Postprint of: Czaja Z., Measurement method for capacitive sensors for microcontrollers based on a phase shifter, MEASUREMENT Vol. 192 (2022), 110890, DOI: 10.1016/j.measurement.2022.110890

Measurement method for capacitive sensors for microcontrollers based on a phase shifter

Abstract

A complete measurement method dedicated to capacitive sensors has been developed. It includes the development of hardware (an analogue interface circuit for microcontrollers with built-in times/counters and analogue comparators) and software (a measurement procedure and a systematic error calibration (correction) algorithm which is based on a calibration dictionary). The interface circuit consists of a low-pass filter and a phase shifter with a capacitive sensor. A prototype circuit based on a mid-range 8-bit microcontroller has been developed and investigated. We also analysed the relative inaccuracy of a measured capacitance of the sensor and performed experimental research. We obtained the relative errors of capacitance determination < 0.06%, which gives a measurement accuracy < 72 fF for the assumed range of the capacitance (100 pF – 300 pF).

Keywords: sensor electronic interface; microcontroller; capacitive sensor; phase shifter

1. Introduction

An important group of capacitive sensors are sensors, which convert a given physical or chemical quantity into capacitance. Hence we distinguish, among others, sensors that are used for measurements of humidity [1–5], pressure [6–8], position [9–14], surface-force [15,16], carbon dioxide concentrations [17,18], flow of fluid and biomass flour materials [19,20], liquid level [21,22], and also oil pollution by water and soil moisture [21,23–25]. In the next step, based on the analogue interface circuits, we have to transform the value of the capacitance of the sensor to a measurable electrical quantity, e.g. voltage, duration, period or frequency of an electrical signal. From the above, the interface circuits can be classified into two main categories of analogue conversion techniques and their subcategories:

- Capacitance-to-voltage (C–V):
    - based on an excitation by a sine wave (Fig. 1a) [9–11,14,15,18,19,22,25–30],
    - based on an excitation by a square wave (Fig. 1b) [31,32],
    - based on switched-capacitor C-V converters (Fig. 1c) [20,33],
    - based on a relaxation oscillator, a comparator or a diode detector, and a low-pass filter (Fig. 1d) [1,34].

- Capacitance-to-time (C–T):
  - Period Modulation (PM) techniques [1,5,16,35–41],
  - Frequency Modulation (FM) techniques [42–45],
  - RC charge/discharge techniques – e.g. based on a direct sensor-to-microcontroller interface [3,4,46–48],
  - A charge balancing method [23],
  - A method based on a phase shifter [24].

We can also add the third category of the interface circuits – based on capacitance-to-digital domain conversion techniques. In this case, the value of the capacitance of the sensor representing any physical or chemical quantity is converted into a digital code and read from the circuit (usually implemented as an integrated circuit) by a serial interface, e.g. commercial integrated circuits: MAS6513 (I^2C, SPI), AD7747 (I^2C) and integrated circuits primarily dedicated to MEMS [6,49].

As shown in Fig. 1 the interface circuits based on the C-V conversion techniques generally consist of three blocks: the block responsible for generating the excitation signal, the block that converts the capacitance to e.g. an amplitude of the signal or a PWM signal, and the last block that converts this amplitude or the PWM signal to DC voltage. The construction of the second block strictly depends on the sensor topology (floating or grounded single element sensor, differential sensor) and also on the dynamics of the changes in the quantity measured by the sensor [14,19]. Proposals of constructions of the interface circuits based on a sinusoidal excitation (Fig 1a) are included in Table 1.

Table 1. Comparison of capacitance-to-voltage analogue techniques based on a sinusoidal excitation approach from the recent literature.

Ref.	**Sensor topology**	**Circuit topology of *C to v_c* conversion blocks**	**Circuit topology of AC-DC conversion signal blocks**
[9]	d.	two frequency-independent AC bridges + differential amplifier	phase sensitive demodulation based on four-quadrant multiplier + LPF
[10]	s. f.	current feedback circuit	peak detector circuit + LPF
[11]	s. f.	capacitance (voltage) divider	LPF
[14]	s. g.	two T-networks + differential amplifier	(to fast ADC)
[15]	s. g.	serial LC oscillator	frequency demodulation circuit + LPF
[18]	s. f.	centre-tap transformer + summing amplifier	LPF

[25]	s. f.	capacitance divider with C_x - capacitance divider with C_{ref} -	full-wave rectifier +LPF - differential amplifier full-wave rectifier +LPF -
[27]	s. f.	C_x - modified gyrator circuit C_{ref} -	rectifier +LPF - summing amplifier rectifier +LPF -
[28]	s. g.	trafo-based AC bridge	phase sensitive demodulation based on multiplier + LPF
[19]	s. f.	modified De-Sauty AC bridge	(to fast ADC)
[26]	s. f.	modified De-Sauty AC bridge	rectifier + LPF
[22]	s. f.	modified De-Sauty AC bridge	rectifier + LPF
[29]	d.	modified De-Sauty AC bridge with a Voltage Controlled Resistor	mixer + LPF
[30]	s. f.	modified De-Sauty AC bridge with Voltage Controlled Resistors	mixer + LPF

Notes: d – differential, s – single, f – floating, g – grounded, LPF – low-pass filter.

The interface circuits using a square-wave excitation (Fig. 1b) are based on a relative phase delay technique [31,32]. In this case, the circuit which converts the capacitance of the sensor into a PWM signal is simple and generally consists of four inverter gates and one XOR gate. Therefore, only a low-pass filter is needed to convert the PWM signal into DC voltage.

C-V conversion based on switched-capacitor techniques (Fig. 1c) [20,33] is especially used for on-chip integrated circuits [6,35].

The last interesting proposal of the interface circuit is a solution based on a relaxation oscillator (Fig. 1d). Instead of a square wave, a triangle wave with an amplitude that depends on the sensor capacity is used. Therefore, a comparator with a fixed voltage reference [1] or a diode rectifier [34] is enough to obtain a PWM signal which is converted into a DC voltage by a low-pass filter.

The second group of interface circuits based on the C-T techniques includes circuits that are structurally simpler than the presented C-V group. For example, most circuits based on the PM technique are relaxation oscillators that generate a square wave with a period T depending on the value of the sensor capacitance. Papers [1,35,37] propose interface circuits for single floating sensors, papers [5,36,38,40,41] for single grounded sensors, and papers [16,39] for differential sensors, while papers [5,38,40] refer to relaxation oscillators based on CCII.

Similar to the C-T technique, in the C-F technique, an oscillator circuit is employed to convert the capacitance change into an equivalent variation in frequency, but this frequency is usually in the order of MHz or even tens of MHz, and frequency estimation techniques are used to improve the measurement accuracy [42], e.g. frequency-locked loop [43] and phase-locked loop [44]. A different approach is proposed in [45] where a closed-loop switched-capacitor capacitance-to-frequency converter for differential sensors and single floating sensors is presented.

The RC charge/discharge techniques are based on the measurements of the charging or discharging times of the RC circuit consisting of a reference component and a capacitive sensor. Thanks to using the direct sensor-to-microcontroller approach, the interface circuit consists of only a few passive components (including the capacitive sensor). Generation of the excitation signal, time measurements, i.e. control of the measurement process, is carried out by the microcontroller, and more precisely by its peripheral devices such as timers/counters, analogue comparators (ACs) or analog-to-digital converters (ADCs). Proposals of interface circuits for single floating sensors are included in [3,4,46,50], for single grounded sensors in [48,51], and for differential sensors in [46,47,50].

The interface circuit based on a charge balancing method [23] consists of a circuit used to generate sinusoidal compensating driving signals, two parallel capacitance-to-phase conversion circuits (one with a reference sensor and the other with a measurement sensor) ending with Schmitt gates connected to one XOR gate which is used for detecting the phase difference between both sensor signals.

The last described method is based on a phase shifter [24]. The interface circuit consisting of two signal paths (reference circuit and sensor circuit paths) is stimulated by a square wave. A phase shift between a reference signal and the output signal of a phase shifter circuit (based on an active all-pass filter) with the capacitance sensor is measured using an oscilloscope.

Often, capacitive sensors are implemented as smart sensors; that is, they work as data acquisition systems. They can be a component of measurement devices or actuators, or, for example, act as sensor nodes of wired or wireless networks. In the latter case, they should be as simple as possible and also small and energy-efficient to reduce production and operating costs. For these reasons, microcontrollers can be used – they are one-chip low-power complete devices that are used as control and measurement systems (including data processing and storage), and they are increasingly equipped with communication interfaces.

For the smart sensors based on microcontrollers, simple and inexpensive direct sensor-to-microcontroller interface circuits have been developed [3,4,46-48]. As mentioned before, the interface circuits consist of only a few passive components, i.e. they do not contain active analogue circuits (operational amplifiers, analogue comparators, CCIIs, etc.). However, they do not have the same measurement accuracy as interface circuits based on, e.g. C-V techniques, which results, among other things, from the influence of the parasitic capacitances of the microcontroller pins. The values of these capacitances are different for different microcontrollers and, moreover, for different operating conditions, e.g. pin loads or operating temperature of the circuits. Hence, the parameters of the interface circuits are strictly dependent on the specific microcontroller used. Obviously, solutions minimising this impact are used: e.g. in [3,4,46,47], measurement procedures containing additional measurement cycles were proposed, and in [48,51,52], software calibration algorithms.

Thus, the paper proposes a new analogue interface circuit and a measurement procedure dedicated to microcontrollers, that are independent of these influences, i.e. they isolate the capacitive sensor from the microcontroller, without causing too much hardware redundancy. Hence, strict requirements were adopted when developing the interface circuit. First, the C-T technique is used – time measurement is much more accurate than voltage measurement by a microcontroller (microcontrollers have 16-bit timers/counters that can be cascaded as needed, and only 10- or 12-bit ADCs). Moreover, the interface circuits based on the C-V technique are complicated, as shown in Fig. 1 and in Table 1. Secondly, the operational amplifiers used must be placed in a single package and must be powered by a single voltage supplying the microcontroller, i.e. we can use the single/dual/quad rail-to-rail operational amplifiers available in one package.

The measurement method (an interface circuit and a measurement procedure) was designed and investigated for single grounded humidity sensors not distant from the system. The following novelties were proposed:

- A new analogue interface circuit based on a cascaded low-pass filter and a phase shifter connected to the microcontroller (Fig. 2).
- What is new is the fact that the low-pass filter, consisting of reference resistor R_3 and capacitive sensor C_x (Fig. 4), is treated as a phase shifter. The use of a typical phase shifter based on an all-pass filter [24] is impossible because it inverts the signal and requires a symmetrical power supply. Moreover, it emphasises the signal distortions, which was found during simulation tests with the use of LTspice.
- A new measurement procedure using to determine the value of C_x of the capacitive sensor.
- Measurement procedure algorithms (including software) for 8-bit microcontrollers implementing the measurement procedure.
- A correction (calibration) algorithm of systematic errors dedicated to the proposed analogue interface circuit based on a calibration dictionary and its software implementation.

The proposed method was analysed and experimentally verified with the use of a set of ceramic capacitors with values specified in [48,51] comprising capacitance values for commercial relative humidity sensors [53–55] instead of capacitive sensor C_x.

2. Operating principle

The smart sensor solution based on the proposed analogue interface circuit designed for microcontrollers for grounded capacitive sensors is presented in Fig. 2. It consists of three blocks: a microcontroller, a low-pass filter (Fig. 3), and a phase shifter (Fig. 4).

The microcontroller is responsible for two functions. The first is that it generates stimulation square wave v_{in} (v_{in} is equal to the supply voltage V_{CC} of the microcontroller) and period T at the output pin Pin 1 using one of its timers/counters. The second function is that it measures times t_1 and t_2 using the other of its timers/counters as shown in Fig. 5 and next calculates $\tau = t_2 - t_1$. Time t_1 is the delay introduced by the low-pass filter (sinus wave v_f), and time t_2 is the sum of delays introduced by the low-pass filter and by the phase shifter (sinus wave v_{ps}), hence the calculated delay τ represents the phase shift introduced only by the phase shifter containing the capacitive sensor. Signals v_f and v_{ps} are connected to the input of the AC with the threshold voltage $V_{th} = V_{CC}/2$ via an AMux (analogue multiplexer) (input pins Pin 2 and Pin 3, respectively). Voltage V_{th} can be set by any internal voltage reference source of the microcontroller (e.g. a digital-to-analogue converter (DAC), a voltage scaler, a set of voltage reference sources) or, if we do not have such capabilities, we can use a resistive divider built of R_4 and R_5 (connected to Pin 4 and Pin 5 as shown in Fig. 2). Hence, a necessary condition required by the method is that the microcontroller should have two timers/counters and one AC. Usually, the AC is equipped with an 8-channel (sometimes even 16-channel) AMux. These criteria are met even by mid-range 8-bit microcontrollers. Therefore, the proposed solution will be shown using the example of an 8-bit ATXmega32A4 microcontroller [56,57].

The low-pass filter block is based on a second-order Sallen-Key low-pass filter configuration (Fig. 3). It converts square wave v_{in} with amplitude V_{CC} and period T into sinus wave $v_f \approx V_{CC}/2 \cdot \sin(\omega t) + V_{CC}/2$, where $\omega = 2\pi/T$, by filtering the higher odd harmonics of the square wave.

The phase shifter consists of reference resistor R_3 and capacitive sensor C_x (Fig. 4) in the RC low-pass filter configuration, and a non-inverting voltage follower based on a rail-to-rail operational amplifier. An LTC2051 dual zero-drift operational amplifier in the MS8 package was chosen [58]. The R_3C_x circuit shifts signal v_f by τ, and also acts as a first-order low-pass filter (Fig. 5), additionally filtering the waveform of odd harmonics. The R_3C_x circuit can be considered a phase shifter because it is assumed that T (i.e. ω) is constant. The amplitude of the v_{ps} signal depends on the changing value of C_x, however, it has no significant effect under the condition $V_{th} = V_{CC}/2$, because both the v_f and v_{ps} signals are fed to the AC.

