

The amount of pre-study required by the solutions could be evaluated by listing the unknown factors of the solutions. In the DDS-based solution the unknown factors are the quality of the square and pulse wave signals generated from the sine wave by a comparator, and if the minimum output frequency can be actually obtained. On the other hand, if these unknown factors present to be insuperable, there is an option to utilize the MCU that is used to control the DDS chip and measure frequency to also generate the square wave and pulses. However, in that case, it might not be profitable to execute the DDS solution as the DDS chip is the most expensive component in the solution and the actual customers' need for sine wave generation is not well known. The pulse and square wave generation method used in the MCU-based solution is already used in the current solution, hence, the unknown factors in the second solution only concern the sine wave generation. The main unknown factor in the sine wave generation of the second solution is the unknown distortion in the generated sine wave. Moreover, without testing, it is difficult to evaluate the effect of different unwanted delays occurring in the system on the output waveform.

In this comparison, the complexity of the solutions is evaluated by the number of different components the solution requires and difficulty of developing the embedded software for the solutions. By considering the software side first, in the MCU-based solution, the square wave and pulse wave generation code can be done by using the code of the current solution as a base which eases the workload. However, the sine wave generation have to be started from the scratch, same case is with the code for the DDS-based solutions. On the other hand, in the DDS-based solution, the only purpose of the DDS chip is to generate signals at certain frequencies, unlike in the MCU-based solution. Hence, it might be more straightforward to develop the software for the DDS-based solution. From the hardware point of view, the complexity of both systems is quite similar. The output stage of the solutions requires a current-to-voltage converter and amplitude control circuit. However, the DDS-based solution requires only one lowpass filter, but the MCU-based at least two. In addition to that, the order of the filters in the MCU-based solution must be higher than in the DDS-based because the transition band in the MCU-based solution must be steeper. Furthermore, the DDS-based solution does not require external DAC, unlike the MCU-based solution.

In this comparison the price of systems is solely judged by the price of the components. In the MCU-based solution, the most expensive components are the MCU itself, the DAC and the op-amps required in the filtering. In the DDS-based solution, the DDS chip, and the MCU used to control it are the most expensive components. Other output stage components, such as the op-amps for the amplitude control are not taken into account as they are the same in both solutions. The price of the DDS chip is most likely much higher than the price of other components in both systems. However, when the price of the op-amps in the lowpass filter and the DAC in the MCU-based system are taken into the account the price difference between solutions is most likely quite small. This can be seen by comparing tables 9 and 10 to each other.

Table 9. *Approximation of the price of the most expensive components of the DDS-based solution.*

Component	Example of suitable component(s)	Price of the example component(s)[Euro]
DDS Chip	AD9854ASTZ	48.78
MCU	MKV30F64VLF10R	3.90

Table 10. *Approximation of the price of the most expensive components of the MCU-based solution.*

Component	Example of suitable component(s)	Price of the example component(s)[Euro]
MCU	LPC5502JBD64E	5.32
DAC	DAC8811ICDGKT	20.45
Filter 1	Active 4th order Butterworth filter: AD8657ARMZ-R7 ,4 x capacitors, and 4 x resistors	~7.3
Filter 2	Active 4th order Butterworth filter: 2 x OP184FSZ-REEL7, 4 x capacitors, and 4 x resistors	~12.8

The total price for the most expensive components in the DDS-based solution is approximately 53 Euros and for the MCU-based solution 46 Euros. In the tables the price of resistors and inductors is assumed to be around 0.1 Euros and the price of the capacitors 0.9 Euros. These assumptions were made by shortly reviewing the prices of basic surface-mount device (SMD) resistors, capacitors, and inductors on the market.

