AN13523

LPC553x/LPC55S3x ADC with Hardware Trigger and ADC Calculator Tool

Rev. 1 — 20 April 2022 Application Note

Contents

1 Introduction

The LPC553x/LPC55S3x MCU family is part of the EdgeVerse edge computing platform and builds on the world's first general-purpose Cortex-M33 based microcontroller introduced with the LPC5500 series. LPC553x/LPC55S3x uses a 16-bit Analog-to-Digital Converter (ADC) which is a dual successive approximation ADC. The ADC allows for differential 16/13-bit resolution and single-ended 16/12-bit resolution operations.

This application note introduces the ADC features that are available with the LPC553x/LPC55S3x device as well as the attached tool to calculate sampling time or source impedance. In addition, an example is provided to demonstrate the hardware triggering capability implemented with the Input Multiplexing (INPUTMUX) module using the CTimer to trigger ADC conversions.

2 ADC features

The ADC module has the following features available on two instances, ADC0 and ADC1:

- Linear successive approximation algorithm
- Differential operation with 16-bit or 13-bit resolution
- Single-ended operation with 16-bit or 12-bit resolution
- Support for two simultaneous single ended conversions.

There is channel support for analog input channels (up to 20 channels) for conversion of external pins and from internal sources. It has a configurable analog input sample time, as well as speed options to accommodate low-power modes. It is capable of having up to 4 trigger sources with different priority.

The ADC module supports three different modes of operation.

Table 1. Different modes of operation

Modes	Description
Run	Normal Operation
Deep-sleep or Sleep	Can continue operating provided the Doze Enable bit (CTRL[DOZEN]) is clear and the module is using an external or internal clock source which remains operating during Deep-sleep/Sleep modes.
Deep Power-down	The Doze Enable (CTRL[DOZEN]) bit is ignored and the module waits for the current transfer to complete any pending operation before acknowledging Deep Power-down mode entry.

3 ADC trigger interconnections

ADC command execution is initiated from up to 4 trigger sources. Each trigger can be software generated by writing 0b1 to the corresponding SWTRIG[SWTn] bit field. Alternatively, hardware triggers can be generated from asynchronous input sources at the periphery of the module. For example, we can use a PWM signal to trigger a conversion periodically. When a hardware trigger input is enabled, hardware trigger events are detected on the rising-edge of the associated hardware trigger source. Each trigger source is assigned a priority via the associated priority control field (TCTRLa[TPRI]). Each of the trigger sources is associated with a command buffer via the associated command select field (TCTRLa[TCMD]).

INPUTMUX provides signal routing options for internal peripherals. Some peripheral inputs are multiplexed to multiple input sources. The sources can be external pins, interrupts, output signals of other peripherals, or other internal signals.

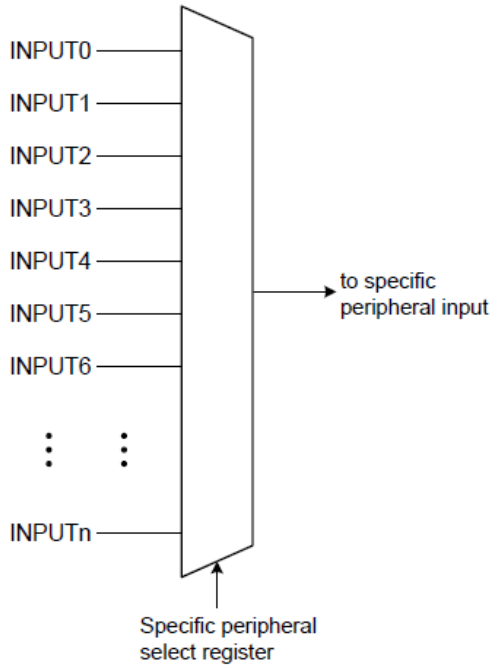

Figure 1. Generic input multiplexing block

The ADC has multiple options for both instances, ADC0 and ADC1, to trigger the analog conversions. The four ADC0 Trigger input connections can be chosen from the list found in the “ADCn Trigger Input Connections ADCn_TRIG0-ADCn_TRIG3” section from the LPC553x/LPC55S3x Reference Manual. This document focuses on using the CTimer as a trigger.

000101 - T0_MAT3

000110 - T1_MAT3

000111 - T2_MAT3

001000 - T3_MAT3

001001 - T4_MAT3

Figure 2. CTimer trigger options for ADC0

4 ADC CTimer example

For this example, the following three modules are set up: ADC, CTimer, and the INPUTMUX. This example is created with the SDK version 2.10.2 using MCUXpresso 11.4.1.