2.1. Measurement procedure algorithm

The ATXmega32A4 is powered by a supply voltage $V_{CC} = 3.3$ V and its clock source is a 16 MHz crystal oscillator. During the measurement procedure, square wave v_{in} at output pin PC0 of the microcontroller is generated by a 16-bit TC0 (Time/Counter C0) in Frequency (FRQ) Waveform Generation Mode. Simultaneously, the 16-bit TC1 (Time/Counter C1) used to determine t_1 and t_2 works in a Normal Mode with Capture on Channel A Enable, and it is controlled by the Event System (ES) by the Analogue Comparator number 0 (AC A0). Threshold voltage V_{th} of the AC A0 is controlled by the 12-bit DAC connected to the PORTB (DACB) with the reference set to V_{CC} [56]. Signal v_f is connected to input pin PA1, and signal v_{ps} to input pin PA2.

It is proposed a low energy consumption measurement algorithm which flowchart is presented in Fig. 6. To reduce energy consumption, only $M + 1$ measurements of times t_1 and t_2 are made, of which M is then averaged (e.g. $M = 64$). The first dummy single measurement is to bring the analogue interface circuit out of its initial state.

The algorithm of a single-measurement is shown in Fig. 7 and also in Fig. 8. This is due to the fact that this algorithm is partly implemented in software (Fig. 7), and more specifically in two sections of the code: the section of the main function (Main function) and the section of interrupt services (Interrupt services), and partly “included” in the configuration of the peripheral modules of the microcontroller setting up the time measurement microsystem (Fig. 8): cascaded connected AC0, ES, TC1, which, among other things, enables optimisation of energy consumption [59].

The main section code (Fig. 7)of the single-measurement function is responsible for executing:

- It sets pin PA1 as the input of the AC and sets a variable *end_meas* which is used to synchronise the hardware and software (handled by the interrupt service).
- Then we wait for the time measurement of t_1 to be completed by testing variable *end_meas*. When the value of voltage v_f achieves the value V_{th}, the AC generates an event (step 1 in Fig. 7 and Fig. 8), which writes the current value of TC1 to the register TCC1_CCA. Then a write event to this register generates an interrupt (step 2). At the end, the measured value of time t_1 is saved and variable *end_meas* is cleared – the result is to go to the main function.
- Next, it sets pin PA2 as the input of the AC and also sets variable *end_meas*.
- After that, step two of the algorithm is performed for the v_{ps} signal, and time t_2 is measured.
- At the end, time τ is calculated.

The code of_single- and multiple-measurement functions written in ANSI C takes only 1192 Bytes (including 54 Bytes of the interrupt service) of the program memory, and the configuration function of the peripheral devices occupies only 226 Bytes. It should be also mentioned that by using the subtraction operation ($\tau = t_2 - t_1$), we eliminate the influence of propagation delays introduced by the measurement path, including the low-pass filter, which is an advantage of this approach.

2.2. Analysis of the phase shifter circuit

The formula for transfer function $K_{ps}(s)$ of the phase shifter (Fig. 4) has the form:

$$K_{ps}(s) = \frac{1}{s(R_3 \cdot C_x) + 1} \tag{1}$$

Substituting $s = j \cdot \omega$, we obtain the following formulas for modulus $|K_{ps}(\omega)|$ and phase shift φ_{ps} of the transfer function of the circuit:

$$\left|K_{ps}(\omega)\right| = \frac{1}{\sqrt{(\omega \cdot R_3 \cdot C_x)^2 + 1}} \tag{2}$$

$$\varphi_{ps} = -\arctan(\omega \cdot R_3 \cdot C_x) \tag{3}$$

where:

$$\omega = \frac{2\pi}{T}\ ,\ \varphi_{ps} = \frac{2\pi}{T} \cdot \tau \ \text{ and } \tau = t_2 - t_1 \tag{4}$$

In practice, the measured value of sensor capacitance C_m is loaded with parasitic capacitance C_p, hence we can write that $C_m = C_x + C_p$. From this fact and (3) and (4), we can derive the formula for C_m:

$$C_m = \frac{T}{2\pi \cdot R_3} \cdot \tan\left(\frac{2\pi}{T} \cdot \tau\right) \tag{5}$$

Square wave v_{in} is generated, and times t_1 and t_2, i.e. time τ, are measured by 16-bit timers of the microcontroller clocked directly by the system clock with the period t_{clk}. If we assume that $T = N_{max} \cdot t_{clk}$ and $m_\tau = \tau / t_{clk}$ ($N_{max} = 2^{16}$ – the maximum value counted by the timer, m_τ – the value counted (determined) by the timer for time τ), we can write that:

$$C_m = \beta \cdot \tan(\gamma \cdot m_\tau) \tag{6}$$

where: $\beta = \frac{N_{\max} \cdot t_{clk}}{2\pi \cdot R_3}$, $\gamma = \frac{2\pi}{N_{\max}}$, and C_p – interface circuit constant parameters.

That is, we finally get the following formula for C_x:

$$C_x = C_m - C_p \tag{7}$$

Parameters β and γ are constant and describe the configuration of the system. They can be specified only once during writing of the software of the smart sensor. Whereas, parameter C_p, which represents the physical properties of the phase shifter circuit, can be measured or either determined by the PCB design software. These parameters can be saved as constants in the program code of the microcontroller. Therefore, calculations of the value of C_x are simple.

2.3. Determination of reference resistor R_3

Measurements of times t_1 and t_2 should be carried out with the best possible resolution of the 16-bit timer of the microcontroller to minimise the quantisation error. Therefore, it was assumed that period T of stimulation square wave v_{in} is equal to $N_{max} \cdot t_{clk}$. In the proposed prototype of the smart compact sensor, the microcontroller is clocked at 16 MHz (this is its maximum clock frequency [57, page 20]), therefore period T = 4096 μs. For this value of period T, the values of low-pass filter elements R_1, R_2, C_1, and C_2 (Fig. 3) were determined. If we would like to speed up the measurement procedure, i.e. shorten period T (for the assumed measurement resolution), then a microcontroller clocked with a higher frequency signal should be used.

It was also assumed that the value range for C_x is as follows [48,51]: $C_x \in \left\langle C_{x,\min} ; C_{x,\max} \right\rangle$, $C_{x,min}$ = 100 pF, and $C_{x,max}$ = 300 pF. Hence the middle value (which can be considered as a nominal value) is $C_{x,mean}$ = 200 pF. For this value, we should be in the middle of the phase shift measurement range, i.e. $\varphi_{ps} = \pi / 4$. Since the phase shifter is also a low-pass filter, $\omega = 2 \cdot \pi \cdot R_3 \cdot C_{x,mean} / T$. Hence, using (3) and (4), the determined value of R_3 is 3.18 MΩ (e.g. 3.62 MΩ for the nominal value 180 pF of a HS1101LF relative humidity sensor [53]).

Ultimately, the value of R_3 should be selected so that $m_{max\text{-}min} = m_\tau(C_{x,max}) - m_\tau(C_{x,min})$, which determines the dynamics of changes in the value of m_τ ($\tau = m_\tau \cdot t_{clk}$), is as high as possible, and thus that the measurement resolution of the sensor's capacitance is as high as possible. Therefore, the graphs shown in Fig. 9 were prepared. Fig. 9a shows the graphs of the value of m_τ depending on capacitance C_x and resistance R_3, while Fig. 9b and Fig. 9c show the graphs of $m_{max\text{-}min}$ and $m_{min} = m_\tau(C_{x,min})$ as a function of R_3, respectively. Therefore the value R_3 = 3.3 MΩ was chosen (based on the E6 value series). For this value, the theoretical measurement resolution is about 24 fF (1 pF → 41 bits).

For example, for a range of 33 pF – 100 pF we have R_3 = 10 MΩ and for a range of 10 pF – 30 pF we have R_3 = 33 MΩ. For these ranges, only simulation studies using LTspice were performed. They confirmed the possibility of using the proposed method also for such capacitance values.

3. Analysis of errors introduced by the interface circuit

The accuracy of determining the value of C_x is influenced by systematic errors resulting from inaccurate setting or stability of the constant parameters of the interface circuit, and systematic and random measurement errors of times t_1 and t_2, i.e. the time τ – the value m_τ.

A maximum possible relative uncertainty of an indirectly measurable variable $y = f(x_1,..,x_I)$ is given by the following formula [48,51,60,61]:

$$\frac{\Delta f}{|f|} = \frac{1}{|f|}\cdot\sum_{i=1}^{I}\left|\frac{\partial f}{\partial x_i}\right|\cdot|\Delta x_i|\,, \tag{8}$$

where: x_i – directly measurable variable (i = 1, 2, .., I), Δx_i – maximum absolute uncertainty of x_i.

We can notice from (5) and (6) that indirectly measurable value C_m depends on the accuracy of the R_3 value – ΔR_3, the stability of a quartz oscillator and the discretisation error of t_{clk} – Δt_{clk}, and the accuracy of the measured value of m_τ – Δm_τ. For these direct variables based on (8), we obtain the formula:

$$\frac{\Delta C_m}{|C_m(R_3,t_{clk},m_\tau)|} = \frac{1}{|C_m(R_3,t_{clk},m_\tau)|}\cdot\left(\left|\frac{\partial C_m(R_3)}{\partial R_3}\right|\cdot\Delta R_3 + \left|\frac{\partial C_m(t_{clk})}{\partial t_{clk}}\right|\cdot\Delta t_{clk} + \left|\frac{\partial C_m(m_\tau)}{\partial m_\tau}\right|\cdot\Delta m_\tau\right) \tag{9}$$

where: $C_m > 0$, $R_3 > 0$, $t_{clk} > 0$, and $m_\tau > 0$.

After the transformations of (9), we obtain the following formula for the maximum possible relative uncertainty of C_m:

$$\frac{\Delta C_m}{|C_m(R_3,t_{clk},m_\tau)|} = \frac{\Delta R_3}{R_3} + \frac{\Delta t_{clk}}{t_{clk}} + g(m_\tau)\cdot\Delta m_\tau \tag{10}$$

where:

$$g(m_\tau) = \frac{2\cdot\gamma}{|\sin(2\cdot\gamma\cdot m_\tau)|} \tag{11}$$

The first element $\Delta R_3 / R_3$ of formula (10) represents the tolerance of R_3 and also the thermal drift of R_3 [48,51]. For a prototype device or for small production series, we can measure the value of R_3 and save it in the microcontroller program code, which eliminates this error. Otherwise, it is suggested that resistor R_3 has a tolerance of 0.01%, which enables this influence on the accuracy of the determination of C_m to be neglected.

The second element $\Delta t_{clk} / t_{clk}$ of this formula can also be omitted. The stability of a quartz oscillator is at the level of 50 ppm (0.005%) and the discretisation error for measured times t_1 and t_2 is less than 0.006% and 0.005%, respectively, and it does not depend on the trigger noise [62,63] as the v_f and v_{ps} signals are fed to the AC.

Hence, the third element $\delta m_\tau = g(m_\tau)\cdot\Delta m_\tau$ of formula (10) has a significant influence on the maximum possible relative uncertainty of C_m. The graph of the $g(m_\tau)$ function is shown in Fig. 10a. For comparison purposes, the Δm_τ values obtained during the measurements for $M = 1$ (in Fig. 15a) and for $M = 64$ (in Fig. 16a) were used to prepare the charts in Fig. 10b and Fig. 10c, respectively. Comparing the relevant charts with each other, it can be seen that in both theoretical (Fig. 10b and Fig. 10c) and practical (Fig. 15b and Fig. 16b) cases, the δm_τ errors have similar values, which confirms the compliance of the theoretical assumptions. The main differences result from the fact that for smaller C_x values, we have less noise filtering.

The Δm_τ error depends on the precision of the AC triggering, i.e. the accuracy of the setting of the threshold voltage V_{th}:

$$V_{th} = \frac{V_{cc}}{2} \pm \Delta V_{th} + v_{noise} \tag{12}$$

Whereas ΔV_{th} represents the accuracy of the value of V_{th} set e.g. by the DAC or a voltage divider (R_4 and R_5) and also an input offset voltage of the AC ($v_{off} \leq 10$ mV [57, page 66] and is constant for a given microcontroller) – a systematic error which can be minimised. Whereas v_{noise} is the noise (or more precisely, disturbances) mostly generated by the digital systems of the microcontroller (e.g. by its core processor, timers and clock system) – a random error. Therefore, based on (12), we can write:

$$\Delta m_\tau = \Delta m_{\tau\_s}(\Delta V_{th}) + \Delta m_{\tau\_noise}(v_{noise}) \tag{13}$$

where: Δm_{τ_s} – a systematic error part of Δm_τ, Δm_{τ_noise} – a random error part of Δm_τ.

If we assume that $\Delta V_{th} = V_{off}$, $m_1 = t_1 / t_{clk}$, and $m_2 = t_2 / t_{clk}$, we can write the equation:

$$m_\tau = m_2 - m_1 \tag{14}$$

where:

$$m_1 = m'_1 + \Delta m_1(V_{off}, V_f) \quad \text{and} \quad m_2 = m_1 + m'_2 + \Delta m_2(V_{off}, V_{ps}(C_x)) \tag{15}$$

where: V_f – an amplitude of the v_f signal, and V_{ps} – an amplitude of the v_{ps} signal, m'_1 and m'_2 – ideal values of the measured times t_1 and t_2, and Δm_1 and Δm_2 – absolute error values derived from V_{off} and V_f or $V_{ps}(C_x)$, respectively.

After the transformations of (15), we obtain the formula for the Δm_{τ_s} error:

$$\Delta m_{\tau\_s} = \Delta m_2(V_{off},V_{ps}(C_x)) - \Delta m_1(V_{off},V_f) \tag{16}$$

Amplitude V_f is constant, while amplitude V_{ps} (i.e. the “slope” of the v_{ps} signal (Fig. 5)) depends on the value of capacitance C_x of the sensor when $V_{off} > 0$ V, as shown in Fig. 11b.