6. CONCLUSIONS

The aim of the Thesis was to investigate the improvements which can be made in the frequency and pulse generation functions of the next-generation multifunctional calibrator and to suggest possible designs that could be utilized to achieve the found improvements. The investigation for improvements was done by reviewing multifunctional calibrators on the market, interviewing experts to produce a customer needs analysis, and reading customers feedback published in the quality feedback database of Beamex. In the investigation of the improvements, it was found that the specification for frequency and pulse generation is quite similar in the calibrators of different manufacturers. Moreover, the specification of the pulse and frequency generation in the current solution by Beamex has a competitive output frequency and amplitude range, and resolution. However, at least two out of the four reviewed calibrators had a possibility to generate sine waves which is not possible in the current solution by Beamex. Also, the demand for sine wave generation was raised in the customer needs analysis, however, interviewees did not evaluate it to be significant. The second improvement point found was the capability of dual pulse generation, which was not an option in any of the four reviewed calibrators. One application, in which a dual pulse is required, is in the validation of the high-security level flow meters which have dual transmission lines. Moreover, the validation of flowmeters seems to be the main application for both frequency and pulse generation. Based on the interviews, it could be concluded that the frequency and pulse generation are not the most used functions in a multifunctional calibrator and most customers are satisfied with the capabilities of the current solution. Hence, the improvements, which are the sine wave, and the dual pulse generation should be only added if they can be implemented with low cost and effort.

The next step in this Thesis was to investigate how to implement the improvements. With a modular structure the dual pulse can be easily added to the frequency and pulse generation function by adding another signal generator module to the calibrator. However, to generate sine waves, larger changes in the current solution must be made. The first solution suggestion presented in this Thesis was to implement the frequency and pulse generation with a DDS chip, which would allow the generation of quite pure sine waves, requiring only one lowpass filter at the output stage. However, due to the high output frequency resolution specification, only the DDS chips with a tuning word width of 48-bits could be used in this solution. The DDS chips with 48-bit tuning word currently on the market do not have an option to digitally change the output waveform.

Therefore, the square and pulse wave should be generated from the sine wave by employing a comparator. The quality of the square wave made by this way must be further investigated. Another option could be to add a DDS chip to the circuit only if the clients request a sine wave generation capability, and to generate the square and pulse waves by utilizing the method that is used in the current solution.

The second solution suggestion presented in this Thesis was to employ a timer of an MCU to generate square and pulse waves. Moreover, the input of the timer is the clock frequency which is delayed in the timer accordingly to output a square or a pulse wave at the desired frequency. The amount of delay induced by the timer is defined by calculating the number of clock pulses a half-cycle of the generated frequency consists of. Furthermore, to improve the resolution of the frequency generation jittering can be induced to generation by generating a sequence of square waves which length is close to the gate time of the receiving device. Furthermore, this sequence consists of two different half-cycle lengths which differ from each other by one clock pulse. The ratio of the half-cycles in the sequence is calculated in a such a way that the average over the sequence has the desired resolution. In this solution, the sine wave generation utilizes the output signal of the timer to control the speed of the transfer of sampling points of a sine wave from a LUT to a DAC. In order to decrease the contribution of the CPU in the transfer process, a DMA can be employed. The benefit of the second solution is that the timer-based square and pulse generation method is already used in the current solution, hence its capabilities are well-known. Also, in the second method sine, square, and pulse waves can be all done with a single MCU. The main drawback of the second solution is the complex output stage which includes a DAC, at least two active lowpass filters, a current-to-voltage converter circuit, and an amplitude control circuit. The external current steering DAC with 16-bits or over is required to achieve a proper output amplitude resolution and fast digital-to-analog conversions in the generation of sine waves. Due to the properties of the DAC, the current-to-voltage converter circuit is needed. Furthermore, the amplitude control circuit is required to achieve the large output amplitude range. Both of the aforementioned circuits are also needed in the first solution. From both solution suggestions, it can be seen that the addition of a sine wave generation would require pre-studying, testing, and additional components that may lead to a more complex and expensive system.

REFERENCES

[1] "What is calibration?" *beamex.com*. https://www.beamex.com/resources/what-is-calibration/ (accessed Oct. 04, 2021).