Although it is not necessary to assign an external pin for the CTIMER, in this example, the PIO1_22 is used in order to verify the behavior of the CTIMER. In addition, the ADC0 uses an external pin to measure an external analog source. A simple block diagram of the example is as shown below.

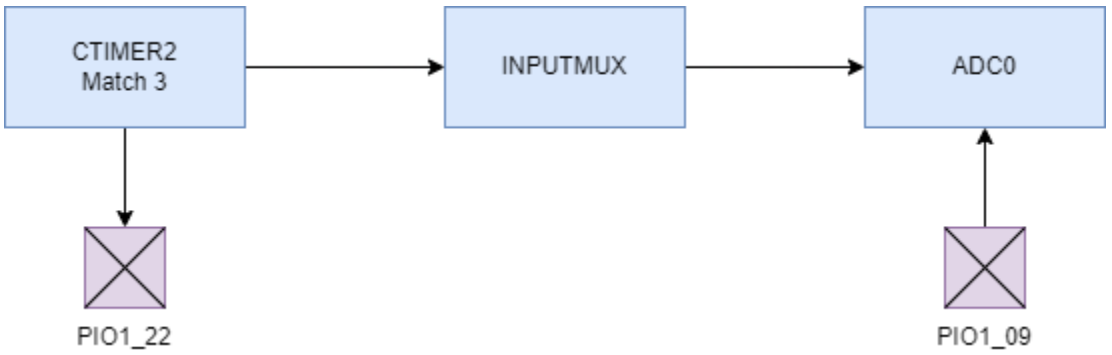

Figure 3. Interconnections for ADC CTimer example

The ADC is configured in the following manner.

ADC input frequency is 5.7 MHz connected from the PLL0 clock which is configured to be at 45.8 MHz. It is recommended to use low frequencies to execute the calibration flow. Select 128 ADC conversions which are averaged to calculate each calibration value. Selecting a higher number of averages leads to more accurate conversions after completing calibration. ADC analog circuits are pre-enabled and ready to execute conversions without startup delays (at the cost of higher DC current consumption). The reference voltage is the voltage on the VDDA pin.

```
/*Attaching low frequency clock from PLL for calibration purposes: ADC0 Frequency at 5.7MHz*/
CLOCK_SetClkDiv(kCLOCK_DivAdc0Clk, 0U, true);           /*!< Reset ADC0CLKDIV divider counter and halt it */
CLOCK_SetClkDiv(kCLOCK_DivAdc0Clk, 8U, false);          /*!< Set ADC0CLKDIV divider to value 8 */
CLOCK_AttachClk(kPLL0_to_ADC0);                /*!< Switch ADC0 to PLL0 */

/* Disable VREF power down */
POWER_DisablePD(kPDRUNCFG_PD_VREF);

LPADC_GetDefaultConfig(&mLpadcConfigStruct);
mLpadcConfigStruct.enableAnalogPreliminary = true;
mLpadcConfigStruct.referenceVoltageSource = kLPADC_ReferenceVoltageAlt3;
mLpadcConfigStruct.conversionAverageMode = kLPADC_ConversionAverage128;
LPADC_Init(DEMO_LPADC_BASE, &mLpadcConfigStruct);
```

Figure 4. ADC configuration

Once the auto-calibration is finished, you can choose a higher frequency clock for the ADC. In this example, we use 48 MHz. The command configuration is set for channel 0 associated to the A-side using a high-resolution conversion. This selects analog signal ADC0IN0A for conversion, available on pin PIO1_9. Each ADC command independently makes a channel and conversion type selection. In this example, single ended operation is chosen. However, it is possible to convert to differential mode but only limited pairs are intended to be set as differential channels. Refer to the LPC553x/LPC55S3x Reference Manual for the available pin pairings. Additionally, the trigger configuration is set to hardware trigger since, we are using another module's signal to trigger the ADC conversions.

```
/* Set conversion CMD configuration. */
LPADC_GetDefaultConvCommandConfig(&mLpadcCommandConfigStruct);
mLpadcCommandConfigStruct.channelNumber = 0U;
mLpadcCommandConfigStruct.sampleChannelMode = kLPADC_SampleChannelSingleEndSideA;
mLpadcCommandConfigStruct.conversionResolutionMode = kLPADC_ConversionResolutionHigh;

LPADC_SetConvCommandConfig(DEMO_LPADC_BASE, DEMO_LPADC_USER_CMDID, &mLpadcCommandConfigStruct);

/* Set trigger configuration. */
LPADC_GetDefaultConvTriggerConfig(&mLpadcTriggerConfigStruct);
mLpadcTriggerConfigStruct.targetCommandId       = 1U;
mLpadcTriggerConfigStruct.enableHardwareTrigger = true;
LPADC_SetConvTriggerConfig(DEMO_LPADC_BASE, 0U, &mLpadcTriggerConfigStruct); /* Configurate the trigger0. */
```