By transforming (16), we obtain the final form of the formula for Δm_{τ_s}:

$$\Delta m_{\tau\_s} = \frac{N_{\max}}{2\pi} \cdot \left[\arcsin\left(\frac{V_{off}}{V_{ps}(C_x)} \right) - \arcsin\left(\frac{V_{off}}{V_f} \right) \right] \tag{17}$$

A graph of Δm_{τ_s} as a function of C_x for four values of V_{off} = 1, 2, 5, 10 mV is drawn in Fig. 12. As shown in Fig. 12, in the case of the ideal $V_{th} = V_{CC} / 2$ ($Voff = 0$ V), the value of C_x has no influence on systematic error Δm_{τ_s}. However, in the case of a mismatch: $V_{th} \neq V_{CC} / 2$ ($Voff > 0$ V), this influence is proportional to $Voff$. Hence, in order to eliminate this error ($\Delta m_{\tau\_s} < 1$ LSB for the entire range of C_x), the condition $|V_{th} - V_{CC} / 2| < 1$ mV should be met. Obviously, the values of Δm_{τ_s} taken from this graph can be used to correct m_τ.

It should be mentioned that $\alpha = V_{th} / V_{CC} = 0.5$ set by the 12-bit DAC, as proposed in this paper, is constant for a given microcontroller and is not sensitive to temperature changes, as shown in [48,51]. To meet the above condition for the reference voltage $V_{ref} = V_{CC} = 3.3$ V, the value in the DAC should be set with accuracy to 3 LSB.

In the case of using resistive divider R_4, R_5 (Fig. 2) as the threshold voltage source of the AC, in order to minimise the Δm_{τ_s} error, the following conditions must be met: $R_4 + R_5 >> R_{pin3\_on}$ and $R_4 << R_{pin3\_off}$, where R_{pin3_on} – output pin resistance at a high voltage level (about 50 Ω), R_{pin3_off} – input pin resistance (about 3 GΩ) [57, page 67]. Fig. 13a shows a diagram of coefficient $\alpha = R_4 / (R_4 + R_5)$, and Fig. 13b shows the relative error $\delta\alpha$ of this coefficient depending on the value of $R = R_4 = R_5$ and the values of pin resistances R_{pin3_on}, and R_{pin3_off}. Hence, the smallest error $\delta\alpha = 0.0013\%$ is obtained for $R_4 = R_5 = 330$ kΩ (Fig. 13b).

It should also be remembered that tolerances $\Delta R_4 / R_4 = \Delta R_5 / R_5$ of R_4 and R_5 should be such that $V_{CC} \cdot |\alpha - 0.5| < 1$ mV. Hence, for a tolerance of 0.05%, we have $V_{CC} \cdot |\alpha - 0.5| < 0.825$ mV, and for a tolerance of 0.1%, we have $V_{CC} \cdot |\alpha - 0.5| < 1.6$ mV (the value of about 1.6 mV is for the worst case when $\Delta R_4 = -\Delta R_5$). Moreover, resistors R_4 and R_5 must have the same temperature coefficient of resistance so that α is not sensitive to temperature changes of these resistors.

4. Methods and materials

The measurement method including the analogue interface circuit (built on an LTC2051) was experimentally tested based on the prototype of the smart compact sensor controlled by the ATXmega32A4. The supply voltage $V_{CC} = 3.2997$ V, the threshold voltage $V_{th} = 1.6495$ V (the value 2073 written to the DACB_CH0DATA data register of the DAC B) – $\alpha = 0.49989$, and the reference resistor $R_3 = 3.2994$ MΩ. An Agilent 34410A was used to measure the voltages V_{CC} and V_{th} and the resistance R_3.

The sensor was replaced by a set $\{C_{x,i}\}_{i = 1, .., I}$ of $I = 12$ ceramic capacitors. An Agilent 4263B and a Hewlett-Packard 16047A were used to measure values of these capacitors (Table 2).

Table 2. Values of ceramic capacitors.

index i	1	2	3	4	5	6	7	8	9	10	11	12
$C_{x,i}$ [pF]	99.84	109.29	119.67	130.92	147.11	162.74	179.32	200.24	219.67	238.12	286.12	297.09

We carried out $L = 1024$ measurements for all $C_{x,i}$ values, and for two low energy consumption measurement modes: for a single-measurement of times (for $M = 1$) and for a multiple-measurement of times (for $M = 64$). Therefore, we performed a total of 66,560 measurements (dummy single measurements are not included). E.g. for the assumed range of capacitance values the multiple-measurements procedure (for $M = 64$ and $t_{clk} = 62.5$ ns) takes approximately 87 ms to 109 ms.

After each l-th single- and multiple-measurement ($l = 1, .., L$) for each i-th capacitance value $C_{x,i}$ the measured values of times $t_{1,i,l}$ and $t_{2,i,l}$, i.e. values read directly from the data register of timer/counter $m_{1,i,l}$ and $m_{2,i,l}$, and also the values $C_{m,i,l}$ computed by the microcontroller were sent to a computer used a USB interface.

5. Experimental results and discussion

For both measurement modes ($M = 1$ and $M = 64$), for each reference value $C_{x,i}$, the lowest values $m_{\tau\_min,i} = \min\limits_{l}\{m_{\tau,i,l}\}$, highest values $m_{\tau\_max,i} = \max\limits_{l}\{m_{\tau,i,l}\}$, and middle values $m_{\tau\_mean,i} = \left(m_{\tau\_min,i} + m_{\tau\_max,i}\right)/2$ from the measurement result sets $\{m_{\tau,i,l}\}_{l = 1, .., L}$ were calculated.

The scaling results of the analogue interface circuit are shown in Fig. 14, where the middle values $m_{\tau\_mean,i}$ were used to plot the measurement scaling curve. Additionally, in Fig. 14, there are simulation scaling curves generated with the use of Matlab and LTspice. It can be seen in Fig. 14 that these curves are similar in shape. The differences result from the fact that the ideal operational amplifier model was used in Matlab (it was also assumed that the parasitic capacitance C_p = 10 pF), and the simulations in LTspice were based on the LTC2051 amplifier model with parameters determined for a supply voltage of 5 V (which may affect the accuracy of the model for the supply voltage of 3.3 V). These differences, especially between the measurement curve and the Matlab simulation curve, can be treated as a graphic illustration of systematic errors. Therefore, we can use these results to develop an algorithm for correcting systematic errors, which will be described in the following subsection.

The noise effect (mainly due to disturbances) on the accuracy of measurements, that is, the random relative errors for single mode, are depicted in Fig. 15, and for multiple mode, in Fig. 16. These figures show absolute errors $\Delta m_{\tau\_noise,i}$ and relative errors $\delta m_{\tau\_noise,i}$ of determining time τ (m_τ), and the absolute error of determining capacitance $\Delta C_{m,i}$, as defined below:

$$\Delta m_{\tau\_noise,i} = \left(m_{\tau\_max,i} - m_{\tau\_min,i}\right)/2 \tag{18}$$

$$\partial m_{\tau\_noise,i} = \Delta m_{\tau,i} / \Delta m_{\tau\_mean,i} \cdot 100\% \tag{19}$$

$$\Delta C_{m,i} = \left(C_{m\_\max,i} - C_{m\_\max,i}\right)/2 \tag{20}$$

where: $C_{m\_min,i}$, and $C_{m\_max,i}$ are calculated based on (6) and the values of $m_{\tau\_min,i}$, $m_{\tau\_max,i}$, respectively.

In the first case (Fig. 15), Δm_{τ_noise} ranges from 33 to 72 LSB, with the result that δm_{τ_noise} is less than 0.7% and ΔC_m is less than 2 pF. E.g. this accuracy should be less than 0.2 pF to obtain an RH resolution at a level of 1% [53]. For this reason, a single-measurement mode is useless. However, when we want a fast and low-energy measurement and to only estimate the value of the sensor capacitance, we can use this approach.

However, for the multiple-measurement case (Fig. 16), error Δm_{τ_noise} is already in the range of 2–3 LSB. Thanks to this, error δm_{τ_noise} is smaller than 0.06%, and error ΔC_m is smaller than 72 fF. That is, a 64-fold increase in the measurement time reduced error ΔC_m by approximately 28 times. Of course, we can still extend the measurement time in order to obtain greater accuracy [48], but this is at the expense of the energy consumed by the device. For example, the measurements were carried out for the value $C_{x,1}$ and M = 256. Under these conditions, absolute error $\Delta m_{\tau\_noise,1}$ is

±1.5 LSB (ΔC_m less than 36 fF.), i.e. an additional fourfold increase in the measurement time contributed to an approximately twofold reduction of this error. It is even possible to continuously measure the sensor capacitance and read the result on demand, as it depends only on the microcontroller software.

For example, when compared to similar solutions presented in the literature, e.g. for a more complicated analogue interface circuit (3 rail-to-rail operational amplifiers and a comparator, two analogue multiplexers, some simple digital gates) for grounded capacitive sensors [41] dedicated to microcontrollers equipped with timers/counters for $M = 128$, error ΔC_m is less than 86 fF. Whereas, e.g. for the methods based on a direct sensor-to-microcontroller approach, this error is less than 100 fF for the method proposed in [48] for $M = 64$, and less than 62 fF for a 3-point calibration technique [3] for $M = 100$, respectively. So, the proposed method enables the sensor capacitance to be measured with a resolution similar to the presented methods, and moreover, it has a simple structure of the interface circuit and, importantly, isolates the sensor from microcontroller influences (both rail-to-rail operational amplifiers placed in one package operate in voltage follower configurations, as shown in Fig. 3 and Fig. 4).

Moreover, a proposed solution in relation to the direct sensor-microcontroller approaches increases the safety of the microcontroller operation, as the microcontroller is isolated from the sensor. In many cases, sensors must be located outside the housing of the device, therefore they are exposed to various types of damage. In such a case, for the direct connection of the sensor to the microcontroller (when using the direct sensor-microcontroller interface), damage to the sensor may result in damage to the microcontroller. By isolating the microcontroller from the sensor we eliminate this problem, what is an advantage of the proposed solution from a practical point of view. Furthermore, we can design the interface circuit as a separate module or e.g. as a PCB part design, calibrate it once and use it with any microcontroller.

In addition, the methods were compared in terms of energy consumption by the interface circuits. In order to be able to objectively compare the methods the following assumptions were made:

- the discharging times for the methods in [3,41], the charging time for the method in [48], and the time of a single-measurement for the proposed method are similar for maintaining the same measurement resolution and similar power consumption of the analogue interface circuit with the capacitance sensor,
- the same microcontroller is used (the ATXmega32A4 microcontroller, $V_{cc} = 3.3$ V, $f_{osc} = 16$ MHz),
- energy consumption by the microcontroller and active circuits during one measurement cycle will be compared.

Table 3. Comparison of current consumption by the interface circuits

Reference	**[3]**	**[41]**	**[48]**	**This work**
Microcontroller peripherals	core processor (a), timer (b)	core processor (a), timer (b)	core processor (a), timer, DAC, AC (b)	core processor (a), 2 · timer, DAC, AC (b)
Active components	-	some gates, MAX4560, OPA2350, MAX987, AD8607	-	LTC2051 (two op-amps)
Total current consumption by the microcontrller	7.519 mA ·3 (c) = 22.557 mA	7.519 mA ·3 (c) = 22.557 mA	8.792 mA	8.811 mA
Total current consumption by active components	-	15.199 mA · 3 (c) = 45.597 mA	-	1.5 mA
Total current consumption	22.557 mA	68.154 mA	8.792 mA	10.248 mA

Notes: (a) - the core processor of the ATXmega32A4 microcontroller consumes about 7.5 mA [57, page 70] in the active mode at 3.3 V, for a 16 MHz external clock and at 25°C (there is no information about using low-power consumption techniques in [3,41,48]), (b) - current consumption of peripheral devices read from the documentation [57, page 62], (c) – methods proposed in [3,41] are based on a 3- point calibration technique – one measurement cycle consists of measurements of three discharging times, hence the multiplier of 3.

As shown in Table 3, the proposed solution is characterized by low energy consumption. This consumption is much lower than for the methods proposed in [3,41] and only about 16.6% higher than for the method described in [48], which is mainly due to the current consumption of two operational amplifiers.

5.1. A systematic error calibration (correction) approach

It was assumed that the calibration (correction) of the systematic errors should be done by software, so as not to complicate (expand) the analogue interface circuit. The proposal to minimize the systematic errors is based on a calibration dictionary, similar to that presented in [51,64]. The calibration dictionary and an algorithm of the systematic error corrections described below were implemented in the microcontroller software (written in ANSI C).

The calibration dictionary is created as follows. First, a set of points $C_{x,i} = f(m_{\tau\_mean,i})$ is plotted in a plane with coordinates m_{τ_mean}, C_x (Fig. 17). We then use the spline functions described by 3rd-degree polynomials $P(\cdot)$ to approximate this set of points. Hence, the cœfficients $a_{j,k}$ ($k = 1,.., 4$) of a given j-th polynomial $P_j(\cdot)$ are always determined based on four points from I points, so that the polynomials overlap at least in one knot, in the following way:

$j = 1$ for $n = 1, 2, 3, \mathbf{4}$,
$j = 2$ for $n = \mathbf{4}, 5, 6, \mathbf{7}$,
$j = 3$ for $n = \mathbf{7}, 8, \mathbf{9}, \mathbf{10}$,
$j = 4$ for $n = \mathbf{9}, \mathbf{10}, 11, 12$,

where point number n in bold is the common knot for adjacent functions, as shown in Fig. 17.

So we can write that for each $j = 1, 2 ,3, ...$, we have four points with indexes $n = r + 3\cdot(j - 1)$, where $r = 1, .., 4$ and $n \leq I$. Thanks to this, we obtain a set of values $\{a_{j,k}\}_{j = 1,.., 4,\ k = 1,.., 4}$ which is a component of the calibration dictionary. The rest of the calibration dictionary consists of values $\{m_j\}_{j = 1,.., 4}$ used to select the given j-th spline function. These values are determined as follows: $m_j = m_{\tau\_max,p}$, where $p = 3j + 1$, if $p > I$ then $p = I$. Hence finally this dictionary can be written as follows: $\{a_{j,k}, m_j\}_{j = 1,.., 4,\ k = 1,.., 4}$. Hence the dictionary is small because it contains only 16 elements of the type "double" and 4 ones of the type "integer". That is, it takes only 104 bytes of program memory of the microcontroller.