[2] M. Cable, L. Diagram, I. S. Sheet, P. Specifications, and F. I. Calibration, "Calibration principles," in *Calibration: A technician' s Guide*, pp. 1–10, 2005.

[3] T. Kuphaldt, *Lessons In Industrial Instrumentation*. 2019. [Online]. Available: https://control.com/uploads/Lessons-In-Industrial-Instrumentation-Control.pdf

[4] H. Laurila, M. Cable, A. Brunelli, and N. Espy, *Calibration Essentials*. Beamex Oy Ab, International Society of Automation (ISA). [Online]. Available: https://resources.beamex.com/calibration-essentials-e-book?_gl=1*1rlyvto*_ga*MTMwMDc0NDI0OC4xNjI5MDQxNTQ4*_ga_DFQ48QBW49*MTYzMzMzNTExMC44LjEuMTYzMzMzNTEyMy4w.

[5] U. Tietze, C. Schenk, and E. Gamm, *Electronic Circuits*. Springer Berlin Heidelberg, 2008. doi: 10.1007/978-3-540-78655-9.

[6] D. I. Crecraft and S. Gergely, "Signal generation," in *Analog Electronics*, 2002, pp. 263–281. doi: 10.1016/b978-075065095-3/50011-8.

[7] R. Senani, D. R. Bhaskar, V. K. Singh, and V. K. Singh, *Sinusoidal oscillators and waveform generators using modern electronic circuit building blocks*. 2015. doi: 10.1007/978-3-319-23712-1.

[8] "Generating a Signal: Types of Function Generators, DAC Considerations, and Other Common Terminology Overview," 2019. [Online]. Available: https://www.ni.com/fi-fi/innovations/white-papers/15/generating-a-signal--types-of-function-generators--dac-considera.html

[9] L. Wang, W. Chen, K. Chen, R. He, and W. Zhou, "The Research on the Signal Generation Method and Digital Pre-Processing Based on Time-Interleaved Digital-to-Analog Converter for Analog-to-Digital Converter Testing," in *Applied sciences*, vol. 12, no. 1704, p. 1704, 2022, doi: 10.3390/app12031704.

[10] L. Wang and Y. Zhao, "Signal generation techniques based on arbitrary sample rate conversion," in *Proceedings - 2014 IEEE International Conference on Computer and Information Technology, CIT 2014*, Dec. 2014, pp. 446–449. doi: 10.1109/CIT.2014.121.

[11] A. Franck, "Arbitrary sample rate conversion with resampling filters optimized for combination with oversampling," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 2011, pp. 149–152. doi: 10.1109/ASPAA.2011.6082271.

[12] "A Technical Tutorial on Digital Signal Synthesis," 1999.

[13] H. Hsu, *Introduction to mixed-signal ic test and measurement [Book Reviews]*, vol. 17, no. 4. 2005. doi: 10.1109/mcd.2001.950062.

[14] H. Malloug, M. J. Barragan, and S. Mir, "Practical Harmonic Cancellation Techniques for the On-Chip Implementation of Sinusoidal Signal Generators for Mixed-Signal BIST Applications," in *Journal of Electronic Testing: Theory and Applications (JETTA)*, vol. 34, no. 3, pp. 263–279, Jun. 2018, doi: 10.1007/s10836-018-5720-2.

[15] A. Mamgain, M. J. Barragan, and S. Mir, "Analysis and mitigation of timing inaccuracies in high-frequency on-chip sinusoidal signal generators based on harmonic cancellation," in *Proceedings of the European Test Workshop*, May 2021, vol. 2021-May. doi: 10.1109/ETS50041.2021.9465424.

[16] H. Malloug, M. J. Barragan, S. Mir, L. Basteres, and H. le Gall, "Design of a sinusoidal signal generator with calibrated harmonic cancellation for mixed-signal BIST in a 28 nm FDSOI technology," Jul. 2017. doi: 10.1109/ETS.2017.7968214.