Figure 5. LPADC CMD and trigger configuration

The CTimer is configured in the following manner:

CTimer input frequency is 96 MHz. From the trigger options that are available for the ADC0 instance, we can configure any CTimer0 - CTimer4 Match 3 signal, in this case CTimer 2 Match 3 is used. It is configured as timer mode and increments on every APB bus clock. The counter is reset after every match and toggles the output at a frequency of 1 kHz. It is not necessary to enable the interrupt for the CTimer, since we are attaching the signal regardless to the ADC trigger. However, if it is necessary to change the match value or other setting of the CTimer, you may enable an interrupt for additional actions that may be required.

```
/* Use 96MHz clock for CTimer2*/
CLOCK_SetClkDiv(kCLOCK_DivCtimer2Clk, 0u, false);
CLOCK_SetClkDiv(kCLOCK_DivCtimer2Clk, 1u, true);
CLOCK_AttachClk(kFRO_HF_to_CTIMER2);

CTIMER_GetDefaultConfig(&config);

CTIMER_Init(CTIMER, &config);

/* Configuration 0 */
matchConfig0.enableCounterReset = true;
matchConfig0.enableCounterStop  = false;
matchConfig0.matchValue         = CTIMER_CLK_FREQ / 2000;
matchConfig0.outControl         = kCTIMER_Output_Toggle;
matchConfig0.outPinInitState    = false;
matchConfig0.enableInterrupt    = false;

CTIMER_SetupMatch(CTIMER, CTIMER_MAT3_OUT, &matchConfig0);
CTIMER_StartTimer(CTIMER);
```

Figure 6. CTimer configuration

The INPUTMUX is configured in the following manner:

The INPUTMUX module needs to be initialized prior to attaching the appropriate signals. In this case, we are using the CTIMER2 Match 3 signal to trigger the ADC0 instance.

```
INPUTMUX_Init(INPUTMUX);
INPUTMUX_AttachSignal(INPUTMUX, 0U, kINPUTMUX_Ctimer2M3ToAdc0Trigger);
```

Figure 7. INPUTMUX configuration

By checking the CTimer external pin (PIO1_22, at header J10 pin 9 on LPC55S36-EVK), it is possible to verify that the signal is effectively working as expected. It is toggling at a rate of 1 kHz. As discussed previously, each ADC conversion is triggered on the

rising edge of the CTimer waveform that is seen below. The three blue arrows in the image below represent rising edges where the ADC0 is being triggered.