Thus, the determination of the value of C_x based on the systematic error calibration algorithm consists of the following steps:

1. The value of $m_\tau = m_2 - m_1$ is determined based on the measured values m_1 and m_2 of times t_1 and t_2 – performed by the measurement function.
2. The value of index j is determined based on $\{m_j\}_{j = 1,.., 4}$ and a simple procedure presented in Listing 1.
3. The corrected value of capacitance sensor C_{m_corr} is calculated based on $\{a_{j,k}\}_{k = 1,.., 4}$ for a given j and (21).

```
...
uint_16_t m_j[4] = {6587, 8237, 9729, 10818};
                // i = 4,    7,   10,    12
...
j = 0;    // j = 0, .., 3 for ANSI C
if(m_tal > m_j[3]) {
     j = 3;
}
else {
     while(!(m_tal <= m_j[j])){
          j++;
     }
}
```

Listing. 1. Code implementing the procedure for selecting the value of index *j*

$$C_{m\_corr} = \sum_{k=1}^{4} a_{j,k} \cdot m_{\tau}^{k-1} \tag{21}$$

a_{jk} – coefficients of the *j*-th spline function determined by the calibration algorithm.

Thanks to the presented approach, it is possible to completely reduce the systematic error for a given analogue interface circuit. Thus, the determined value of C_{m_corr} is burdened with only the random errors, as presented earlier.

Moreover, the computational complexity of formula (21) implemented in the C language is lower than that of formula (6) containing the function *tan*() from the GNU C library libm.a (math.h). Only three additions and four multiplication operations are enough to calculate the value of C_{m_corr} using (21).

6. Conclusions

We presented a new measurement method developed for determining values of capacitive sensors, dedicated, inter alia, to grounded relative humidity sensors. The method is intended for microcontrollers with buit-in timers/counters and ACs. The new structure of the analogue interface circuit consists of a low-pass filer (a second-order Sallen-Key low-pass filter) and a phase shifter with a capacitive sensor (an RC low-pass filter). Hence, we use only two rail-to-rail operational amplifiers placed in one package. Also, a measurement procedure and a algorithm of systematic errors correction based on a calibration dictionary together with their implementations in the microcontroller's software were elaborated.

The experimental studies were carried out based on an example of a prototype controlled by an 8-bit microcontroller. They confirmed that for the multiple-measurement approach ($M = 64$) relative errors of capacitance measurements introduced by the analogue interface circuit are < 0.06%, which gives a capacitance resolution < 72 fF for the range of assumed capacitance (100 pF to 300 pF).

Despite the simple structure of the analogue interface circuit, we obtained so good results thanks to the use of an algorithm of systematic errors correction (calibration) based on a calibration dictionary dedicated to this circuit and the multiple-measurement approach.

In addition, software has been written in ANSI C for an exemplary 8-bit microcontroller (ATXmega32A4) implementing the measurement method, which confirms that the limited computing power and sizes of data and program memories of mid-range 8-bit microcontrollers are sufficient for this method.

References

[1] T. Islam, A. U. Khan, J. Akhtar, M. Z. U. Rahman, A Digital Hygrometer for Trace Moisture Measurement, IEEE Transactions on Instrumentation and Measurement 61 (10) (2014) 5599-5605.

[2] A. Rivadeneyra, J. Fernandez-Salmeron, M. Agudo-Acemel, J. A. Lopez-Villanueva, L. F. Capitan-Vallvey, A. J. Palmac, Printed electrodes structures as capacitive humidity sensors: A comparison, Sensors and Actuators A 244 (2016) 56–65.

[3] F. Reverter, O. Casas, Direct interface circuit for capacitive humidity sensors, Sensors and Actuators A 143 (2008) 315–322.

[4] J. Pelegrí-Sebastiá, E. García-Breijo, J. Ibáńez, T. Sogorb, N. Laguarda-Miro, J. Garrigues, Low-Cost Capacitive Humidity Sensor for Application Within Flexible RFID Labels Based on Microcontroller Systems, IEEE Transactions on Instrumentation and Measurement 61 (2) (2012) 545-553.

[5] P. B. Petrović, M. V. Nikolić, M. Tatović, New Electronic Interface Circuits for Humidity Measurement Based on the Current Processing Technique, MEASUREMENT SCIENCE REVIEW 21 (1) (2021) 1-10.

[6] J. P. Sanjurjo, E. Prefasi, C. Buffa, R. Gaggl, A Capacitance-To-Digital Converter for MEMS Sensors for Smart Applications, Sensors 17 (2017) 1312.

[7] C. Mahata, H. Algadi, J. Lee, S. Kim, T. Lee, Biomimetic-inspired micro-nano hierarchical structures for capacitive pressure sensor applications, Measurement 151 (2020) 107095.

[8] K B. Balavalad, B. G. Sheeparamatti, A Critical Review of MEMS Capacitive Pressure Sensors, Sensors & Transducers 187 (4) (2015) 120-128.

[9] F. Han, Q. Wu, R. Zhang, J. Dong, Capacitive Sensor Interface for an Electrostatically Levitated Micromotor, IEEE Transactions on Instrumentation and Measurement 58 (10) (2009) 3519-3526.

[10] Y. Piao, X. Xiao, X. Wang, Q. Zhou, K. Fan and P. Liu, Conditioning circuit for capacitive position sensor with nano-scale precision based on AC excitation principle, 2011 IEEE International Conference on Electro/Information Technology (2011) 1-6.

[11] L. Faller, J. Leitzke and H. Zangl, Design of a fast, high-resolution sensor evaluation platform applied to a capacitive position sensor for a micromirror, 2017 IEEE International Instrumentation and Measurement Technology Conference (2017) 1-6.

[12] S. Rombacha, M. Marxa, S. Gu-Stoppelc, Y. Manolia, Low Power Closed-Loop Driving Circuit for Piezoelectric Microscanners based on Tuneable Capacitive Position Sensors, Procedia Engineering 120 (2015) 63–66.

[13] R. Nojdelov and S. Nihtianov, Capacitive sensor interface with improved dynamic range and stability, 2014 IEEE International Instrumentation and Measurement Technology Conference (I2MTC) Proceedings (2014) 1373-1376.

[14] N. Anandan, B. George, A Wide-Range Capacitive Sensor for Linear and Angular Displacement Measurement, IEEE Transactions on Industrial Electronics 64 (7) (2017) 5728-5737.

[15] F. Restagno, J. Crassous, E. Charlaix, M. Monchanin, A new capacitive sensor for displacement measurement in a surface-force apparatus, Measurement Science and Technology 12 (2001) 16–22.

[16] R. A. Brookhuis, T.S.J. Lammerink, R.J. Wiegerink, Differential capacitive sensing circuit for a multi-electrode capacitive force sensor, Sensors and Actuators A 234 (2015) 168–179.

[17] J. Boudaden, A. Klumpp, H. E. Endres, I. Eisele, Capacitive CO_2 Sensor, Proceedings 1 (2017) 472.

[18] P. L. Kebabian, A. Freedman, Fluoropolymer-based capacitive carbon dioxide sensor, Measurement Science and Technology 17 (2006) 703–710.

[19] K.. A S Al Khateeb, R. T. Anika, S. Khan, M. Mohamud, A. Arshad, K. Hasan, S. S. Haider, M. M. Shobaki, Experimental evaluation of agricultural biomass flow sensing behaviour using capacitive technique, IOP Conf. Series: Materials Science and Engineering 53 (2013) 012034.

[20] C. Chiang and Y. Huang, A Semicylindrical Capacitive Sensor With Interface Circuit Used for Flow Rate Measurement, IEEE Sensors Journal 6 (6) (2006) 1564-1570.

[21] C. Wang, K. Shida, A new method for on-line monitoring of brake fluid condition using an enclosed reference probe, Measurement Science and Technology 18 (2007) 3625–3635.

[22] B. Kumar, G. Rajita, N. Mandal, A Review on Capacitive-Type Sensor for Measurement of Height of Liquid Level, Measurement and Control 47 (7) (2014) 219–224.

[23] M. Z. Aslam, T. B. Tang, A High Resolution Capacitive Sensing System for the Measurement of Water Content in Crude Oil, Sensors 14 (2014) 11351-11361.

[24] A. Jaworek, A. Krupa, Phase-shift detection for capacitance sensor measuring void fraction in two-phase flow, Sensors and Actuators A: Physical 160 (1–2) (2010) 78-86.

[25] A. Fendri, R. Jribi, O. Kanoun, H. Ghariani, Interface circuit for oil quality assessment considering dielectric losses and stray capacitances: Advanced Applications: Battery Research, Bioimpedance, System Design (2018) 10.1515/9783110558920-010.

[26] S. C. Bera, H. Mandal, A Flow Measurement Technique Using a Noncontact Capacitance-Type Orifice Transducer for a Conducting Liquid, IEEE Transactions on Instrumentation and Measurement 61 (9) (2012) 2553-2559.

[27] D. M. G. Preethichandra, K. Shida, A Simple Interface Circuit to Measure Very Small Capacitance Changes in Capacitive Sensors, IEEE Transactions on Instrumentation and Measurement 50 (6) (2001) 1583-1586.

[28] M. A. Haberman, E. M. Spinelli, F. Reverter, High-Linearity Front-End Circuit for Remote Grounded Capacitive Sensors, IEEE Transactions on Instrumentation and Measurement 70 (2021) 2003108.

[29] G. Ferri, F. R. Parente, V. Stornelli, G. Barile, L. Pantoli, Automatic bridge-based interface for differential capacitive full sensing, Procedia Engineering 168 (2016) 1585 – 1588.

[30] P. Mantenuto, A. De Marcellis, G. Ferri, Novel Modified De-Sauty Autobalancing Bridge-Based Analog Interfaces for Wide-Range Capacitive Sensor Applications, IEEE Sensors Journal 14 (5) (2014) 1664-1672.

[31] Y. Meng, R. N. Dean, A Technique for Improving the Linear Operating Range for a Relative Phase Delay Capacitive Sensor Interface Circuit, IEEE Transactions on Instrumentation and Measurement 65 (3) (2016) 624-630.

[32] Y. Meng , R. N Dean, M. L. Adams, Improving the phase delay capacitive interface circuit technique using MOSFET switches, Measurement Science and Technology 31 (2020) 025107.

[33] B. George, V. J. Kumar, Switched Capacitor Signal Conditioning for Differential Capacitive Sensors, IEEE Transactions on Instrumentation and Measurement 56 (3) (2007) 913-917.

[34] L. Polak, R. Sotner, J. Petrzela, J. Jerabek, CMOS Current Feedback Operational Amplifier-Based Relaxation Generator for Capacity to Voltage Sensor Interface, Sensors 18 (12) (2018) 4488.

[35] A. U. Khan, T. Islam, J. Akhtar, An Oscillator Based Active Bridge Circuit for interfacing Capacitive Sensors with Microcontroller Compatibility, IEEE Transactions on Instrumentation and Measurements 65 (11) (2016) 2560-2568 .

[36] Z. Ignjatovic, M. F. Bocko, An Interface Circuit for Measuring Capacitance Changes Based Upon Capacitance-to-Duty Cycle (CDC) Converter, IEEE Sensors Journal 5 (3) (2005) 403-410.

[37] M. Gasulla, X. Li, G. C. M. Meijer, The Noise Performance of a High-Speed Capacitive-Sensor Interface Based on a Relaxation Oscillator and a Fast Counter, IEEE Transactions on Instrumentation and Measurement 54 (5) (2005) 1934-1940.

[38] Q. Jia, G. C. M. Meijer, X. Li, C. Guan, An integrated interface for grounded capacitive sensors, SENSORS, 2005 IEEE (2005) doi: 10.1109/ICSENS.2005.1597890.

[39] A. De Marcellis, C. Reig,M. Cubells-Beltrán, A Capacitance-to-Time Converter-Based Electronic Interface for Differential Capacitive Sensors, Electronics 80 (8) (2019) 8010080.

[40] S. Malik, K. Kishore, Artee, S. A. Akbar, T. Islam, A CCII-based relaxation oscillator as a versatile interface for resistive and capacitive sensors, 2016 3rd International Conference on Signal Processing and Integrated Networks (2016) 359-363.

[41] F. Reverter, X. Li, G. C M Meijer, A novel interface circuit for grounded capacitive sensors with feedforward-based active shielding, Measurement Science and Technology 19 (2008) 025202 (5pp).

[42] T. Tyagi, P. Sumathi, Frequency estimation techniques in capacitance-to-frequency conversion measurement, Review of Scientific Instruments 91 (2020) 015005.

[43] R. N. Dean, A. K. Rane, A digital frequency-locked loop system for capacitance measurement, IEEE Transactions on Instrumentation and Measurement 62 (4) (2013) 777–784.

[44] T. M. Supon, K. Thangarajah, R. Rashidzadeh, M. Ahmadi, A PLL based readout and built-in self-test for MEMS sensors, IEEE 54th Int. Midwest Symp. Circuits Syst., Seoul, Korea, (2011) 1–4.

[45] L. Areekath, B. George, F. Reverter, A Closed-Loop Capacitance-to-Frequency Converter for Single-Element and Differential Capacitive Sensors, IEEE Transactions on Instrumentation and Measurement 69 (11) (2020) 8773-8782.

[46] F. Reverter, The Art of Directly Interfacing Sensors to Microcontrollers, Journal of Low Power Electronics and Applications (2) (2012) 265-281.

[47] F. Reverter, O. Casas, Interfacing Differential Capacitive Sensors to Microcontrollers: A Direct Approach, IEEE Transactions on Instrumentation and Measurement 59 (2010) 2763-2769.

[48] Z. Czaja, A measurement method for capacitive sensors based on a versatile direct sensor-to-microcontroller interface circuit, Measurement 155 (2020) 107547.

[49] C. Jiang, C. Chai, Y. Yang, Y. Yang, A high-linearity capacitance-to-digital converter with capacitive offset cancellation technique, IEICE Electronics Express 17 (18) (2020) 1–6.

[50] T. Islam, Advanced interfacing techniques for the capacitive sensors. In Advanced Interfacing Techniques for Sensors, Springer (2017), SSMI 25, 73-109.

[51] Z. Czaja, A measurement method for lossy capacitive relative humidity sensors based on a direct sensor-to-microcontroller interface circuit, Measurement 170 (2021) 108702.

[52] Z. Czaja, An implementation of a compact smart resistive sensor based on a microcontroller with an internal ADC, Metrology and Measurement Systems 23 (2016) 255-238.