[17] J. Vankka, *Direct Digital Synthesizers: Theory, Design and Applications*, Ph.D. dissertation, Dept. Elect. and Com. Eng., Tech. Univ. Helsinki, November. 2000.

[18] R. Jiang *et al.*, "Gray-code Input DAC Architecture for Clean Signal Generation," in *International Symposium on Intelligent Signal Processing and Communication Systems*, 2017, pp. 669–674.

[19] "ECE353: Introduction to Microprocessor Systems: Timers Basics," *University of Wisconsin-Madison*. https://ece353.engr.wisc.edu/timers/timers/ (accessed May 12, 2022).

[20] "TUSKAn taajuusgeneroinnin astetteluresoluutio." unpublished

[21] D. Aarno and J. Engblom, *Software and System Development Using Virtual Platforms*. San Francisco: Elsevier Science & Technology, 2014. doi: 10.1016/C2013-0-14366-8.
[22] A. Miller, “AN3312 Arbitrary Waveform Generator Using DAC and DMA Introduction,” 2019.
[23] “AN3126 Application note Audio and waveform generation using the DAC in STM32 products,” 2020. [Online]. Available: www.st.com
[24] J. Terney, C. M. Rader, and B. Gold, “A Digital Frequency Synthesizer,” in *IEEE Transactions on Audio and Electroacoustics*, vol. 19, no. 1, pp. 48–57, 1971, doi: 10.1109/TAU.1971.1162151.
[25] C. E. Calosso, Y. Gruson, and E. Rubiola, “Phase noise and amplitude noise in DDS,” in *2012 IEEE International Frequency Control Symposium, IFCS 2012, Proceedings*, 2012, pp. 772–777. doi: 10.1109/FCS.2012.6243619.
[26] H. He, J. Z. Zhang, and Z. H. Zhang, “Design and implementation of signal generator based on DDS,” in *Applied Mechanics and Materials*, 2013, vol. 380–384, pp. 3643–3647. doi: 10.4028/www.scientific.net/AMM.380-384.3643.
[27] J. Qi, Q. Sun, X. Wu, C. Wang, and L. Chen, “Design and Analysis of a Low Cost Wave Generator Based on Direct Digital Synthesis,” in *Journal of Electrical and Computer Engineering*, vol. 2015, 2015, doi: 10.1155/2015/367302.
[28] T. S. Wei, N. Sulaiman, N. Kamsani, and N. A. M. Yunus, “Dual control Direct Digital Synthesizer (DCDDS) for electronic testing and experimental work,” in *2014 4th International Conference on Engineering Technology and Technopreneuship, ICE2T 2014*, Jan. 2015, vol. 2014-August, pp. 137–142. doi: 10.1109/ICE2T.2014.7006234.
[29] N. Ain, A. M. Nasir, M. Nazim Mohtar, N. Amziah, and M. Yunus, “Development of Signal Generator for Lab on a Chip Application; Development of Signal Generator for Lab on a Chip Application,” 2018.
[30] X. Chen and J. Chen, “Design of an arbitrary waveform signal generator,” in *Procedia Engineering*, 2011, vol. 15, pp. 2500–2504. doi: 10.1016/j.proeng.2011.08.470.
[31] H. Lei, G. Gao, and Y. Huang, “Design and Simulation of Multi-modal Signal Generator,” in *Proceedings - 2020 7th International Conference on Information Science and Control Engineering, ICISCE 2020*, Dec. 2020, pp. 2366–2369. doi: 10.1109/ICISCE50968.2020.00464.
[32] Z. Zhenyu, C. Jufang, L. Yand, W. Lianming, C. Zengyu, and Yuan Feng, “The Design and Implementation of Signal Generator Based on DDS,” in *2017 IEEE 9th International Conference on Communication Software and Networks (ICCSN)*, 2017, pp. 920–923.
[33] Z. Zhang and F. Dong, “A Harmonic Signal Generator Based on DDS and SOPC,” in *2010 Chinese Control and Decision Conference*, 2012, pp. 1542–1547.
[34] M. Suchenek and T. Starecki, “Programmable pulse generator based on programmable logic and direct digital synthesis,” *Review of Scientific Instruments*, vol. 83, no. 12, Dec. 2012, doi: 10.1063/1.4771921.
[35] A. Sharma, Y. Sun, and O. Simpson, “Design and Implementation of a Re-Configurable Versatile Direct Digital Synthesis-Based Pulse Generator,” *IEEE Trans Instrum Meas*, vol. 70, 2021, doi: 10.1109/TIM.2021.3094240.
[36] “Fluke 754 Documenting Process Calibrator-HART Key features.” Fluke Corporation. Accessed: Jan. 11, 2022. [Online]. Available: https://www.fluke.com/en-vn/product/calibration-tools/multifunction-calibrators/fluke-754
[37] “Fluke 750 Series Documenting Process Calibrators: Work smarter. Work faster.” Fluke Corporation, 2015. Accessed: Jan. 12, 2022. [Online]. Available: https://dam-assets.fluke.com/s3fs-public/1263323_6110_ENG_M_W.PDF
[38] “Additel 227, 227Ex Documenting Multifunction Process Calibrator,” *Additel*. Accessed: Jan. 12, 2022. [Online]. Available: https://www.additel.com/download/datasheet/227_Multifunction_Process_Calibrator.pdf
[39] “DPI 620 Genii Advanced modular calibrator and HART ® /Foundation Fieldbus ™ communicator.” Druke, a Barker Hughes business, Mar. 2021. Accessed: Jan. 12, 2022. [Online]. Available: https://dam.bakerhughes.com/m/2d9ad6916ff41dbe/original/Druck-DPI-620-Genii-Advanced-Modular-Calibrator-Datasheet-EN.pdf
[40] “Beamex MC6 ADVANCED FIELD CALIBRATOR AND COMMUNICATOR.” Beamex Oy Ab. Accessed: Jan. 12, 2022. [Online]. Available: https://www.beamex.com/wp-content/uploads/2020/08/Beamex-MC6-brochure-ENG.pdf
[41] C. Castillo and R. Shaw, “AN OIL FLOW MONITORING SYSTEM BASED ON WEB SERVICES,” Tampere, 2010.