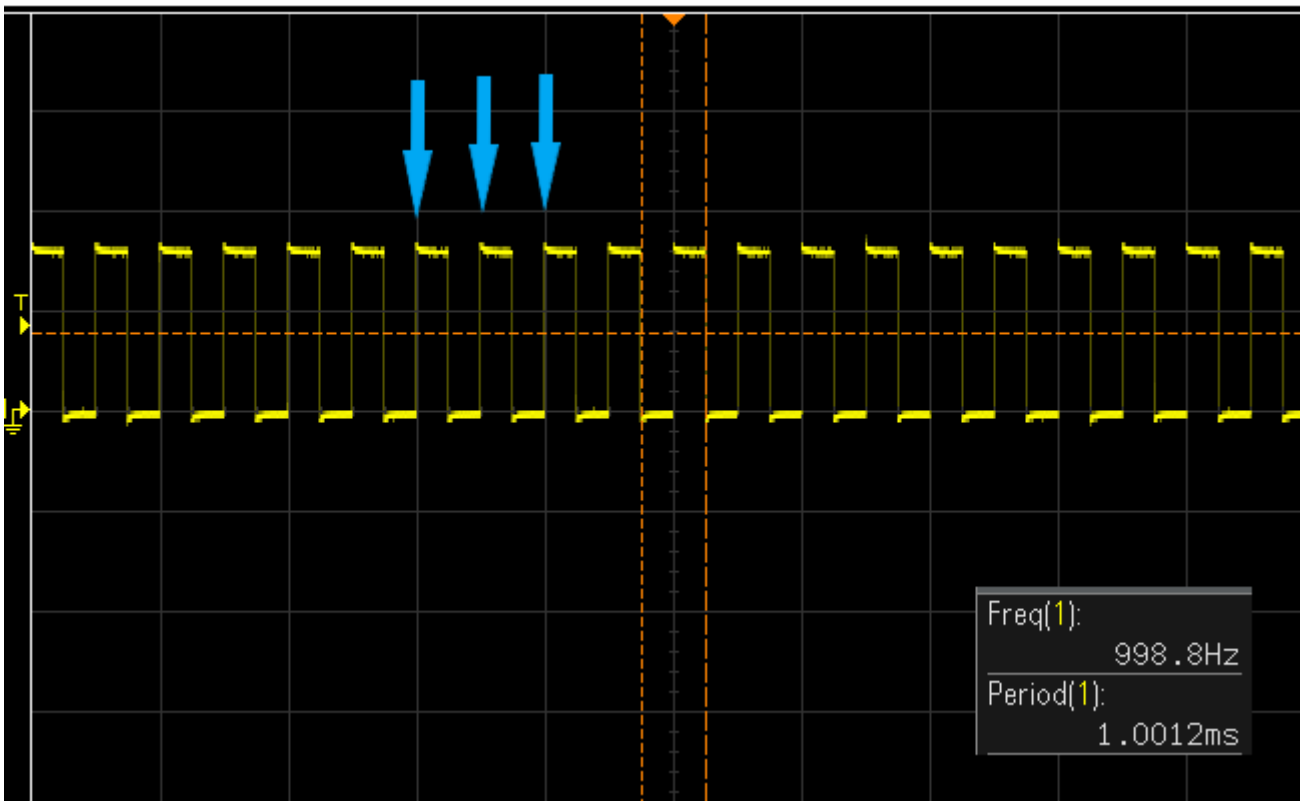

Figure 8. CTimer match rate shown through external pin

In the image below, an example is shown of the ADC result when measuring 3.3 V. We should expect the maximum value printed in our terminal window for the 16-bit resolution, in this case 65535. To change the converted value, change the voltage on pin PIO1_9, on header J7 pin 1 on LPC55S36-EVK.

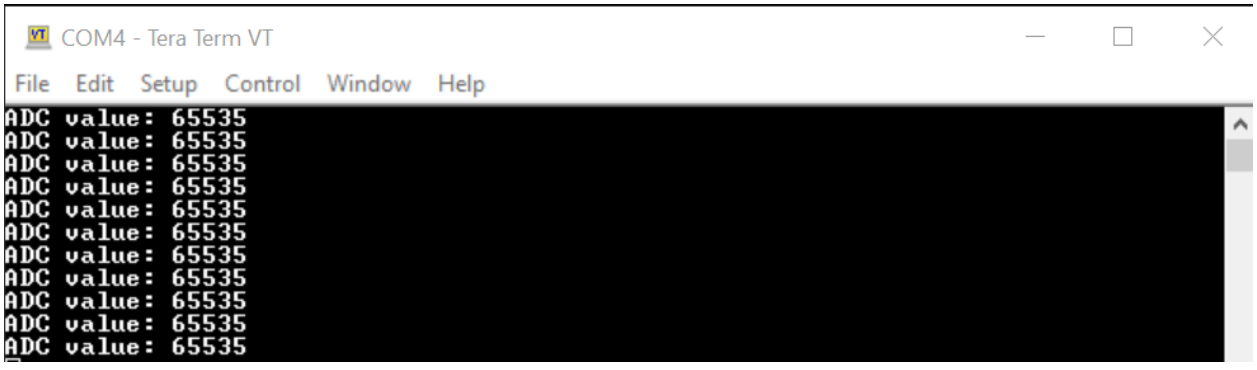

Figure 9. Printed results shown from measuring 3.3 V source

5 ADC calculation tool

The objective of the tool attached to this application note is to define max sampling rates that can be achieved depending on the input signal impedance characteristics. In order to sample the input voltage accurately, the source resistance, and ADC sample time must be chosen appropriately. For a fixed source resistance (R_{AS}), the required sample time is given by:

$$\min t_{SMP} = B \times [R_{AS} \times (C_{AS} + C_P + C_{ia}) + C_{ia} \times (R_{AS} + R_I)]$$

$$B = -\ln\left(\frac{LSB_{ERR}}{2^N}\right)$$

N – ADC resolution, that is, 12 for 13-bit and 12-bit mode, 16 for 16-bit mode.