[53] TE Connectivity Ltd., HS1101LF Relative Humidity Sensor, SENSOR SOLUTIONS /// HS1101LF HPC052_J (2015).

[54] Philips Components, Humidity sensor 2322 691 90001 Product specification (1996).

[55] Innovative Sensor Technology IST AG, P14 Rapid Capacitive Humidity Sensor, Available at: https://www.ist-ag.com/sites/default/files/DHP14-Rapid_E.pdf.

[56] Atmel Corporation, 8-bit XMEGA A Microcontroller, XMEGA AU MANUAL, (2013), Available at: http://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-8331-8-and-16-bit-AVR-Microcontroller-XMEGA-AU_Manual.pdf.

[57] Atmel Corporation, 8/16-bit AVR XMEGA A4 Microcontroller. ATxmega128A4, ATxmega64A4, ATxmega32A4, ATxmega16A4, (2013), Available at: http://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-8069-8-and-16-bit-AVR-AMEGA-A4-Microcontrollers_Datasheet.pdf.

[58] Linear Technology Corporation, LTC2051/LTC2052 20512fd Dual/Quad Zero-Drift Operational Amplifiers (2000), Available at: https://www.analog.com/media/en/technical-documentation/data-sheets/20512fd.pdf.

[59] F. Reverter, Toward Non-CPU Activity in Low-Power MCU-Based Measurement Systems, IEEE Transactions On Instrumentation and Measurement 69 (1) (2020), 15-17.

[60] K. Kolikov, G. Krastevy, Y. Epitropov, A. Corlat, Analytically determining of the relative inaccuracy (error) of indirectly measurable variable and dimensionless scale characterizing quality of the experiment, Computer Science Journal of Moldova 20 (58) (2012) 15-32.

[61] I. Farrance, R. Frenkel, Uncertainty of measurement: A review of the rules for calculating uncertainty components through functional relationships, The Clinical Biochemist Reviews 33 (2012), 49-75.

[62] F. Reverter, R. Pallàs-Areny, Effective number of resolution bits in direct sensor-to-microcontroller interfaces, Measurement Science and Technology 15 (2004) 2157–2162.

[63] F. Reverter, R. Pallàs-Areny, Uncertainty reduction techniques in microcontroller-based time measurements, Sensors and Actuators A 127 (2006) 74–79.

[64] W. Toczek, Z. Czaja, Diagnosis of fully differential circuits based on a fault dictionary implemented in the microcontroller systems, Microelectronics Reliability 51 (8) (2011) 1413-1421.

Figures captions

Fig. 1. Classification of methods based on the C-V conversion technique.

Fig. 2. Scheme of a proposed analogue interface circuit connected to a microcontroller.

Fig. 3. Block scheme of a second-order Sallen-Key low-pass filter

Fig. 4. Block scheme of a phase shifter with a capacitive sensor C_x.

Fig. 5. Timings of the stimulation signal $v_{in}(t)$, the output signal of the low-pass filter $v_f(t)$ and the output signal of the phase shifter $v_{ps}(t)$ for $V_{CC} = 3.3$ V, $T = 4.096$ ms, $R_3 = 3.3$ MΩ, $C_x = 100$ pF.

Fig. 6. The multiple-measurement procedure algorithm.

Fig. 7. The single-measurement procedure algorithm.

Fig. 8. Configuration of peripheral measuring circuits of the microcontroller.

Fig. 9. (a) Graphs of the value of m_τ depending on the capacitance C_x and the resistance R_3. (b) Graph of $m_{max\text{-}min}$ as a function of R_3. (c) Graph $m_{min} = m_\tau(C_{x,min})$ as a function of R_3.

Fig. 10. (a) Graph of the $g(m_\tau)$ function of m_τ. (b) Relative uncertainties δm_τ for values Δm_τ obtained during the measurements for $M = 1$ in Fig. 15b. (c) Relative uncertainties δm_τ for values Δm_τ obtained during the measurements for $M = 64$ in Fig. 16b.

Fig. 11. (a) Graph of a phase shift m_τ as a function of C_x values for the capacitive sensor. (b) Graph of a module $|Vps|$ as a function of C_x values for the capacitive sensor.

Fig. 12. Graph of an absolute uncertainty Δm_{τ_s} as a function of C_x for four values of $V_{off} = 1, 2, 5, 10$ mV.

Fig. 13. (a) Graph of the coefficient $\alpha = R_4 / (R_4 + R_5)$ as a function of $R = R_4 = R_5$. (b) Graph of the relative error $\delta\alpha$ of the coefficient α as a function of $R = R_4 = R_5$.

Fig. 14. Scaling curves of the analogue interface circuit based on measurement results and simulation results with the use of Matlab and LTspice.

Fig. 15. Single-measurement approach case: (a) Absolute random measurement errors Δm_{τ_noise}. (b) Random relative measurement errors δm_{τ_noise}. (c) Absolute random errors of determining the capacitance ΔC_m.

Fig. 16. Multiple-measurement approach case: (a) Absolute random measurement errors Δm_{τ_noise}. (b) Random relative measurement errors δm_{τ_noise}. (c) Absolute random errors of determining the capacitance ΔC_m.

Fig. 17. Graphs of a set of measurement points and a set of spline functions approximating this set of points.

Figure 1.

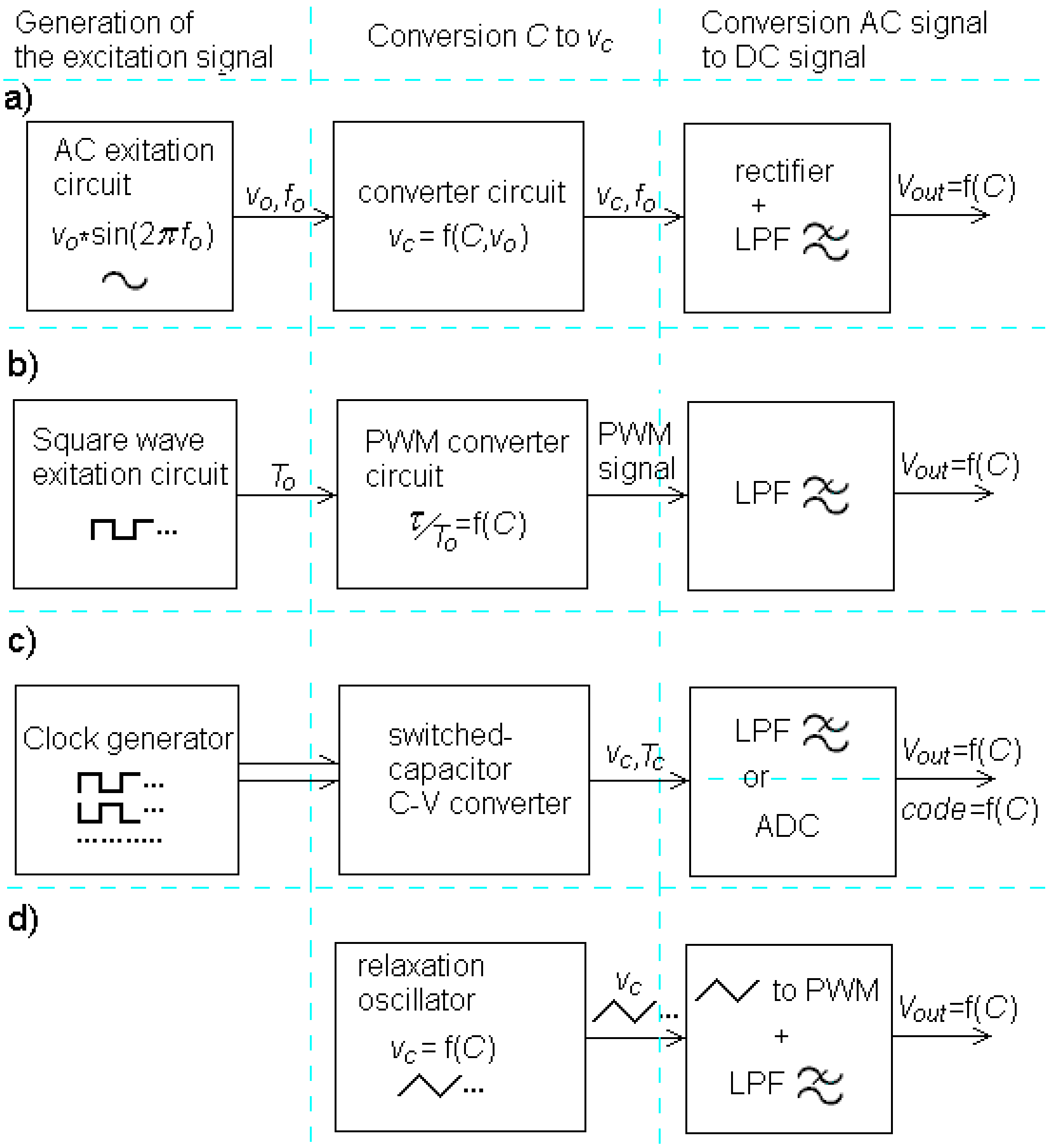

Generation of the excitation signal
Conversion C to v_c
Conversion AC signal to DC signal
a)
AC exitation circuit
v_o*sin(2πf_o)
v_o, f_o
converter circuit
v_c = f(C,v_o)
v_c, f_o
rectifier + LPF
V_out=f(C)
b)
Square wave exitation circuit
T_o
PWM converter circuit
τ/T_o=f(C)
PWM signal
LPF
V_out=f(C)
c)
Clock generator
switched-capacitor C-V converter
v_c,T_c
LPF or ADC
V_out=f(C)
code=f(C)
d)
relaxation oscillator
v_c = f(C)
v_c
to PWM + LPF
V_out=f(C)

Figure 2.

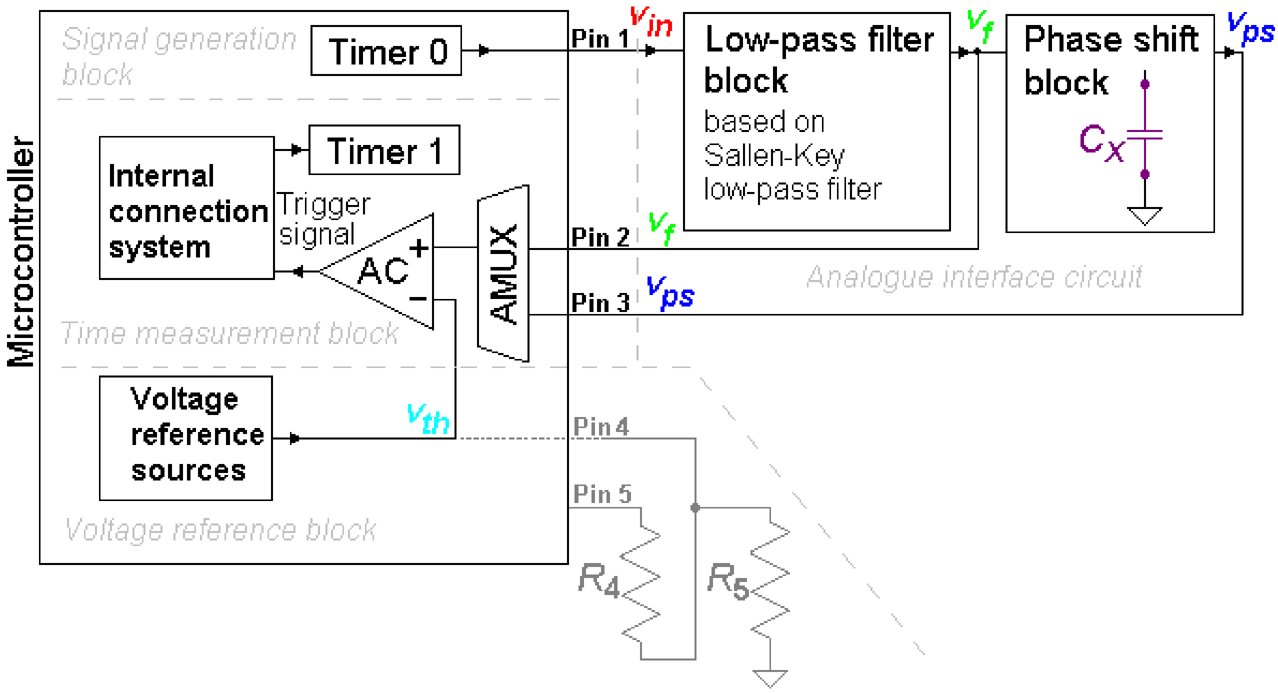

Microcontroller
Signal generation block
Timer 0
Timer 1
Internal connection system
Trigger signal
AC
+
−
AMUX
Time measurement block
Voltage reference sources
v_{th}
Voltage reference block
Pin 1
v_{in}
Low-pass filter block
based on Sallen-Key low-pass filter
v_f
Phase shift block
C_x
v_{ps}
Pin 2
v_f
Pin 3
v_{ps}
Analogue interface circuit
Pin 4
Pin 5
R_4
R_5

Figure 3.

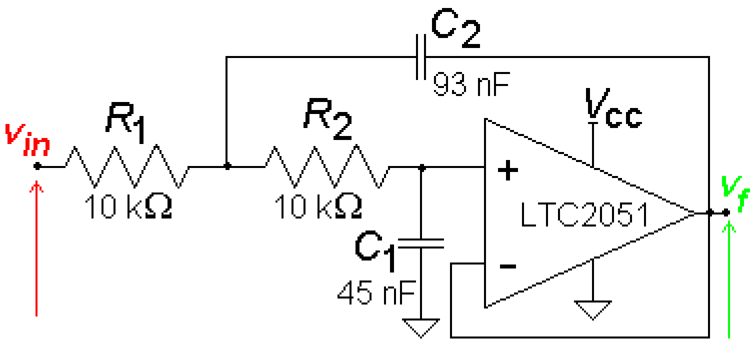


Figure 4.

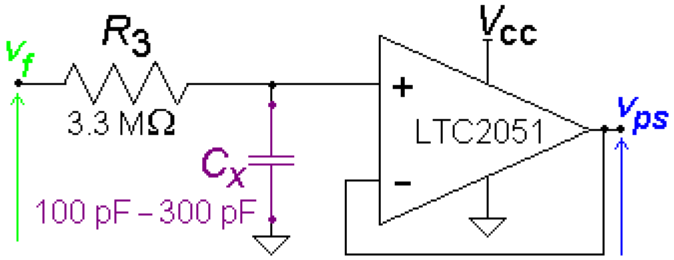


Figure 5.

