AN14679

MCX A3xx ADC Performance Benchmarking

Rev. 1.0 — 15 July 2025 Application note

Document information

Information	Content
Keywords	AN14679, MCX A3xx, 12/16-bit ADC, SAR ADC, ADC static performance, ADC dynamic performance
Abstract	This document describes the SAR ADC workflow and the static/dynamic performance of MCX A3xx ADC. It also describes how to use the MCX A3xx ADC.

1 Introduction

This document introduces the MCX A3xx Analog-to-Digital Converter (ADC) IP features and provides the ADC performance parameter test methods, such as:

- Static performance parameter DNL/INL
- Dynamic performance parameters SNR, SINAD, THD, and ENOB

The document also provides a tool for calculating the MCX A3xx ADC limitation parameters, such as conversion rate and ADC input impedance value.

The Arm Cortex-M33 based NXP MCX A3xx general purpose MCUs operate at up to 180 MHz with high levels of integration and analog functionality. They offer a wide range of low-power and intelligent peripherals, including four 12/16-bit ADCs with 4 MS/s sampling rate and hardware averaging features.

2 Analog-to-Digital (ADC) fundamentals

Successive Approximation Register (SAR) ADC is an analog-to-digital peripheral, which can achieve high-speed analog signal sample. Usually, it contains sample and hold (S/H) circuit, capacitive DAC, comparator, and SAR logic units. Figure 1 shows the block diagram of an N-bit differential SAR ADC.

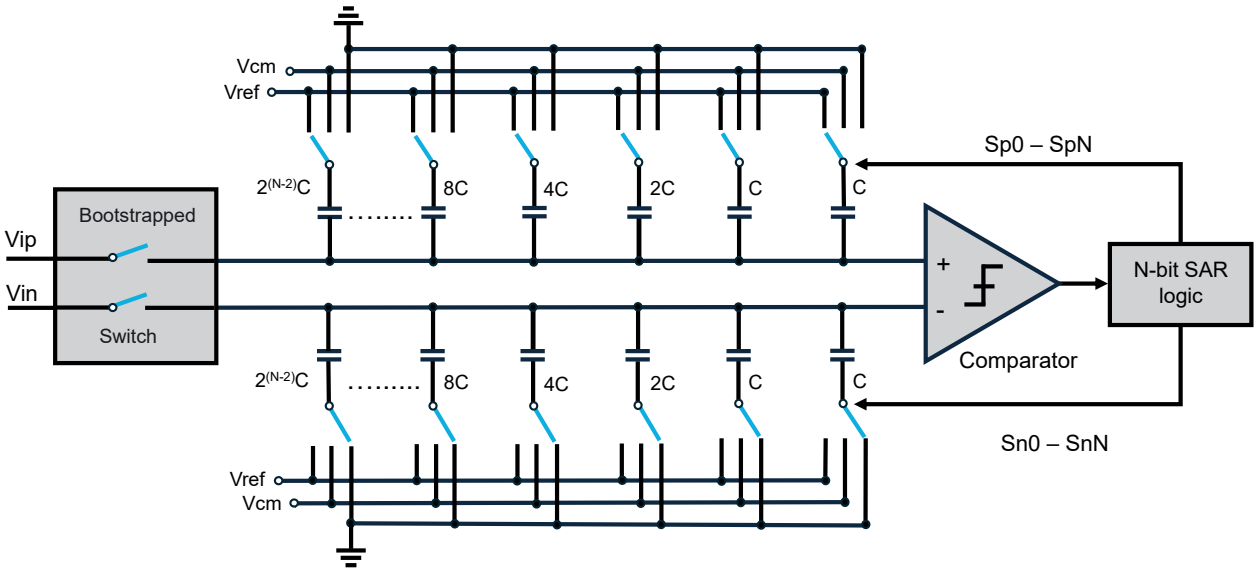

Figure 1. N-bit differential SAR ADC diagram

In an MCX A series MCU, the ADC is a single-ended ADC. Figure 2 shows the block diagram of a single-ended N-bit SAR ADC.

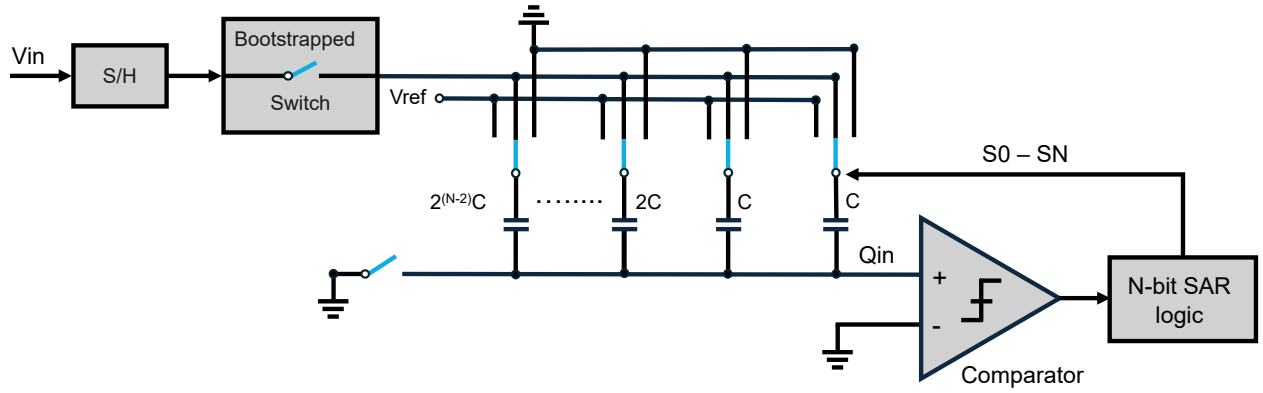

Figure 2. N-bit single-ended SAR ADC diagram

MCX A3xx integrates four 12/16-bit single-ended SAR ADCs with sample rate up to 4 megasamples per second (MS/s) in 12-bit mode. Figure 3 shows the block diagram of MCX A series ADC.