LSB_{ERR} -- value of acceptable sampling error in LSBs, that is, sampling within ¼ LSB accuracy.

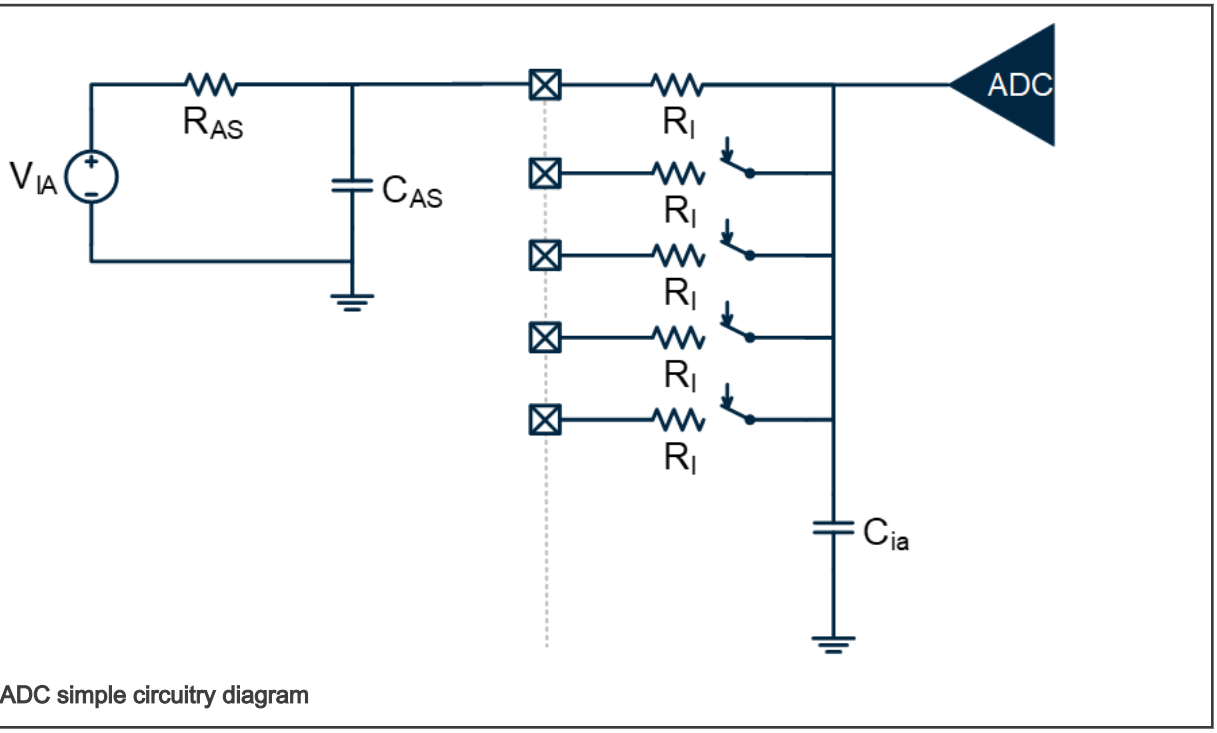

Figure 10. ADC simple circuitry diagram

Where B is the adjusted resolution based on the chosen sampling error. The user configured sample time is determined by the ADC input clock frequency (F_{ADCK}) and the Sample Time Select (STS) bits in the ADC command register which choose the number of the sample cycles. When STS is programmed to a non-zero value, the sample time is (3 + 2^{STS}) ADCK cycles. The shortest sample time maximizes conversion speed for lower impedance inputs. Extending sample time allows higher impedance inputs to be accurately sampled:

$$user\ t_{SMP} = \frac{user\ STS}{f_{ADCK}}$$

User $_{STS}$ is the number of ADC clock cycles during the sample time, and is programmable 3, 5, 7, 11, 19, 35, 67 or 131 ADCK cycles (user $_{STS}$ ≥ min $_{STS}$), it depends on the value chosen at register CMDHn[STS]. User t_{SMP} must be configured to be greater than or equal to min t_{SMP}. If we set user t_{SMP} > min t_{SMP} and solve for R_{AS}, we can find the maximum source resistance that would allow us to sample at the desired accuracy:

$$R_{AS} < \frac{\frac{user\ STS}{f_{ADCK} x B} - (R_I x C_{ia})}{C_{AS} + C_P + 2C_{ia}}$$