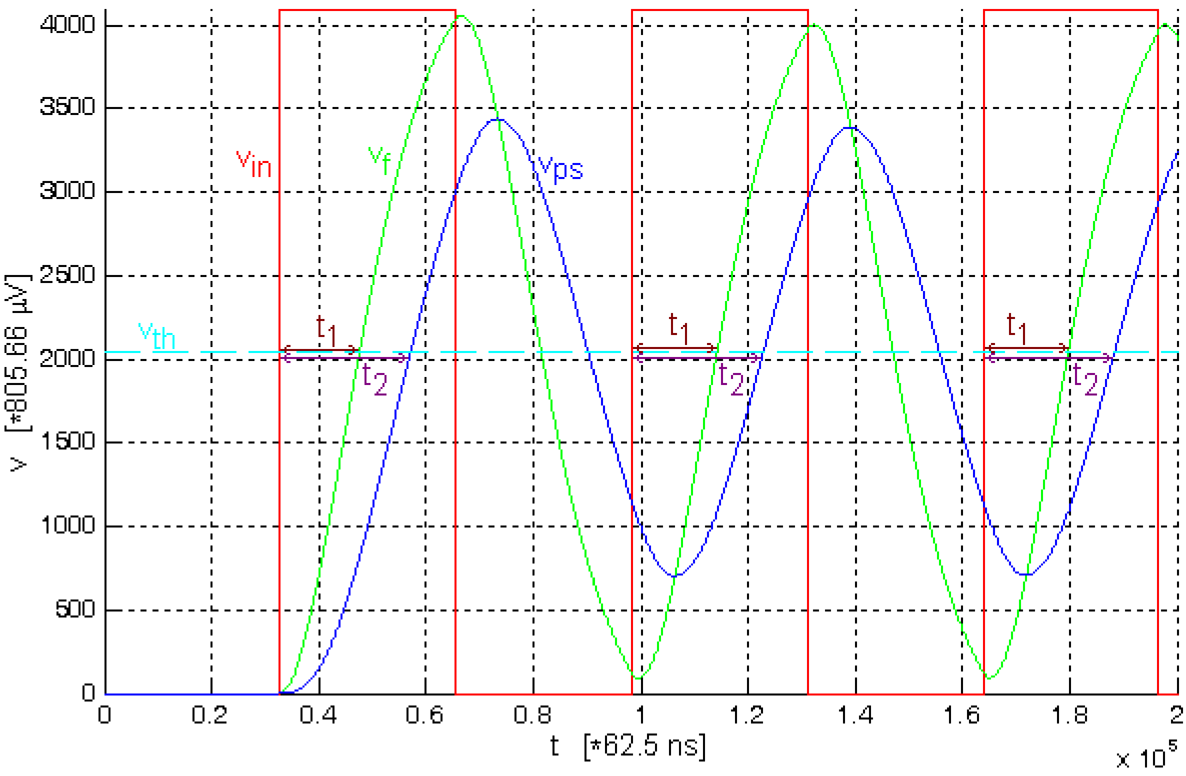

4000
3500
3000
2500
2000
1500
1000
500
0
v [*805.66 µV]
v_{in}
v_f
v_{ps}
v_{th}
t_1
t_2
0
0.2
0.4
0.6
0.8
1
1.2
1.4
1.6
1.8
2
t [*62.5 ns]
x 10^5

Figure 6.

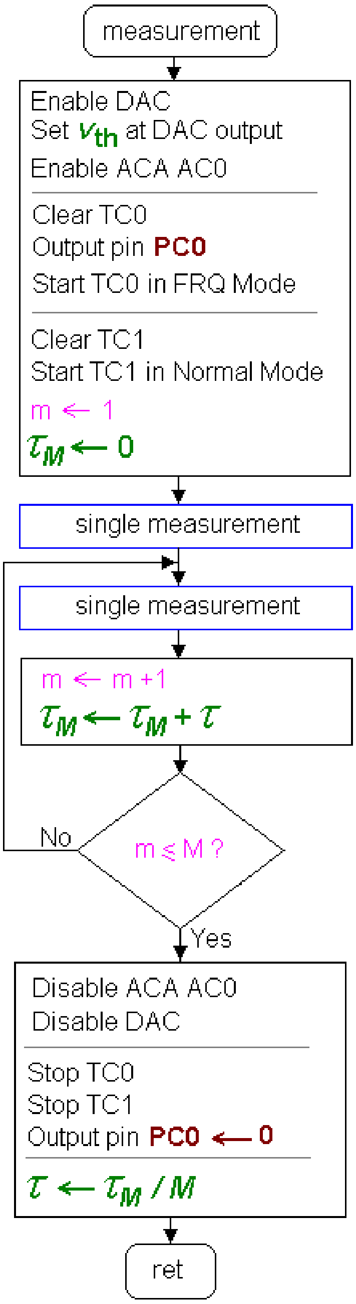

measurement
Enable DAC
Set v_{th} at DAC output
Enable ACA AC0
Clear TC0
Output pin PC0
Start TC0 in FRQ Mode
Clear TC1
Start TC1 in Normal Mode
m ← 1
$\tau_M \leftarrow 0$
single measurement
single measurement
m ← m +1
$\tau_M \leftarrow \tau_M + \tau$
No
m ≤ M ?
Yes
Disable ACA AC0
Disable DAC
Stop TC0
Stop TC1
Output pin PC0 ← 0
$\tau \leftarrow \tau_M / M$
ret

Figure 7.

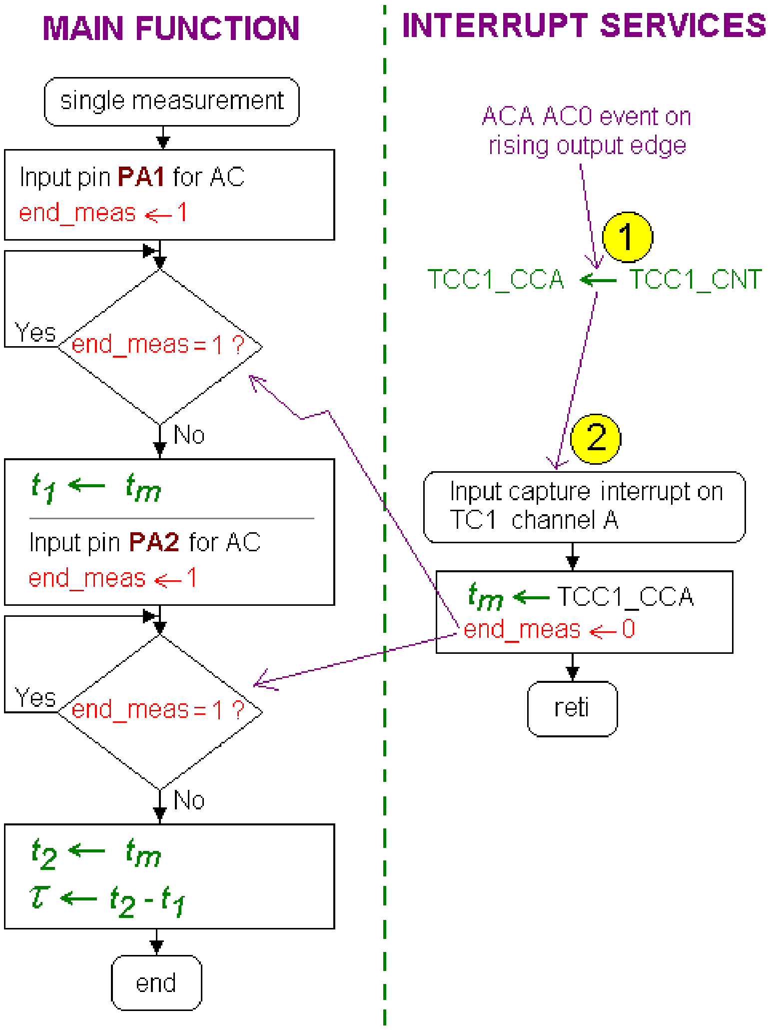

MAIN FUNCTION
INTERRUPT SERVICES
single measurement
Input pin PA1 for AC
end_meas ← 1
Yes
end_meas = 1 ?
No
t1 ← tm
Input pin PA2 for AC
end_meas ← 1
Yes
end_meas = 1 ?
No
t2 ← tm
τ ← t2 - t1
end
ACA AC0 event on rising output edge
1
TCC1_CCA ← TCC1_CNT
2
Input capture interrupt on TC1 channel A
tm ← TCC1_CCA
end_meas ← 0
reti

Figure 8.

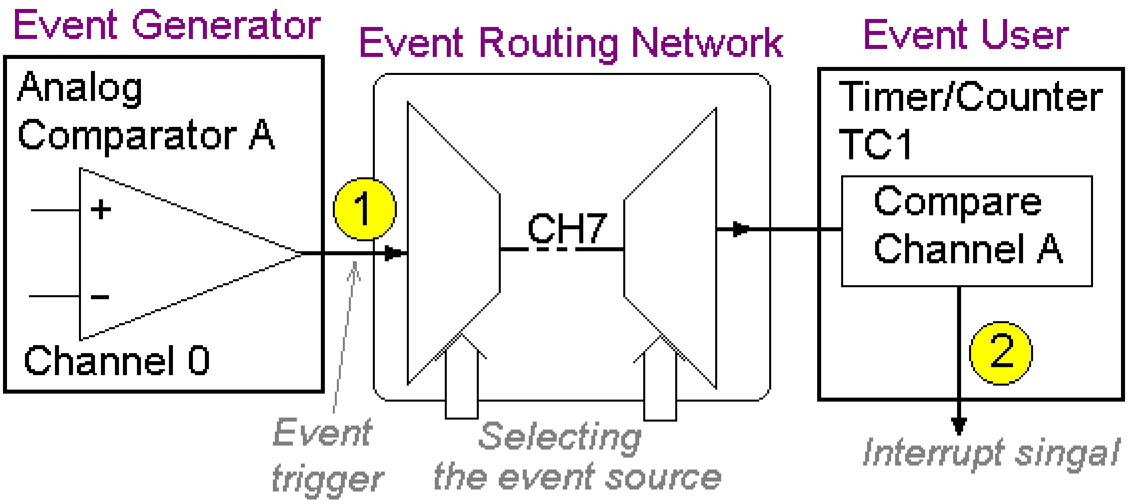

Event Generator
Event Routing Network
Event User
Analog Comparator A
+
–
Channel 0
1
CH7
Timer/Counter TC1
Compare Channel A
2
Event trigger
Selecting the event source
Interrupt singal

Figure 9.

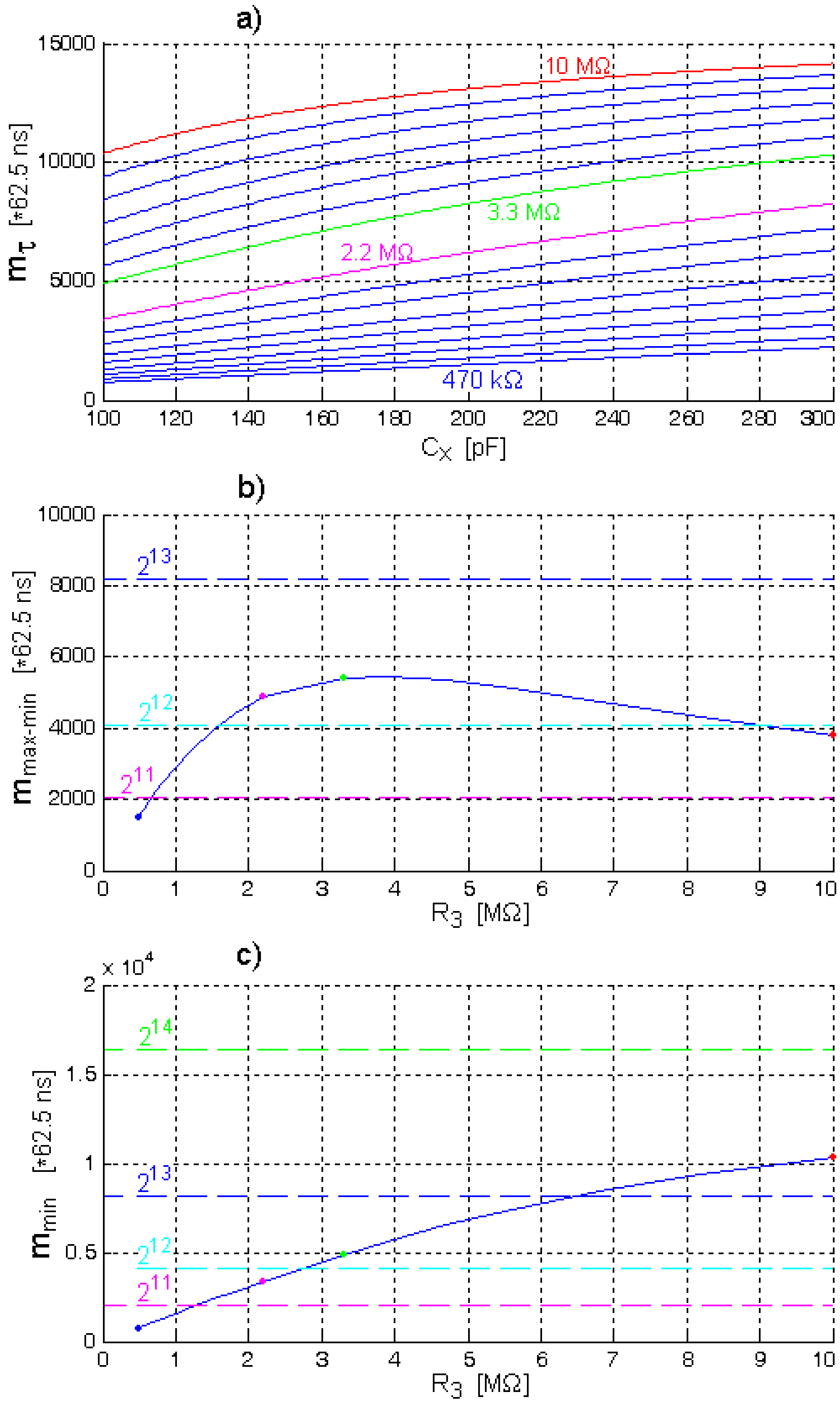

a)
15000
10000
5000
0
m_τ [*62.5 ns]
10 MΩ
3.3 MΩ
2.2 MΩ
470 kΩ
100
120
140
160
180
200
220
240
260
280
300
C_X [pF]
b)
10000
8000
6000
4000
2000
0
m_max-min [*62.5 ns]
2^13
2^12
2^11
0
1
2
3
4
5
6
7
8
9
10
R_3 [MΩ]
c)
x 10^4
2
1.5
1
0.5
0
m_min [*62.5 ns]
2^14
2^13
2^12
2^11
0
1
2
3
4
5
6
7
8
9
10
R_3 [MΩ]

Figure 10.