[42] J. Yoder, “Custody Transfer of Oil & Gas,” *PI Process Instrumentation*, Oct. 24, 2012. https://www.piprocessinstrumentation.com/instrumentation/flow-measurement/article/15558637/custody-transfer-of-oil-gas (accessed Nov. 24, 2021).
[43] “Petroleum liquids and gases - Fidelity and security of dynamic measurement - Cabled transmission of electric and/or electronic pulsed data (ISO 6551:1982),” Oct. 1995. Accessed: Jan. 09, 2022. [Online]. Available: http://mosse2013/sites/quality_management_system/qms/PDF%20standards/SFS-EN%20ISO%206551.pdf
[44] “TB-970901C Technical Bulletin, Dual Pulse Flowmeter Pulse Fidelity Checking,” 2010. [Online]. Available: www.omniflow.com
[45] K. Autio, “MC6-Ex taajuusgenerointialuen muutos,” Aug. 2020. Unpublished.
[46] “ZERO SPEED SWITCH: PROTECT EQUIPMENT, PROCESS AND PEOPLE,” *Control Concepts USA*, Jan. 29, 2021. https://controlconceptsusa.com/zero-speed-switch-protect-equipment/ (accessed Dec. 30, 2021).
[47] C. Boubay, “Issue 4439 - Sinus signal generation.” Quality feedback database of Beamex, Feb. 12, 2013. Accessed: Jan. 10, 2022. [Online]. Available: https://japetus/qfb/show_bug.cgi?id=4439
[48] H. Laurila, “Issue 5402 - Sinus wave genration.” Quality feedback database of Beamex, Mar. 05, 2015. Accessed: Jan. 10, 2022. [Online]. Available: https://japetus/qfb/show_bug.cgi?id=5402
[49] A. Devices, “AD9102 (Rev. A).” [Online]. Available: www.analog.com
[50] A. Devices, “CN-0304.” [Online]. Available: http://www.analog.com/CN0304-DesignSupport.
[51] L. Technology Corporation, “LTC6943 - Micropower, Dual Precision Instrumentation Switched Capacitor Building Block.”
[52] M. Stenback and K. Autio, “MC6 Lämpötila- ja geneointiosan sekä sähkömittausosan arkkitehtuuri.” unbublished.