Table 2. ADC sample time calculator

ADC min Sample Time t_{SMP}		
min sample time based on a fixed R_{AS}		
Input... enter yellow highlighted cells		
Resolution	16	bits
C_{AS}	100E-12	F
R_{AS}	5.0E+3	Ω
LSB_{ERR}	1/8	LSB
C_P	3E-12	F
Cia	4E-12	F
B	13.17	

	R_I(Ω)	min t_{SMP}
High Speed		
V_{REFP} = 1.8V	1.6E+3	7.39E-6
V_{REFP} = 3.0V	1.1E+3	7.37E-6
Standard Muxed		
V_{REFP} = 1.8V	3.2E+3	7.48E-6
V_{REFP} = 3.0V	1.8E+3	7.40E-6
Standard Dedicated		
V_{REFP} = 1.8V	300	7.33E-6
V_{REFP} = 3.0V	300	7.33E-6
Internal		
V_{REFP} = 1.8V	1.9E+3	7.41E-6
V_{REFP} = 3.0V	1.1E+3	7.37E-6

The first section of this tool, is as shown on the left. This table specifies the required sample time for a fixed R_{AS}.

The user can input the source resistance, source capacitance (both external components), the resolution, and the value of the acceptable sampling error in LSB (LSB_{ERR}) of the ADC.

The following values provided in the LPC553x/LPC55S3x Data Sheet.

C_P – Parasitic Cap of pad/package

C_{ia} – Input Capacitance

R_I – Input Resistance

Table 3. ADC sampling frequency and R_{AS} calculator

ADC Conversion Time CT and R_{AS} Calculator						
To calculate maximum R_{AS} with given sample time and ADC input clock frequency						
Input... enter yellow highlighted cells						
Resolution	12	bits	Sample Time Select	131	ADCK	
C_{AS}	1E-9	F	Hardware Average	1		
ADC Source CLK	100E+6	Hz	Power Select	high power		
ADCnCLKDIV[DIV]	3					
LSB_{ERR}	1/2	LSB				
C_P	3E-12	F				
C_{ia}	4E-12	F				
B	9.01					
f_{ADCK}	33E+6	MHz	max f_{ADCK}	60.0E+6	MHz	

	R_I (Ω)	min STS	user STS	user t_{SMP}	max R_{AS}	CT	Msps
High Speed							
V_{REFP} = 1.8V	1.6E+3	5	131	3.9E-6	426.7E+0	4.4E-6	0.23
V_{REFP} = 3.0V	1.1E+3	5	131	3.9E-6	428.7E+0	4.4E-6	0.23
Standard Muxed							
V_{REFP} = 1.8V	3.2E+3	5	131	3.9E-6	420.4E+0	4.4E-6	0.23
V_{REFP} = 3.0V	1.8E+3	5	131	3.9E-6	425.9E+0	4.4E-6	0.23
Standard							
V_{REFP} = 1.8V	300	5	131	3.9E-6	431.9E+0	4.4E-6	0.23
V_{REFP} = 3.0V	300	5	131	3.9E-6	431.9E+0	4.4E-6	0.23
Internal							
V_{REFP} = 1.8V	1.9E+3	5	131	3.9E-6	425.6E+0	4.4E-6	0.23
V_{REFP} = 3.0V	1.1E+3	5	131	3.9E-6	428.7E+0	4.4E-6	0.23

The next section of the tool, can provide the maximum source resistance based on the sample time and ADC frequency that is used.

As in the previous section, the user can change the yellow cells according to the parameters that they are working with.

The following are some other parameters that are given in the calculator tool.

F_{ADCK} - Input clock frequency

CYC_{SMP_MIN} - Min sample cycles required for $T_{SMP} > T_{SMP\_REQ}$

CYC_{SMP_USER} – Sample cycles set the user using CMDHn[STS]

T_{SMP} – Sample time set the user.

6 ADC basic concepts

Resolution: The number of bits in the ADC digital output representing an analog input signal. For LPC553x/LPC55S3x, the resolution can be configured to 12, 13 and 16-bit resolution.

Reference Voltage: The ADC requires a reference voltage used to create a successive approximation comparison with the analog input to produce a digital output. The digital output is the ratio of the analog input with respect to this reference voltage.