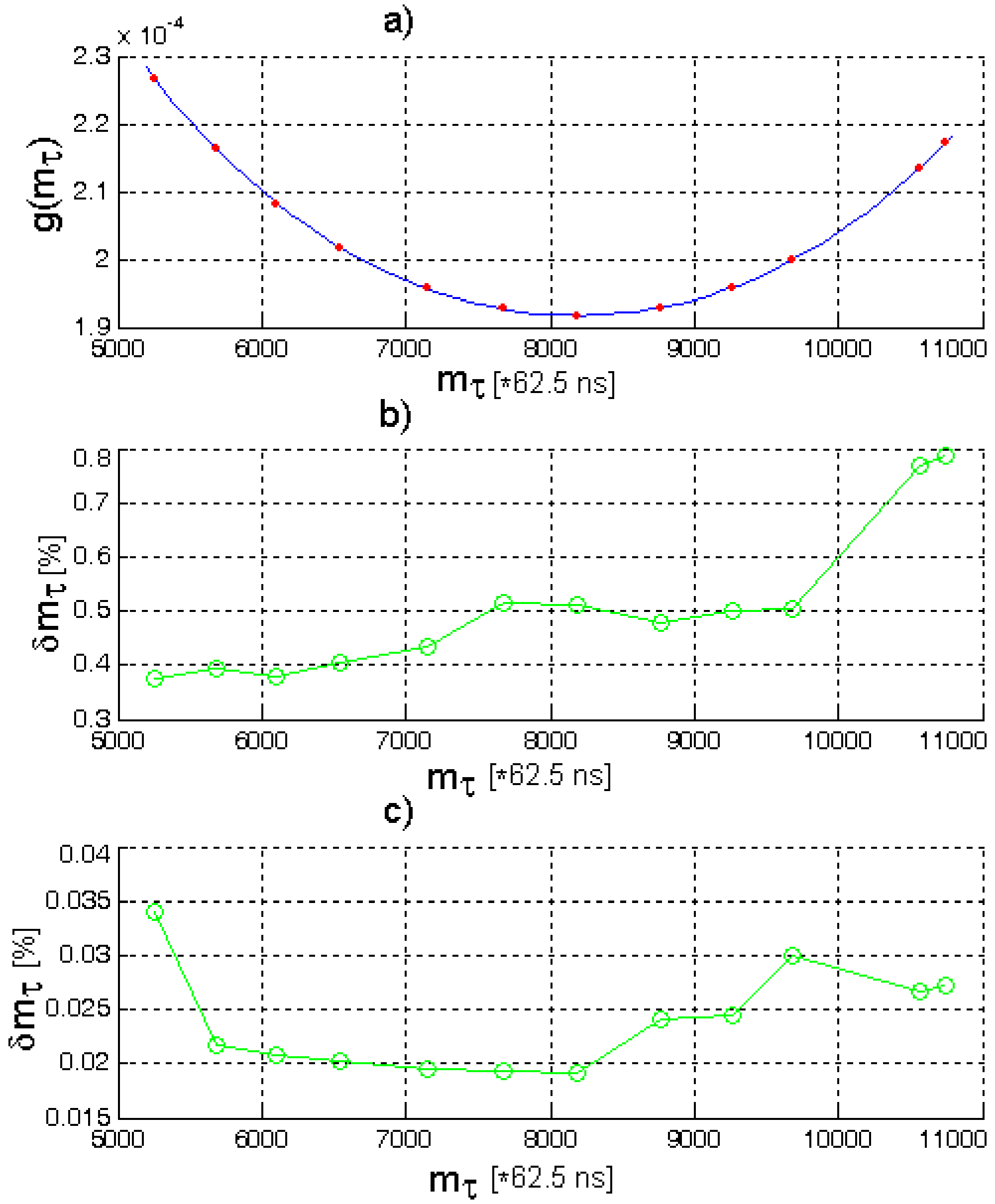

a)
x 10^{-4}
2.3
2.2
2.1
2
1.9
$g(m_\tau)$
5000
6000
7000
8000
9000
10000
11000
m_τ [*62.5 ns]
b)
0.8
0.7
0.6
0.5
0.4
0.3
δm_τ [%]
5000
6000
7000
8000
9000
10000
11000
m_τ [*62.5 ns]
c)
0.04
0.035
0.03
0.025
0.02
0.015
δm_τ [%]
5000
6000
7000
8000
9000
10000
11000
m_τ [*62.5 ns]

Figure 11.

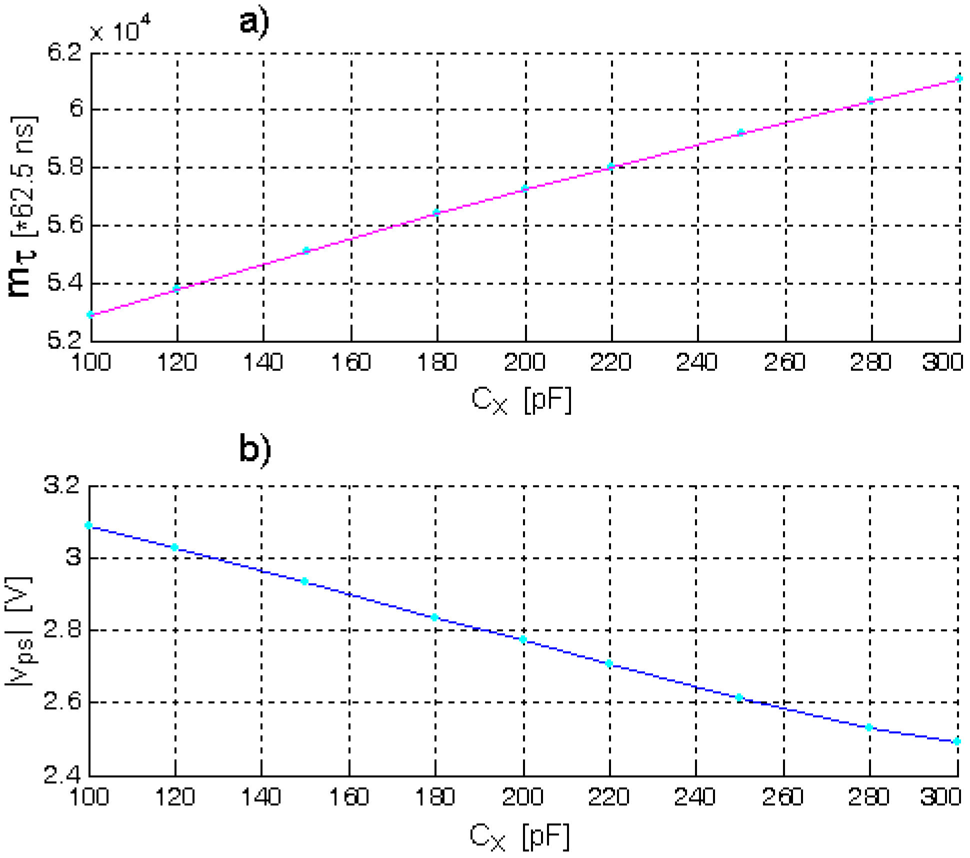

a)
x 10^4
6.2
6
5.8
5.6
5.4
5.2
mτ [*62.5 ns]
100
120
140
160
180
200
220
240
260
280
300
CX [pF]
b)
3.2
3
2.8
2.6
2.4
|vps| [V]
100
120
140
160
180
200
220
240
260
280
300
CX [pF]

Figure 12.

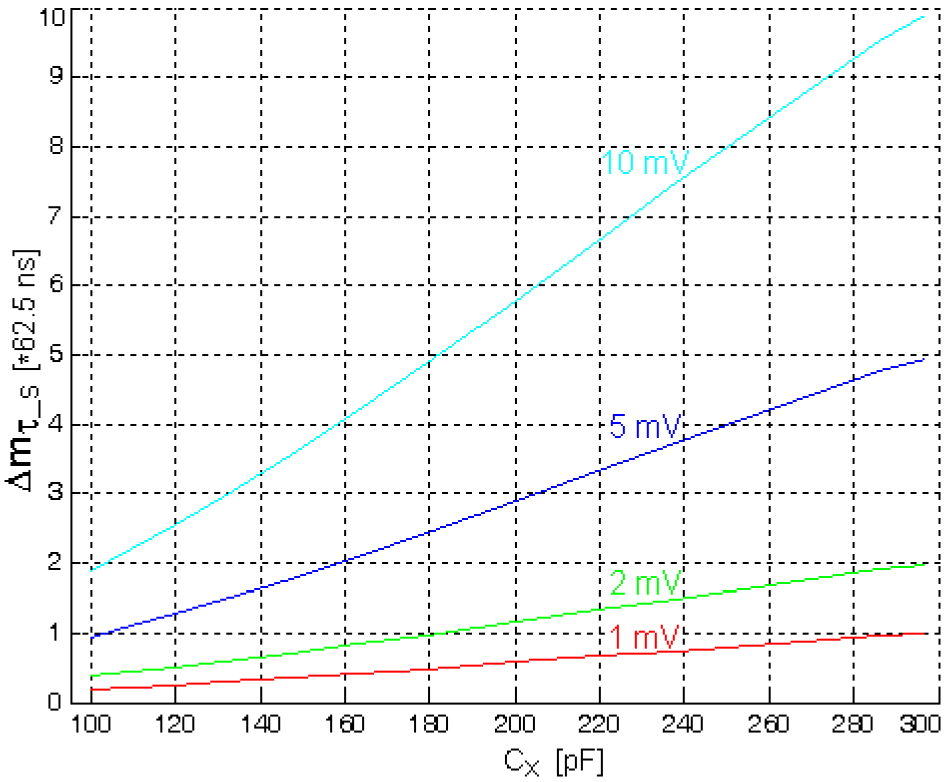

Δm$_{τ\_s}$ [*62.5 ns]
0
1
2
3
4
5
6
7
8
9
10
100
120
140
160
180
200
220
240
260
280
300
C$_X$ [pF]
10 mV
5 mV
2 mV
1 mV

Figure 13.

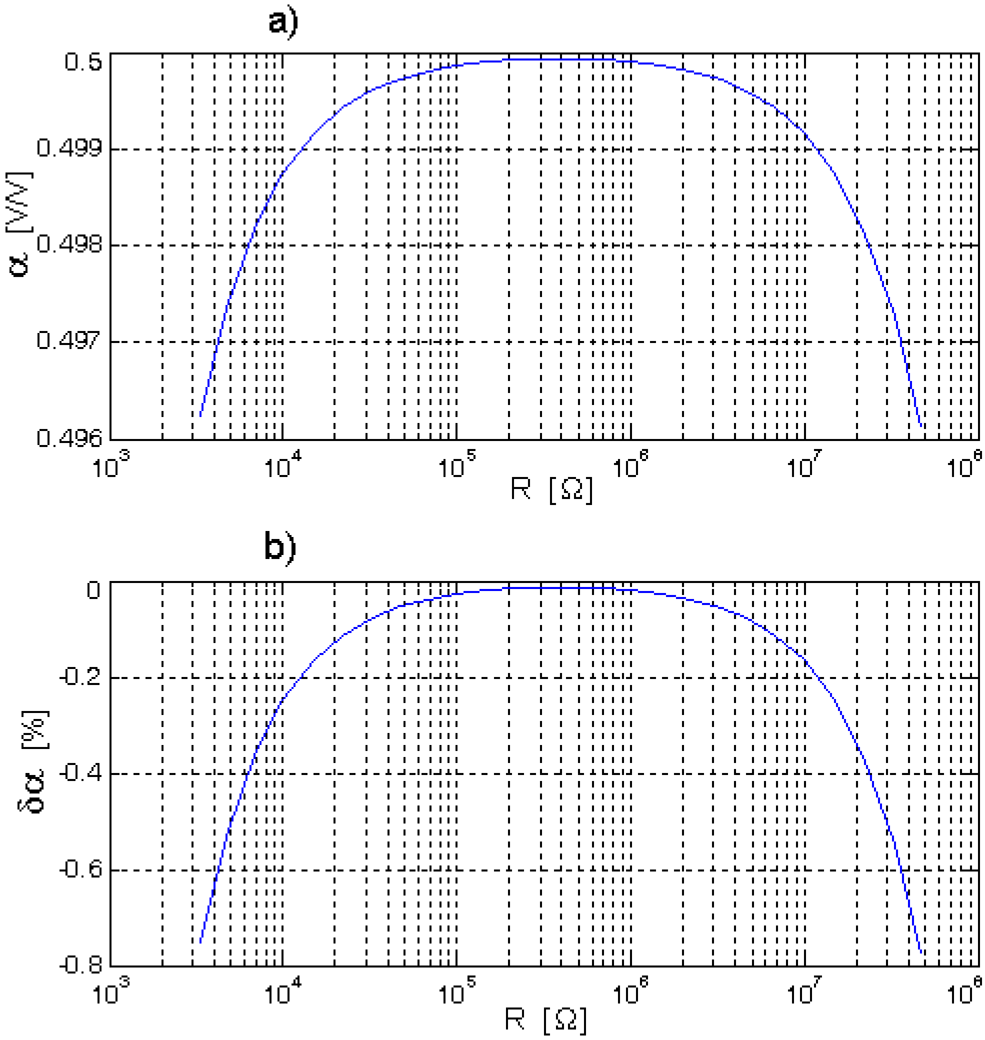

a)
α [V/V]
0.5
0.499
0.498
0.497
0.496
10^3
10^4
10^5
10^6
10^7
10^8
R [Ω]
b)
δα [%]
0
-0.2
-0.4
-0.6
-0.8
10^3
10^4
10^5
10^6
10^7
10^8
R [Ω]

Figure 14.