APPENDIX A: CODE FOR COUNTING THE OUTPUT FREQUENCY RESOLUTION OF THE SQUARE WAVE GENERATION

```
import numpy as np
import math as math
import xlsxwriter

fout_table = [0.0005, 0.5, 5, 50, 500, 5000, 50000] #Calcaulted output
frequencies
fclk = 10*10**6 #The clock frequency
max_divider = 2**15 #Based on the timer size and toggle bit
size = 16 #Size of the divider value as bits
results = []

for fout in fout_table:

    desired_fout = fout

    fs = 2 * fout #Sampling frequency

    #Calculate the sequence length for cycles
    gate_time = 0.2
    max_sequence = math.floor(gate_time * fout)
    if max_sequence > 500: #Maximum length of the sequence in cycles
        max_sequence = 500
        gate_time = max_sequence/fout
    elif max_sequence <= 0:
        max_sequence = 1
        gate_time = 1 / fout


    max_sequence = 2*max_sequence #Convert the sequence length from cycles
to half-cycles

    divider = fclk / fs #Count the divider

    if divider % 1 != 0:  #If the gate time is not integer, count the new
    divider by utilizing jitter

        n = len(str(max_sequence)) - 1
        frac, whole = math.modf(divider)
        y = np.round(frac, n)
        x = 1 - y
        y = max_sequence * y
        x = max_sequence * x
        divider_1 = whole
        divider_2 = whole + 1
        divider = (divider_1 * x + divider_2 * y) / (x + y)
        fout = fclk / (divider * 2)
        resolution = fclk / ((divider * 2 * (max_sequence / 2) - 1) /
(max_sequence / 2)) - fclk / (divider * 2)

    else:
        if max_sequence == 2:
            resolution = fclk / (divider * 2 - 1) - fclk / (divider*2 * 1)
```

```
            else:
                resolution = fclk / ((divider * 2 * (max_sequence / 2) - 1) /
(max_sequence / 2)) - fclk / (divider * 2)


        if divider > max_divider:  #If the divider is larger than timer can
        handle, multiple cycles must be generated
            a, b = math.modf(divider / max_divider)
            b += 1

        else:
            b = 1
        memory_interval = b * size * max_sequence / 8 #Count the memory required
    to store the sequency of dividers in bytes
        result = [desired_fout,fout, max_sequence, fs, divider, resolution,
        memory_interval]
        results.append(result)

    #Save results as an excel
    header = ['Desired Fout [Hz]'
        ,'Actual Fout [Hz]', 'Sequency length', 'Fs [Hz]', 'Divider',
        'Resolution [Hz]', 'Size of the interval table [byte]']


    workbook = xlsxwriter.Workbook('square.xlsx')
    worksheet = workbook.add_worksheet()

    worksheet.write('A1', 'Clock frequency [Hz]')
    worksheet.write('B1', fclk)
    worksheet.write('A2', 'timer size [bit]')
    worksheet.write('B2', size)

    row = 3
    column = 0

    for item in header:
        worksheet.write(row, column, item)
        column += 1

    row = 3
    column = 0

    for result in results:
        row += 1
        column = 0
        for item in result:
            worksheet.write(row, column, item)
            column += 1


    workbook.close()
```