$V_{REF} = V_{REFH} – V_{REFL}$

Where: V_{REFH} = High reference voltage V_{REFL} = Low reference voltage

ADC output formula: The conversion equation of ADC is used to calculate the digital output corresponding to a particular analog input voltage. This equation assumes an ideal A/D conversion with no introduced errors.

$$ADC\ \ Digital\ \ Output = \frac{2^N * Analog\ \ Input\ \ Voltage}{Reference\ \ Voltage}$$

$$N = ADC\ \ Resolution.\ For\ \ the\ \ LPC553x\ \ this\ \ can\ \ be\ 12/13/16$$

Least Significant Bits (LSB): A least significant bit (LSB) is a unit of voltage equal to the smallest resolution of the ADC, that is, the smallest incremental voltage that causes a change in the digital output. The LSB is equal to the reference voltage divided by the maximum count of the ADC:

$LSB = V_{REF}/2^N$

N = ADC resolution

V_{REF} = Analog reference voltage

ADC Actual Transfer Function: The ADC converts an input voltage to a corresponding digital code. The curve describing this behavior is the actual transfer function and includes all the errors inherent to the ADC module itself.

ADC Ideal Transfer Function: The ideal transfer function represents the behavior of the ADC assuming it is perfectly linear, or that a given change in input voltage creates the same change in conversion code regardless of the input's initial level.

7 Sources of error in ADC measurements

This section presents some typical factors that prevent the ADC from performing accurate A/D measurements.

Reference voltage noise: The ADC output is directly proportional to the analog input voltage and the reference voltage. An unstable reference voltage (for example, caused by noise in the supply rail) causes the changes in the converted digital outputs. For example, a reference voltage of 5 V and an input voltage of 1 V gives 819 for a 12-bit resolution using the ADC output formula. With a 50 mV increase in the absolute reference voltage (that is, V_{REF} = 5.05 V), the new converted value for the same 1 V input voltage is now 811. The resulting reference voltage noise error is 811-819 = - 8 LSB.

Analog input signal noise: Small but high-frequency variations in the analog input signal can potentially cause big conversion errors during ADC sampling time. Noise can be induced by electromagnetic emissions from surrounding electrical devices (EMI noise). Therefore, the conversion accuracy is negatively impacted. If the noise present in the input signal is higher than 1LSB, this effectively reduces the number of reliable bits in the conversion result, since the least significant bits are constantly changing due to the signal variations.

Analog-signal source resistance: The impedance of the analog signal source or series resistance (RIN) between the source and the input pin causes a voltage drop across it because of the current flowing into the pin.

Temperature influence: The temperature of the system can have a major influence on ADC accuracy, mainly causing offset error drift and gain error drift. The ADC reference voltage also changes with temperature change. These errors can be compensated with adjustments to the microcontroller firmware, such as monitoring the internal bandgap voltage to verify that the reference voltage has not changed or characterizing the system over the application's temperature range to account for the errors.

8 References

- LPC553x/LPC55S3x Reference Manual

- LPC553x/LPC55S3x Data Sheet
- LPC553x/LPC55S3x Errata
- Hardware Design Guidelines for LPC55(S)xx Microcontrollers (document AN13033)

9 Revision history

Table 4 is the revision history table.

Table 4. Revision history

Revision number	Date	Substantive changes
1	20 April 2022	Updated ADC calculation tool
0	2 February 2022	Initial release

Legal information

Definitions

Draft — A draft status on a document indicates that the content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included in a draft version of a document and shall have no liability for the consequences of use of such information.

Disclaimers

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors.

In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory.

Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms and conditions of commercial sale of NXP Semiconductors.

Right to make changes — NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Suitability for use — NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in life support, life-critical or safety-critical systems or equipment, nor in applications where failure or malfunction of an NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors and its suppliers accept no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products.

NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

Terms and conditions of commercial sale — NXP Semiconductors products are sold subject to the general terms and conditions of commercial sale, as published at http://www.nxp.com/profile/terms, unless otherwise agreed in a valid written individual agreement. In case an individual agreement is concluded only the terms and conditions of the respective agreement shall apply. NXP Semiconductors hereby expressly objects to applying the customer's general terms and conditions with regard to the purchase of NXP Semiconductors products by customer.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from competent authorities.

Suitability for use in non-automotive qualified products — Unless this data sheet expressly states that this specific NXP Semiconductors product is automotive qualified, the product is not suitable for automotive use. It is neither qualified nor tested in accordance with automotive testing or application requirements. NXP Semiconductors accepts no liability for inclusion and/or use of non-automotive qualified products in automotive equipment or applications.