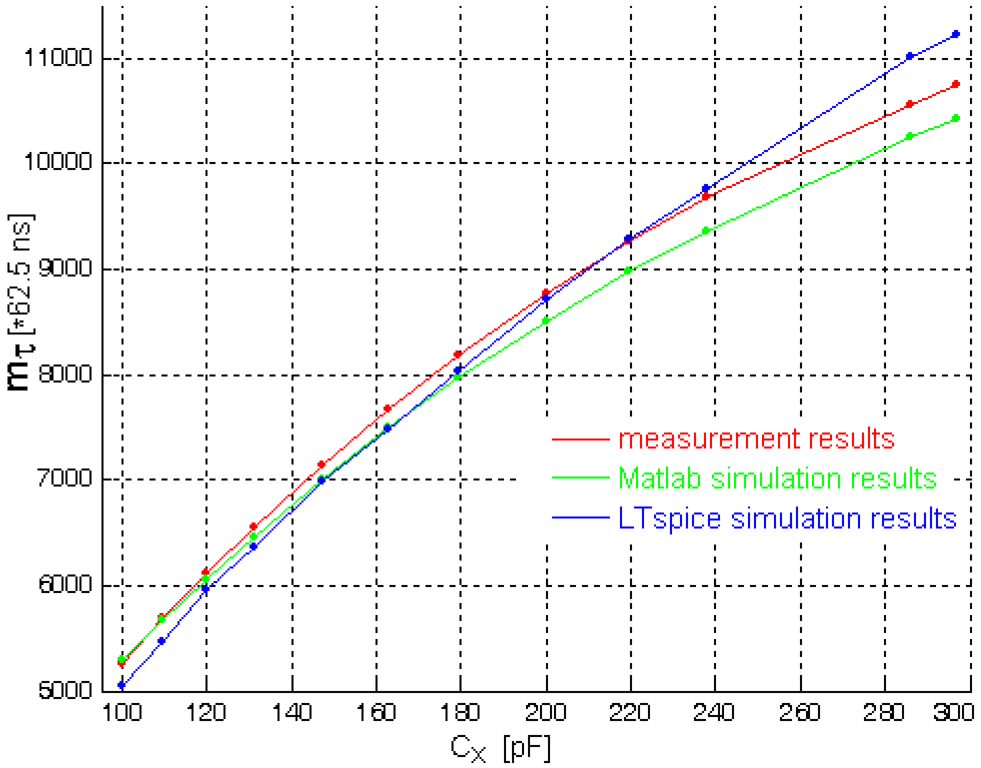

11000
10000
9000
8000
7000
6000
5000
m_τ [*62.5 ns]
100
120
140
160
180
200
220
240
260
280
300
C_X [pF]
measurement results
Matlab simulation results
LTspice simulation results

Figure 15.

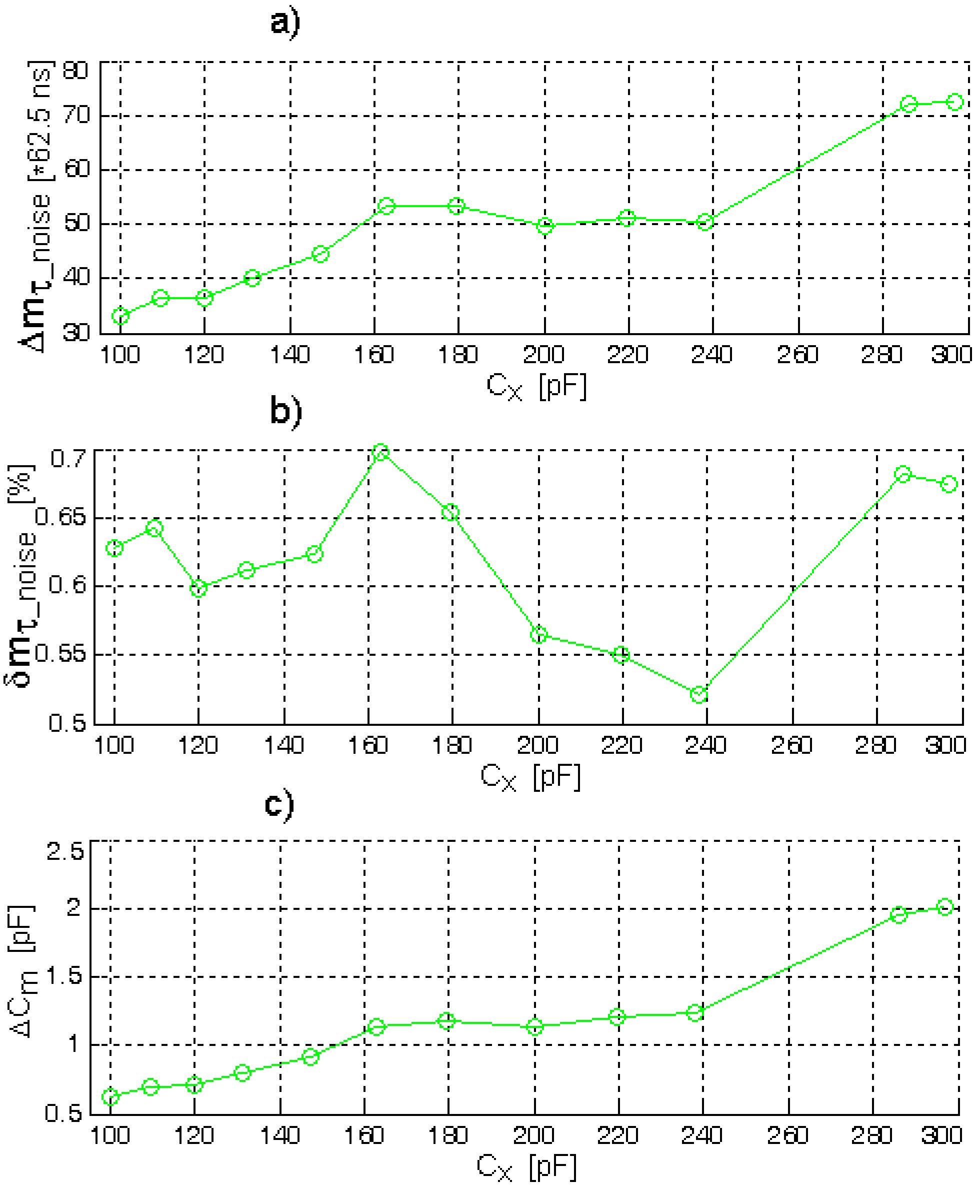

a)
Δmτ_noise [*62.5 ns]
80
70
60
50
40
30
100
120
140
160
180
200
220
240
260
280
300
Cx [pF]
b)
δmτ_noise [%]
0.7
0.65
0.6
0.55
0.5
100
120
140
160
180
200
220
240
260
280
300
Cx [pF]
c)
ΔCm [pF]
2.5
2
1.5
1
0.5
100
120
140
160
180
200
220
240
260
280
300
Cx [pF]

Figure 16.

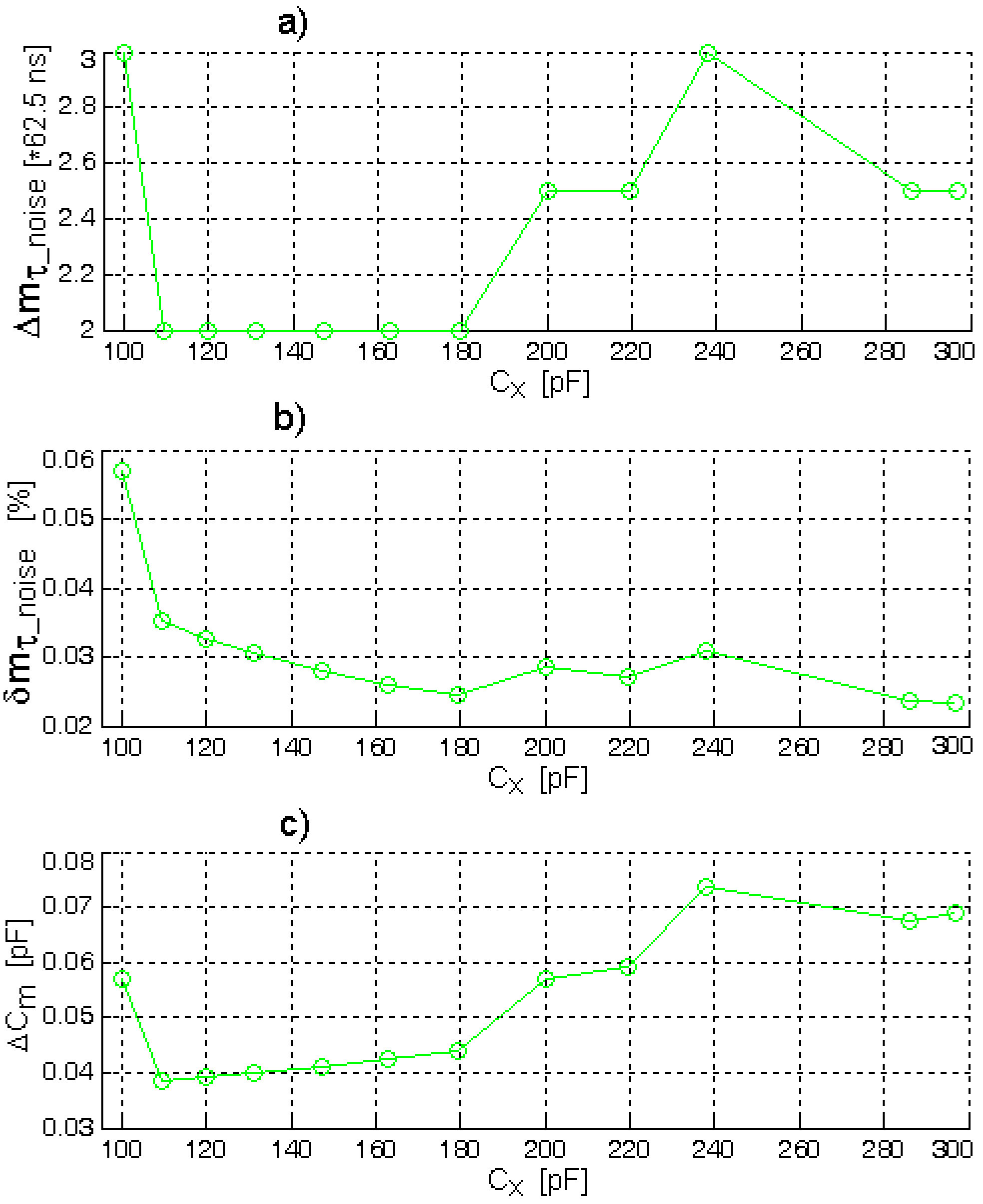
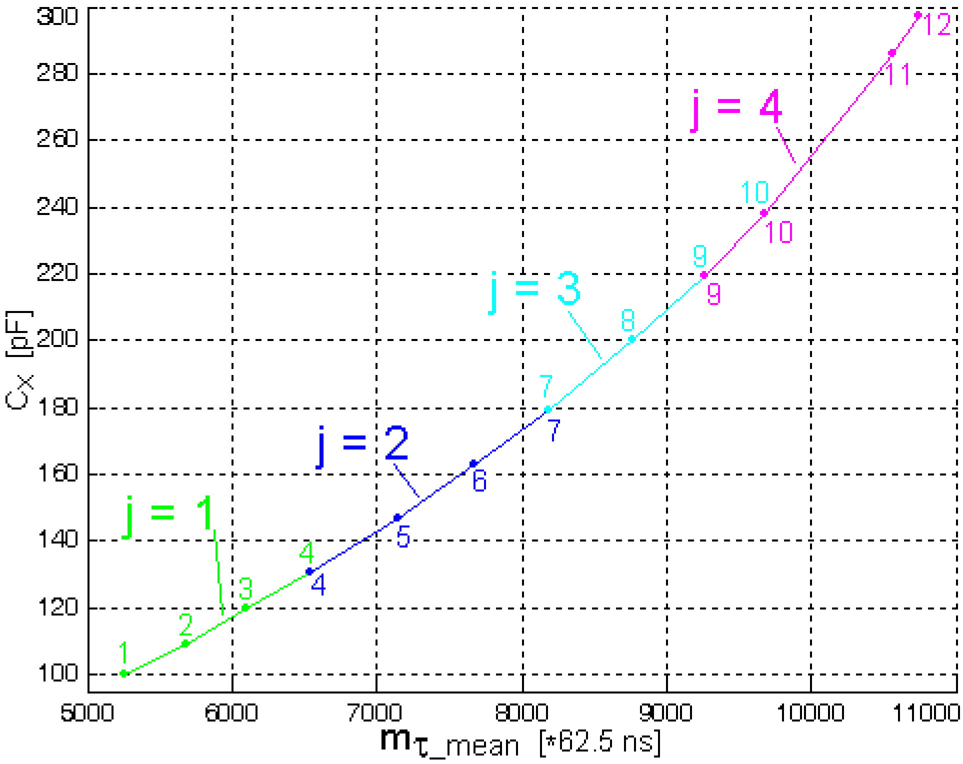

a)
Δmτ_noise [*62.5 ns]
3
2.8
2.6
2.4
2.2
2
100 120 140 160 180 200 220 240 260 280 300
Cx [pF]
b)
δmτ_noise [%]
0.06
0.05
0.04
0.03
0.02
100 120 140 160 180 200 220 240 260 280 300
Cx [pF]
c)
ΔCm [pF]
0.08
0.07
0.06
0.05
0.04
0.03
100 120 140 160 180 200 220 240 260 280 300
Cx [pF]

Figure 17.


j = 1
j = 2
j = 3
j = 4
1
2
3
4
4
5
6
7
7
8
9
9
10
10
11
12
C_x [pF]
100
120
140
160
180
200
220
240
260
280
300
5000
6000
7000
8000
9000
10000
11000
m_{τ}_mean [*62.5 ns]