APPENDIX B: NETLIST FOR FILTER 1 SIMULATION IN LTSPICE

```
"ExpressPCB Netlist"
"LTspice XVII"
1
0
0
""
""
""
"Part IDs Table"
"B1" "V=int(V(vs0))" ""
"V1" "SINE(1 1 {f0})" ""
"A1" "vt=0.5" ""
"V2" "PULSE(0 1 0 0 0 {1/fs/2} {1/fs})" ""
"V3" "5" ""
"V4" "-5" ""
"V5" "0" ""
"R:STAGE3:2" "383E3" ""
"R:STAGE3:1" "102E3" ""
"C:STAGE3:2" "100E-12" ""
"C:STAGE3:1" "1E-9" ""
"U:STAGE3:1" "AD8657" ""
"R:STAGE4:2" "1.13E6" ""
"R:STAGE4:1" "33.2E3" ""
"C:STAGE4:2" "100E-12" ""
"C:STAGE4:1" "1E-9" ""
"U:STAGE4:1" "AD8657" ""

"Net Names Table"
"sq" 1
"0" 2
"s0" 8
"NC_01" 10
"fs" 11
"NC_02" 13
"MP_00" 14
"MP_01" 15
"vs0" 16
"NC_03" 18
"Vp" 19
"Vn" 22
"Vref" 25
"STAGE3:N002" 28
"STAGE3:N001" 31
"OUT1" 34
"STAGE4:N002" 38
"STAGE4:N001" 41
"OUT" 44

"Net Connections Table"
1 1 1 0
2 1 2 3
2 2 2 4
2 4 2 5
```

```
2 5 2 6
2 6 2 7
2 7 2 0
3 2 1 9
3 3 1 0
4 3 2 0
5 3 3 12
5 4 1 0
6 3 4 0
7 3 5 0
8 3 6 0
9 3 7 17
9 9 2 0
10 3 8 0
11 5 1 20
11 12 3 21
11 17 3 0
12 6 1 23
12 12 4 24
12 17 4 0
13 7 1 26
13 10 2 27
13 15 2 0
14 8 1 29
14 10 1 30
14 12 1 0
15 8 2 32
15 9 1 33
15 11 2 0
16 11 1 35
16 12 2 36
16 12 5 37
16 14 2 0
17 13 1 39
17 15 1 40
17 17 1 0
18 13 2 42
18 14 1 43
18 16 2 0
19 16 1 45
19 17 2 46
19 17 5 0
```

APPENDIX C: NETLIST FOR FILTER 2 SIMULATION IN LTSPICE

```
"ExpressPCB Netlist"
"LTspice XVII"
1
0
0
""
""
""
"Part IDs Table"
"B1" "V=int(V(vs0))" ""
"V1" "SINE(1 1 {f0})" ""
"A1" "vt=0.5" ""
"V2" "PULSE(0 1 0 0 0 {1/fs/2} {1/fs})" ""
"V6" "5" ""
"V7" "-5" ""
"V8" "0" ""
"R:STAGE5:2" "60.4E3" ""
"R:STAGE5:1" "15.8E3" ""
"C:STAGE5:2" "20E-12" ""
"C:STAGE5:1" "200E-12" ""
"U:STAGE5:1" "OP284" ""
"R:STAGE6:2" "178E3" ""
"R:STAGE6:1" "5.49E3" ""
"C:STAGE6:2" "20E-12" ""
"C:STAGE6:1" "200E-12" ""
"U:STAGE6:1" "LT6015" ""

"Net Names Table"
"sq" 1
"0" 2
"s0" 8
"NC_01" 10
"fs" 11
"NC_02" 13
"MP_00" 14
"MP_01" 15
"vs0" 16
"NC_03" 18
"Vp" 19
"Vn" 22
"Vref" 25
"STAGE5:N002" 28
"STAGE5:N001" 31
"OUT1" 34
"STAGE6:N002" 38
"STAGE6:N001" 41
"OUT" 44

"Net Connections Table"
1 1 1 0
2 1 2 3
2 2 2 4
2 4 2 5
2 5 2 6
```

```
2 6 2 7
2 7 2 0
3 2 1 9
3 3 1 0
4 3 2 0
5 3 3 12
5 4 1 0
6 3 4 0
7 3 5 0
8 3 6 0
9 3 7 17
9 9 2 0
10 3 8 0
11 5 1 20
11 12 3 21
11 17 3 0
12 6 1 23
12 12 4 24
12 17 4 0
13 7 1 26
13 10 2 27
13 15 2 0
14 8 1 29
14 10 1 30
14 12 1 0
15 8 2 32
15 9 1 33
15 11 2 0
16 11 1 35
16 12 2 36
16 12 5 37
16 14 2 0
17 13 1 39
17 15 1 40
17 17 1 0
18 13 2 42
18 14 1 43
18 16 2 0
19 16 1 45
19 17 2 46
19 17 5 0
```