In the event that customer uses the product for design-in and use in automotive applications to automotive specifications and standards, customer (a) shall use the product without NXP Semiconductors' warranty of the product for such automotive applications, use and specifications, and (b) whenever customer uses the product for automotive applications beyond NXP Semiconductors' specifications such use shall be solely at customer's own risk, and (c) customer fully indemnifies NXP Semiconductors for any liability, damages or failed product claims resulting from customer design and use of the product for automotive applications beyond NXP Semiconductors' standard warranty and NXP Semiconductors' product specifications.

Translations — A non-English (translated) version of a document, including the legal information in that document, is for reference only. The English version shall prevail in case of any discrepancy between the translated and English versions.

Security — Customer understands that all NXP products may be subject to unidentified vulnerabilities or may support established security standards or specifications with known limitations. Customer is responsible for the design and operation of its applications and products throughout their lifecycles to reduce the effect of these vulnerabilities on customer's applications and products. Customer's responsibility also extends to other open and/or proprietary technologies supported by NXP products for use in customer's applications. NXP accepts no liability for any vulnerability. Customer should regularly check security updates from NXP and follow up appropriately.

Customer shall select products with security features that best meet rules, regulations, and standards of the intended application and make the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, and security related requirements concerning its products, regardless of any information or support that may be provided by NXP.

NXP has a Product Security Incident Response Team (PSIRT) (reachable at PSIRT@nxp.com) that manages the investigation, reporting, and solution release to security vulnerabilities of NXP products.

Trademarks

Notice: All referenced brands, product names, service names, and trademarks are the property of their respective owners.

NXP — wordmark and logo are trademarks of NXP B.V.

AMBA, Arm, Arm7, Arm7TDMI, Arm9, Arm11, Artisan, big.LITTLE, Cordio, CoreLink, CoreSight, Cortex, DesignStart, DynamIQ, Jazelle, Keil, Mali, Mbed, Mbed Enabled, NEON, POP, RealView, SecurCore, Socrates, Thumb, TrustZone, ULINK, ULINK2, ULINK-ME, ULINK-PLUS, ULINKpro, µVision, Versatile — are trademarks or registered trademarks of Arm Limited (or its subsidiaries) in the US and/or elsewhere. The related technology may be protected by any or all of patents, copyrights, designs and trade secrets. All rights reserved.

Airfast — is a trademark of NXP B.V.

Bluetooth — the Bluetooth wordmark and logos are registered trademarks owned by Bluetooth SIG, Inc. and any use of such marks by NXP Semiconductors is under license.

Cadence — the Cadence logo, and the other Cadence marks found at www.cadence.com/go/trademarks are trademarks or registered trademarks of Cadence Design Systems, Inc. All rights reserved worldwide.

CodeWarrior — is a trademark of NXP B.V.

ColdFire — is a trademark of NXP B.V.

ColdFire+ — is a trademark of NXP B.V.

EdgeLock — is a trademark of NXP B.V.

EdgeScale — is a trademark of NXP B.V.

EdgeVerse — is a trademark of NXP B.V.

eIQ — is a trademark of NXP B.V.

FeliCa — is a trademark of Sony Corporation.

Freescale — is a trademark of NXP B.V.

HITAG — is a trademark of NXP B.V.

ICODE and I-CODE — are trademarks of NXP B.V.

Immersiv3D — is a trademark of NXP B.V.

I2C-bus — logo is a trademark of NXP B.V.

Kinetis — is a trademark of NXP B.V.

Layerscape — is a trademark of NXP B.V.

Mantis — is a trademark of NXP B.V.

MIFARE — is a trademark of NXP B.V.

MOBILEGT — is a trademark of NXP B.V.

NTAG — is a trademark of NXP B.V.

Processor Expert — is a trademark of NXP B.V.

QorIQ — is a trademark of NXP B.V.

SafeAssure — is a trademark of NXP B.V.

SafeAssure — logo is a trademark of NXP B.V.

StarCore — is a trademark of NXP B.V.

Synopsys — Portions Copyright © 2021 Synopsys, Inc. Used with permission. All rights reserved.

Tower — is a trademark of NXP B.V.

UCODE — is a trademark of NXP B.V.

VortiQa — is a trademark of NXP B.V.

For more information, please visit: http://www.nxp.com
For sales office addresses, please send an email to: salesaddresses@nxp.com

Date of release: 20 April 2022
Document identifier: AN13523