APPENDIX D: CODE FOR COUNTING THE OUTPUT FREQUENCY RESOLUTION OF THE SINE WAVE GENERATION

```
import numpy as np
import tabulate as tab
import xlsxwriter


fout_table = [40,499, 500,4999, 5000, 49999, 50000]
min_ns = 10
fclk = 50*10**6 #The clock frequency
max_divider = 2**31 #Based on the timer size and toggle bit
gate_time = 0.2
results=[]

for fout in fout_table:

    desired_fout = fout

    #Calculate max sequence
    max_sequence = np.floor(gate_time*fout)

    #Count the suitable sampling frequency (fs) and number of sampling points
    based on the filter requirements
    fs = fout*min_ns
    i = 0

    #Filter 1 requirement
    if fout < 2000 and fs-fout < 6000:
        while fs-fout < 6000:
            i += 1
            fs = fout*(min_ns +i)
            ns = min_ns + i

    #Filter 2 requirements
    elif fout >= 2000 and fs-fout < 200000:
        while fs-fout < 200000:
            fs = fout*(min_ns +i)
            i += 1
            ns = min_ns + i
    else:
        ns = min_ns

    #Count the divider
    divider = fclk/fout
    divider = divider/ns

    if max_divider < divider:
        print('error divider too big')
    else:
        if divider%1 != 0:
            frac, whole = np.modf(divider)
            y = round(max_sequence * frac)
            x = max_sequence - y
            divider_1 = whole
```

```
                divider_2 = whole + 1
                cycle_1 = ns * divider_1
                cycle_2 = (ns - 1) * divider_1 + divider_2
                divider = (divider_1 * x + divider_2 * y) / (x + y)
                fout = fclk / (divider * ns)

                resolution = fclk / ((divider * ns * max_sequence - 1) /
   max_sequence) - fclk / (divider * ns)

            else:
                resolution = fclk / ((divider * max_sequence*ns - 1) /
max_sequence) - fclk / (divider*ns)

        memory_interval = 32*ns*max_sequence/8/1024 #Memmory required to store
        the intervals

        memory_wavefrom = 32*ns/8 # Memory required for the LUT of the waveform

        result = [desired_fout,fout, max_sequence,ns, fs, divider, resolution,
        memory_interval,memory_wavefrom, gate_time]
        results.append(result)


   header = ['desired fout [Hz]' ,'Actual fout [Hz]', 'max sequency
   length','number of sampling points', 'fs [Hz]', 'divider', 'resolution
   [Hz]', 'size of the interval table [kbyte]', 'size of waveform table [byte]',
   'gatetime']


   workbook = xlsxwriter.Workbook('short_sine.xlsx')
   worksheet = workbook.add_worksheet()

   worksheet.write('A1', 'Clock frequency [Hz]')
   worksheet.write('B1', fclk)

   row = 3
   column = 0

   for item in header:
       worksheet.write(row, column, item)
       column += 1

   row = 3
   column = 0

   for result in results:
       row += 1
       column = 0
       for item in result:
           worksheet.write(row, column, item)
           column += 1


   workbook.close()
```