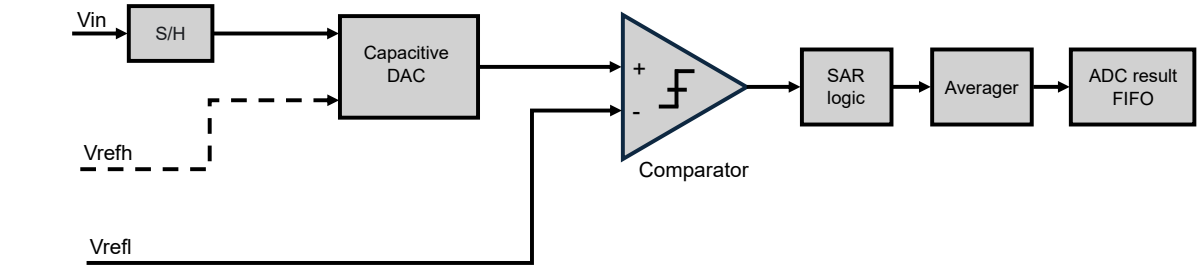

Figure 3. MCX A series SAR ADC diagram

3 Single-ended SAR ADC workflow

Figure 4 shows a 3-bit single-ended SAR ADC sample and hold state. When the bootstrapped switch and Sg are closed, and S0 – S3 are connected to Vin; the capacitors connected to S0 – S3 are charged. The quantity of electric charge is:

$$Qin = -Vin*8C \quad (1)$$

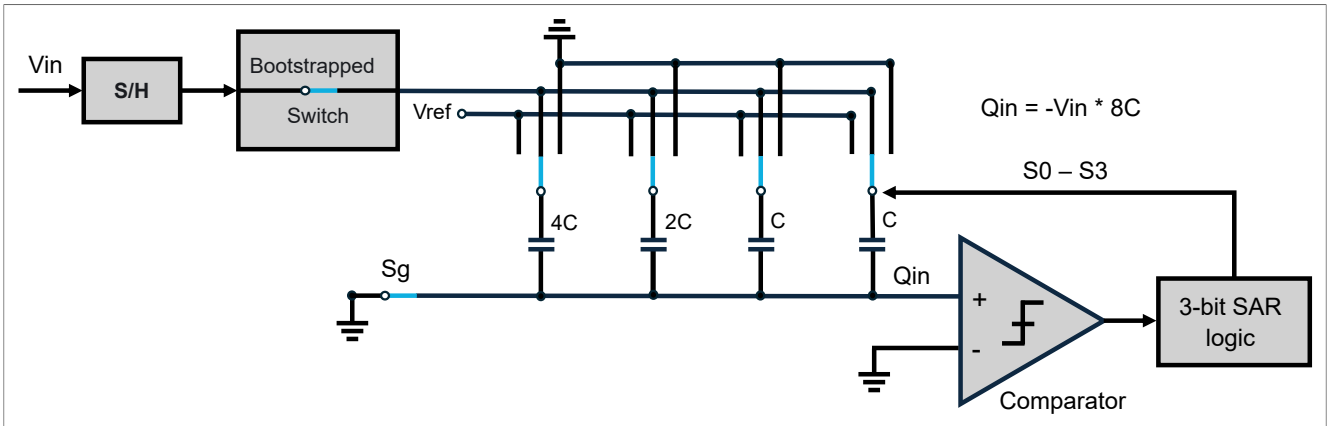

Figure 4. Single-ended SAR ADC sample state

Figure 5 shows the 3-bit SAR ADC 1/8 Vref compare state. When the bootstrapped switch and Sg are open, and S1 is connected to Vref; the quantity of electric charge is:

$$Qin(1) = Vin(1) * 7C + (Vin(1) - Vref) * C \quad (2)$$

Using the principle of conservation of energy:

$$Qin(1) = Qin \quad (3)$$

It implies:

$$Vin(1) = -Vin + \frac{1}{8} * Vref \quad (4)$$

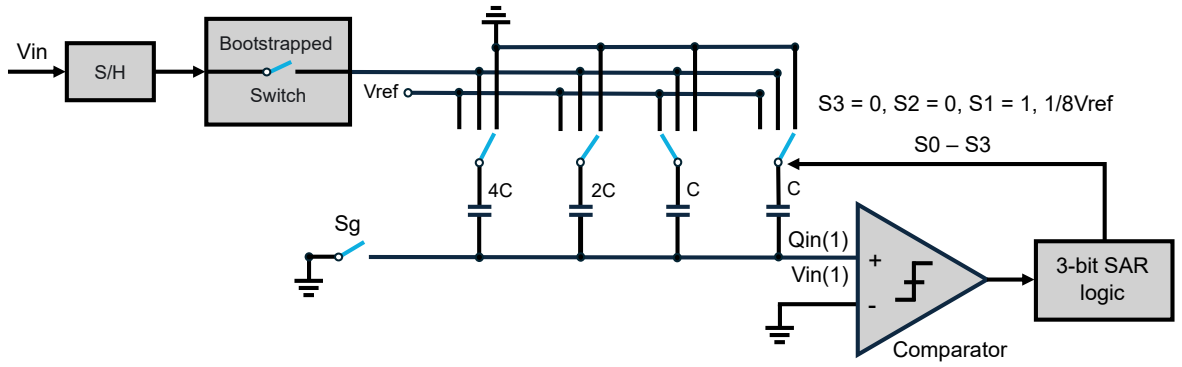

Figure 5. Single-ended SAR ADC 1/8 Vref compare state

Figure 6 shows the 3-bit SAR ADC 1/4 Vref compare state. When the bootstrapped switch and Sg are open, and S2 is connected to Vref; the quantity of electric charge is:

$$Qin(2) = Vin(2) * 6C + (Vin(2) - Vref) * 2C \quad (5)$$

Using the principle of conservation of energy:

$$Qin(2) = Qin \quad (6)$$

It implies:

$$Vin(2) = -Vin + \frac{1}{4} * Vref \quad (7)$$

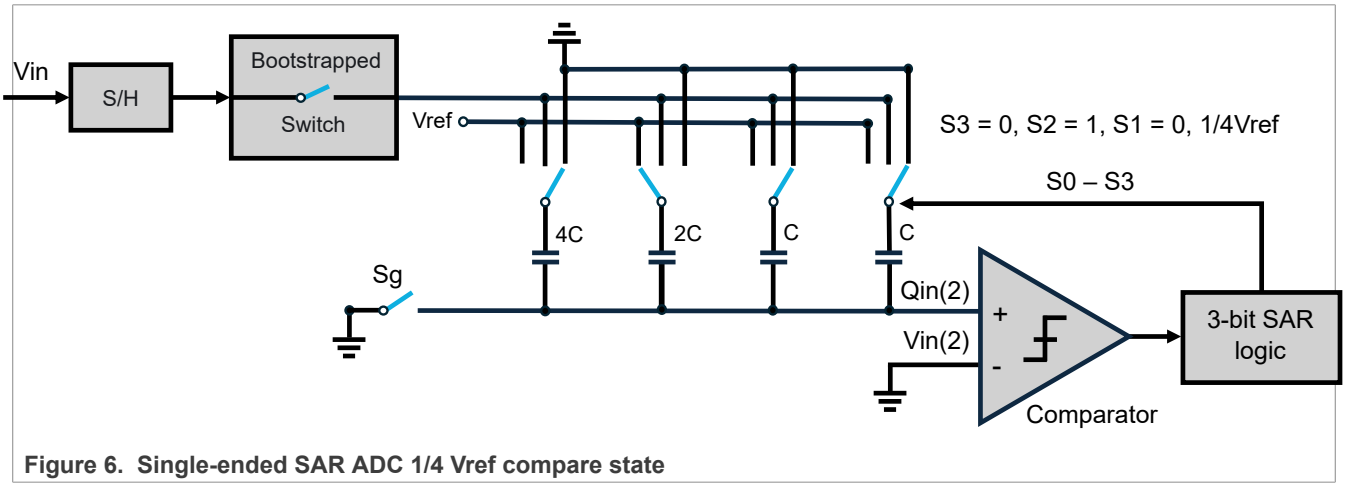

Figure 6. Single-ended SAR ADC 1/4 Vref compare state

Figure 7 shows the 3-bit SAR ADC 3/8 Vref compare state. When the bootstrapped switch and Sg are open, and S1 and S2 are connected to Vref; the quantity of electric charge is:

$$Qin(3) = Vin(3) * 5C + (Vin(3) - Vref) * 3C \quad (8)$$

Using the principle of conservation of energy:

$$Qin(3) = Qin \tag{9}$$

It implies:

$$Vin(3) = -Vin + \frac{3}{8} * Vref \tag{10}$$

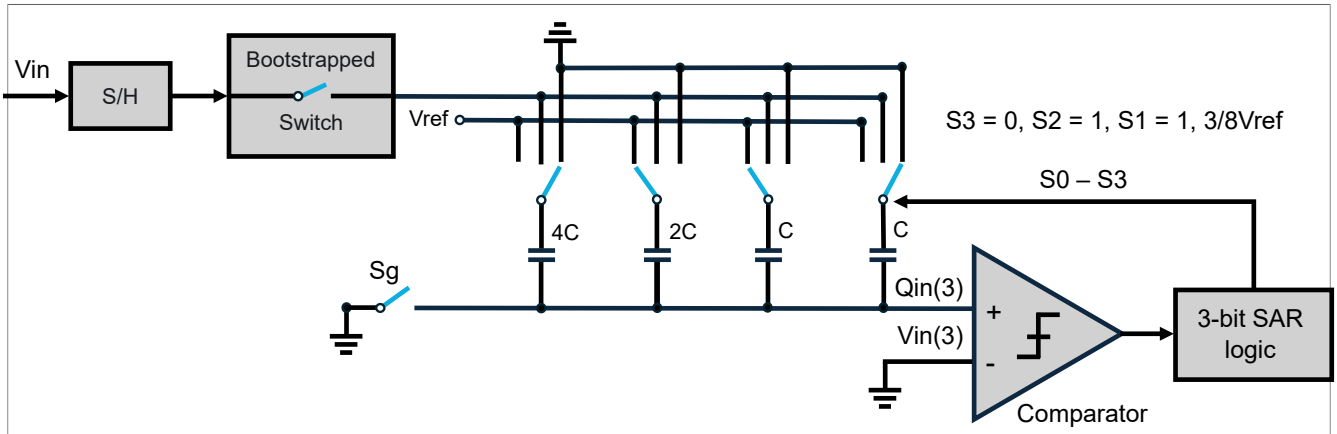

Figure 7. Single-ended SAR ADC 3/8 Vref compare state

Figure 8 shows the 3-bit SAR ADC 1/2 Vref compare state. When the bootstrapped switch and Sg are open, and S3 is connected to Vref; the quantity of electric charge is:

$$Qin(4) = Vin(4) * 4C + (Vin(4) - Vref) * 4C \tag{11}$$

Using the principle of conservation of energy:

$$Qin(4) = Qin \tag{12}$$

It implies:

$$Vin(4) = -Vin + \frac{1}{2} * Vref \tag{13}$$

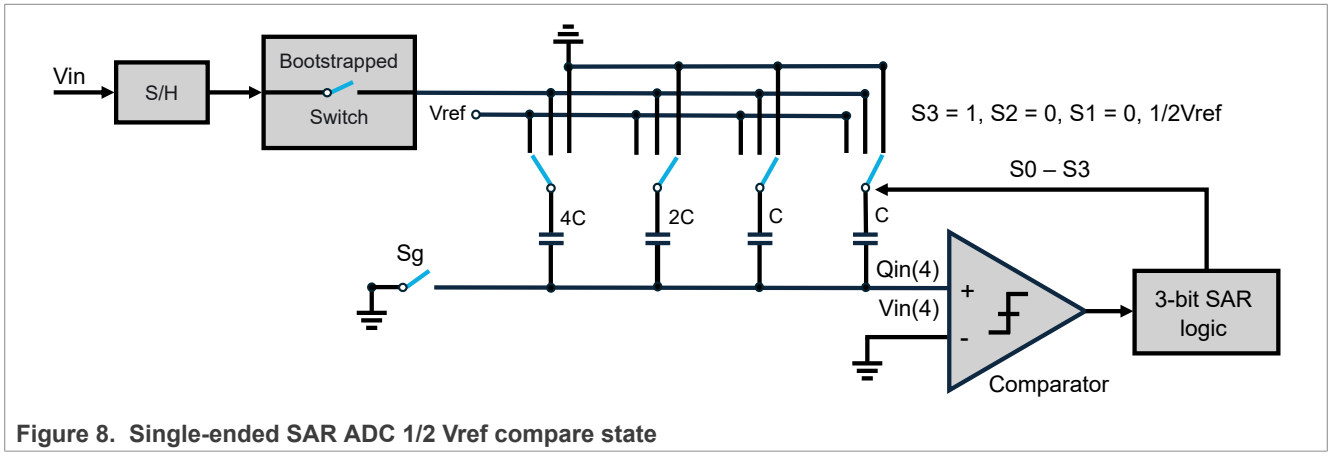

Figure 8. Single-ended SAR ADC 1/2 Vref compare state

Figure 9 shows the 3-bit SAR ADC 5/8 Vref compare state. When the bootstrapped switch and Sg are open, and S1 and S3 are connected to Vref; the quantity of electric charge is:

$$Qin(5) = Vin(5) * 3C + (Vin(5) - Vref) * 5C \tag{14}$$

Using the principle of conservation of energy:

$$Qin(5) = Qin \tag{15}$$

It implies:

$$Vin(5) = -Vin + \frac{5}{8} * Vref \tag{16}$$

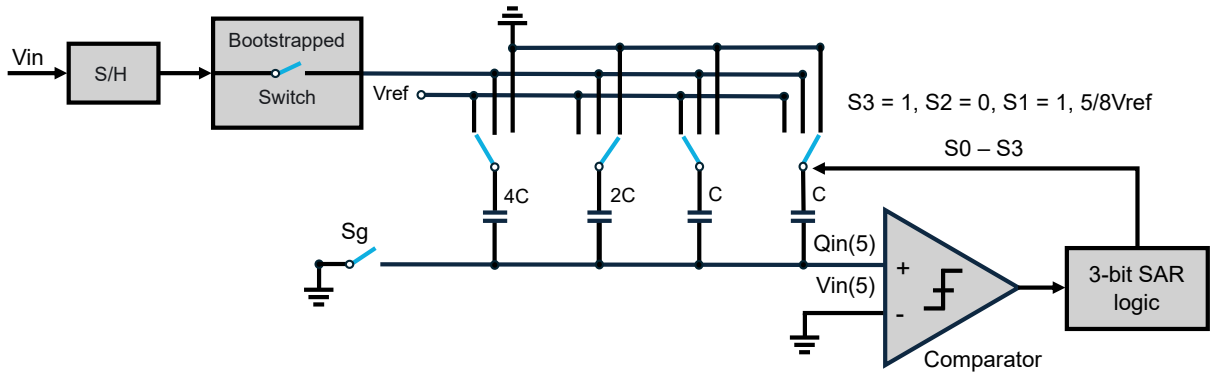

Figure 9. Single-ended SAR ADC 5/8 Vref compare state

Figure 10 shows the 3-bit SAR ADC 6/8 Vref compare state. When the bootstrapped switch and Sg are open, and S2 and S3 are connected to Vref; the quantity of electric charge is:

$$Qin(6) = Vin(6) * 2C + (Vin(6) - Vref) * 6C \tag{17}$$

Using the principle of conservation of energy:

$$Qin(6) = Qin \tag{18}$$

It implies:

$$Vin(6) = -Vin + \frac{6}{8} * Vref \tag{19}$$

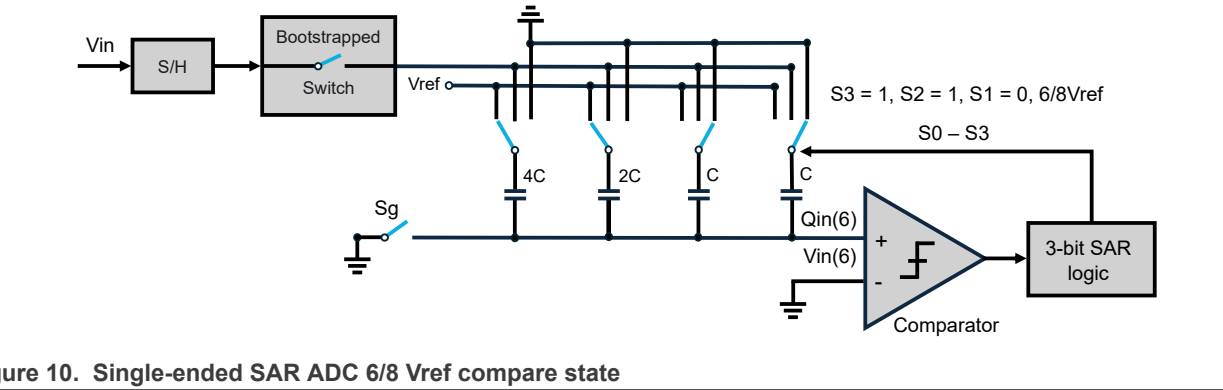

Figure 10. Single-ended SAR ADC 6/8 Vref compare state

Figure 11 shows the 3-bit SAR ADC 7/8 Vref compare state. When the bootstrapped switch and Sg are open, and S1, S2, and S3 are connected to Vref; the quantity of electric charge is:

$$Qin(7) = Vin(7) * C + (Vin(7) - Vref) * 7C \tag{20}$$

Using the principle of conservation of energy:

$$Qin(7) = Qin \tag{21}$$

It implies:

$$Vin(7) = -Vin + \frac{7}{8} * Vref \tag{22}$$

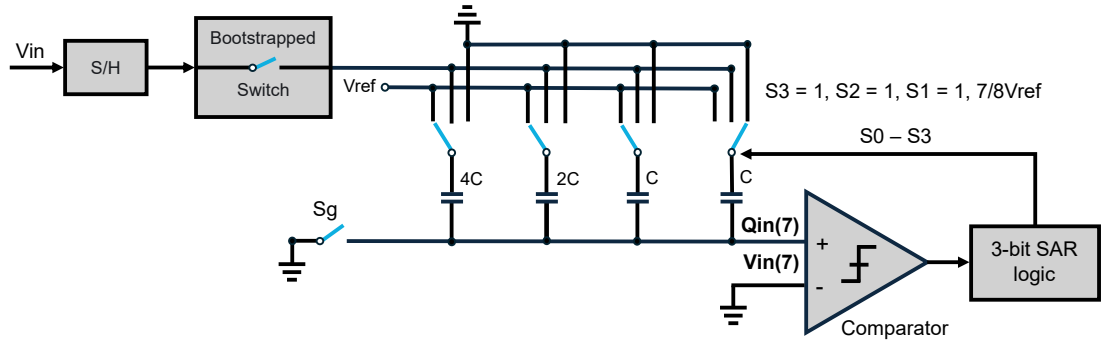

Figure 11. Single-ended SAR ADC 7/8 Vref compare state

4 Differential SAR ADC workflow

Figure 12 shows the sample state of a typical 3-bit differential SAR ADC having a Vcm-based capacitive DAC. When the bootstrapped switch is closed, Sp0 – Sp3 and Sn0 – Sn3 connect to Vcm. The quantity of electric charge is:

$$Qip = (Vip - Vcm) * 8C \tag{23}$$

$$Qin = (Vin - Vcm) * 8C \tag{24}$$

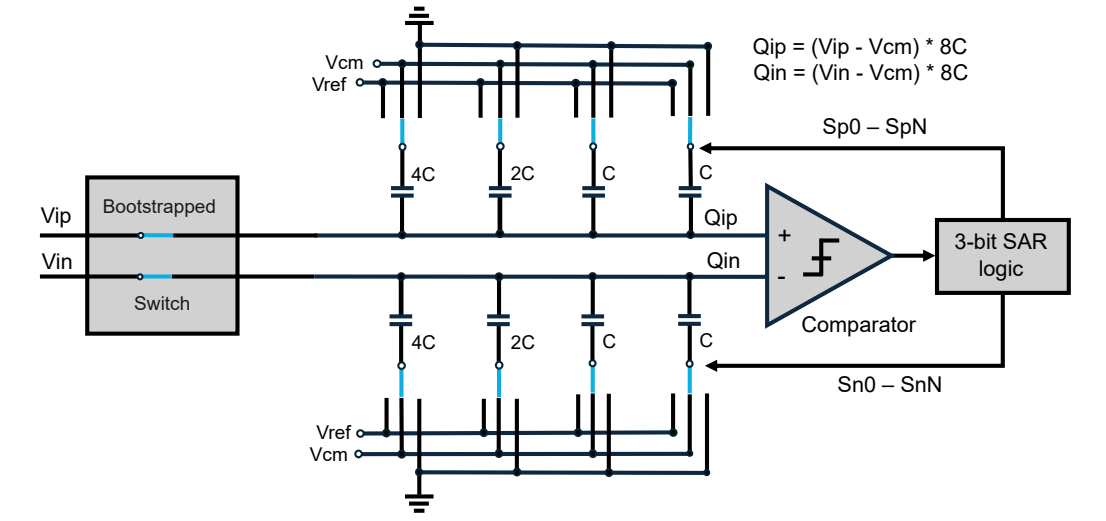

Figure 12. Full differential SAR ADC sample state

Figure 13 shows the full differential SAR ADC 1/8 Vref compare state. When the bootstrapped switch is open and Sp1 is connected to Vref, Sn1 connects to GND. The quantity of electric charge is:

$$Qip(1) = (Vip(1) - Vcm) * 7C + (Vip(1) - Vref) * C \tag{25}$$

$$Qin(1) = (Vin(1) - Vcm) * 7C + Vin(1) * C \tag{26}$$

Using the principle of conservation of energy:

$$Qip(1) = Qip \tag{27}$$

$$Qin(1) = Qin \tag{28}$$

It implies:

$$Vip(1) = Vip - \frac{1}{8} * Vcm + \frac{1}{8} * Vref \tag{29}$$

$$Vin(1) = Vin - \frac{1}{8} * Vcm \tag{30}$$

The differential value is:

$$Vip(1) - Vin(1) = \frac{1}{8} * Vref \tag{31}$$

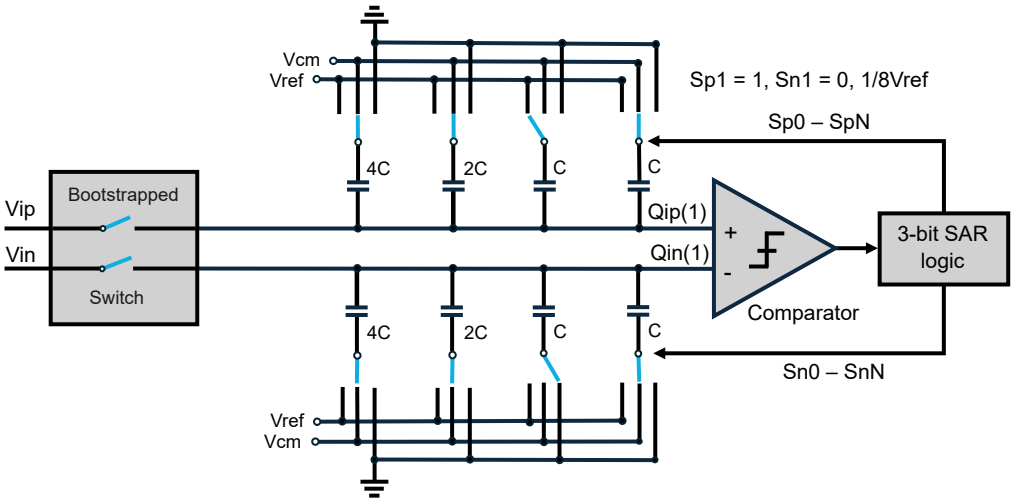

Figure 13. Full differential SAR ADC 1/8 Vref compare state

Figure 14 shows the full differential SAR ADC 1/4 Vref compare state. When the bootstrapped switch is open and Sp2 is connected to Vref, Sn2 connects to GND. The quantity of electric charge is:

$$Qip(2) = (Vip(2) - Vcm) * 6C + (Vip(2) - Vref) * 2C \tag{32}$$

$$Qin(2) = (Vin(2) - Vcm) * 6C + Vin(2) * 2C \tag{33}$$

Using the principle of conservation of energy:

$$Qip(2) = Qip \tag{34}$$

$$Qin(2) = Qin \tag{35}$$

It implies:

$$Vip(2) = Vip - \frac{1}{4} * Vcm + \frac{1}{4} * Vref \tag{36}$$

$$Vin(2) = Vin - \frac{1}{4} * Vcm \tag{37}$$

The differential value is:

$$Vip(2) - Vin(2) = \frac{1}{4} * Vref \tag{38}$$

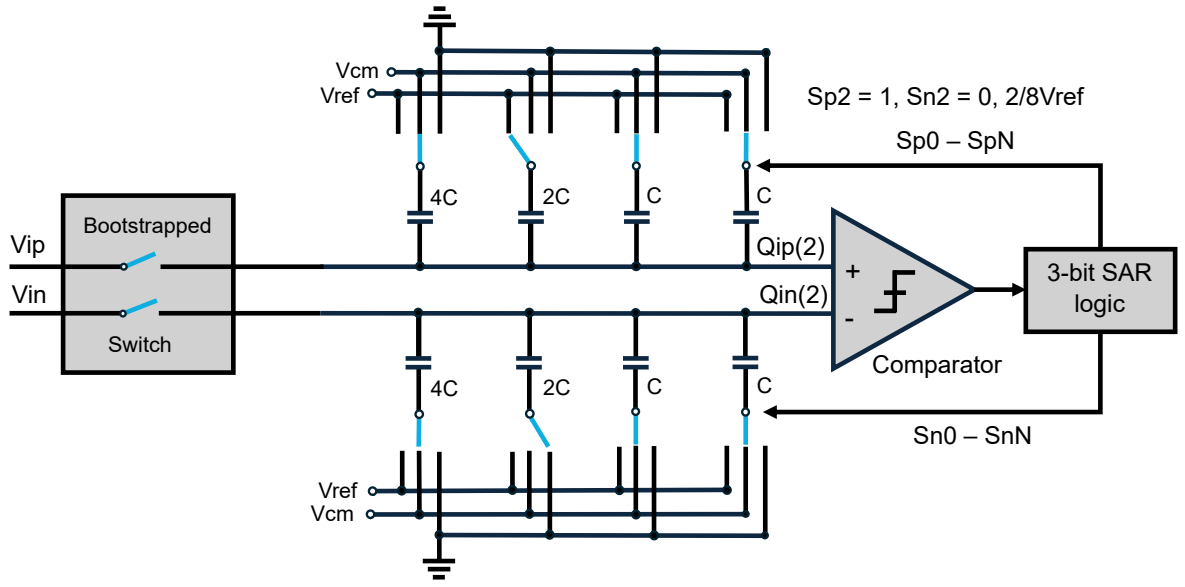

Figure 14. Full differential SAR ADC 1/4 Vref compare state

Figure 15 shows the full differential SAR ADC 3/8 Vref compare state. When the bootstrapped switch is open, and Sp1 and Sp2 are connected to Vref; Sn1 and Sn2 connect to GND. The quantity of electric charge is:

$$Qip(3) = (Vip(3) - Vcm) * 5C + (Vip(3) - Vref) * 3C \tag{39}$$

$$Qin(3) = (Vin(3) - Vcm) * 5C + Vin(3) * 3C \tag{40}$$

Using the principle of conservation of energy:

$$Qip(3) = Qip \tag{41}$$

$$Qin(3) = Qin \tag{42}$$

It implies:

$$Vip(3) = Vip - \frac{3}{8} * Vcm + \frac{3}{8} * Vref \tag{43}$$

$$Vin(3) = Vin - \frac{3}{8} * Vcm \tag{44}$$

The differential value is:

$$Vip(3) - Vin(3) = \frac{3}{8} * Vref \tag{45}$$

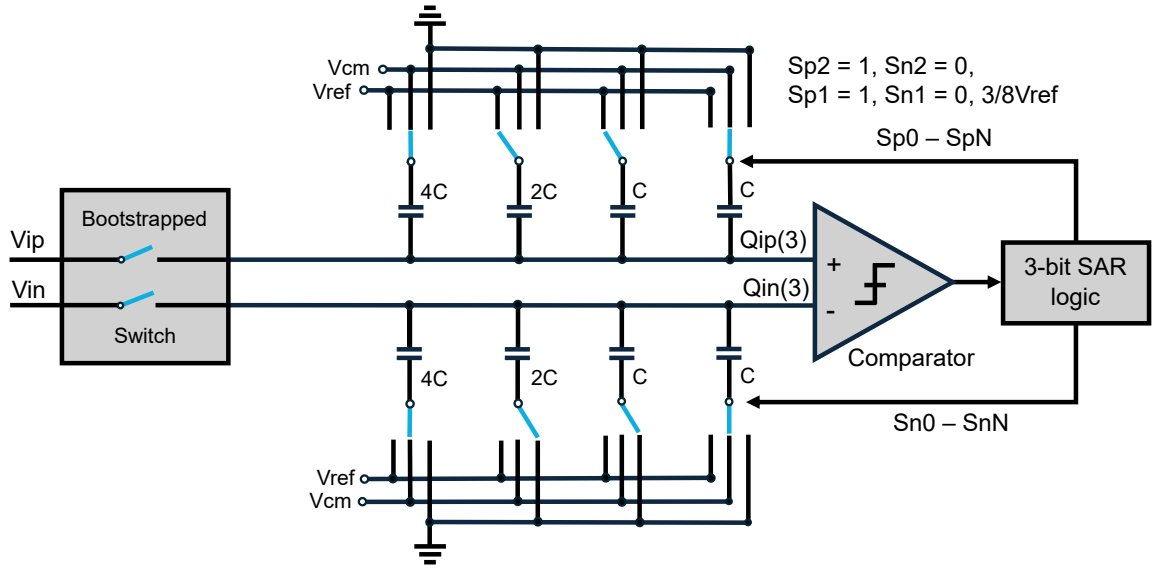

Figure 15. Full differential SAR ADC 3/8 Vref compare state

Figure 16 shows the full differential SAR ADC 1/2 Vref compare state. When the bootstrapped switch is open and Sp3 is connected to Vref, Sn3 connects to GND. The quantity of electric charge is:

$$Qip(4) = (Vip(4) - Vcm) * 4C + (Vip(4) - Vref) * 4C \tag{46}$$

$$Qin(4) = (Vin(4) - Vcm) * 4C + Vin(4) * 4C \tag{47}$$

Using the principle of conservation of energy:

$$Qip(4) = Qip \tag{48}$$

$$Qin(4) = Qin \tag{49}$$

It implies:

$$Vip(4) = Vip - \frac{1}{2} * Vcm + \frac{1}{2} * Vref \tag{50}$$

$$Vin(4) = Vin - \frac{1}{2} * Vcm \tag{51}$$

The differential value is:

$$Vip(4) - Vin(4) = \frac{1}{2} * Vref \tag{52}$$

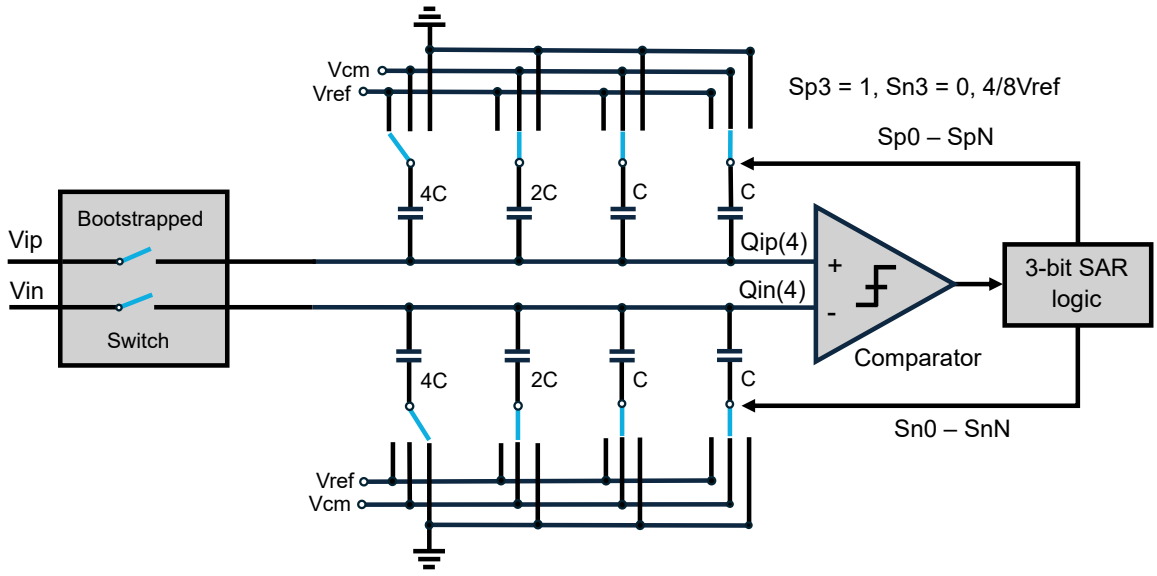

Figure 16. Full differential SAR ADC 1/2 Vref compare state

Figure 17 shows the full differential SAR ADC 5/8 Vref compare state. When the bootstrapped switch is open, and Sp1 and Sp3 are connected to Vref; Sn1 and Sn3 connect to GND. The quantity of electric charge is:

$$Qip(5) = (Vip(5) - Vcm) * 3C + (Vip(1) - Vref) * 5C \tag{53}$$

$$Qin(5) = (Vin(5) - Vcm) * 3C + Vin(1) * 5C \tag{54}$$

Using the principle of conservation of energy:

$$Qip(5) = Qip \tag{55}$$

$$Qin(5) = Qin \tag{56}$$

It implies:

$$Vip(5) = Vip - \frac{5}{8} * Vcm + \frac{5}{8} * Vref \tag{57}$$

$$Vin(5) = Vin - \frac{5}{8} * Vcm \tag{58}$$

The differential value is:

$$Vip(5) - Vin(5) = \frac{5}{8} * Vref \tag{59}$$

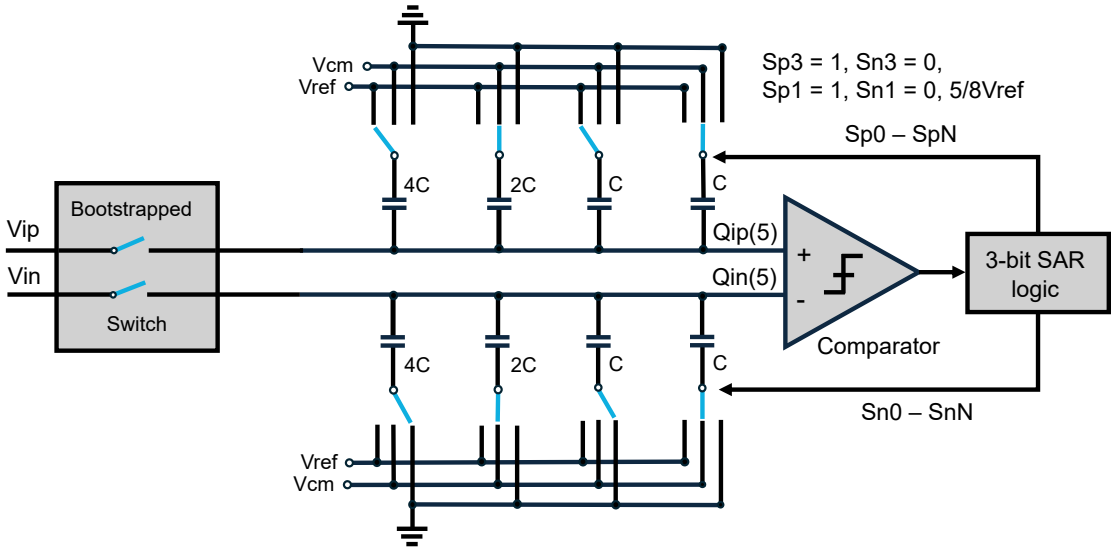

Figure 17. Full differential SAR ADC 5/8 Vref compare state

Figure 18 shows the full differential SAR ADC 6/8 Vref compare state. When the bootstrapped switch is open, and Sp2 and Sp3 are connected to Vref; Sn2 and Sn3 connect to GND. The quantity of electric charge is:

$$Qip(6) = (Vip(6) - Vcm) * 2C + (Vip(6) - Vref) * 6C \tag{60}$$

$$Qin(6) = (Vin(6) - Vcm) * 2C + Vin(6) * 6C \tag{61}$$

Using the principle of conservation of energy:

$$Qip(6) = Qip \tag{62}$$

$$Qin(6) = Qin \tag{63}$$

It implies:

$$Vip(6) = Vip - \frac{3}{4} * Vcm + \frac{3}{4} * Vref \tag{64}$$

$$Vin(6) = Vin - \frac{3}{4} * Vcm \tag{65}$$

The differential value is:

$$Vip(6) - Vin(6) = \frac{3}{4} * Vref \tag{66}$$

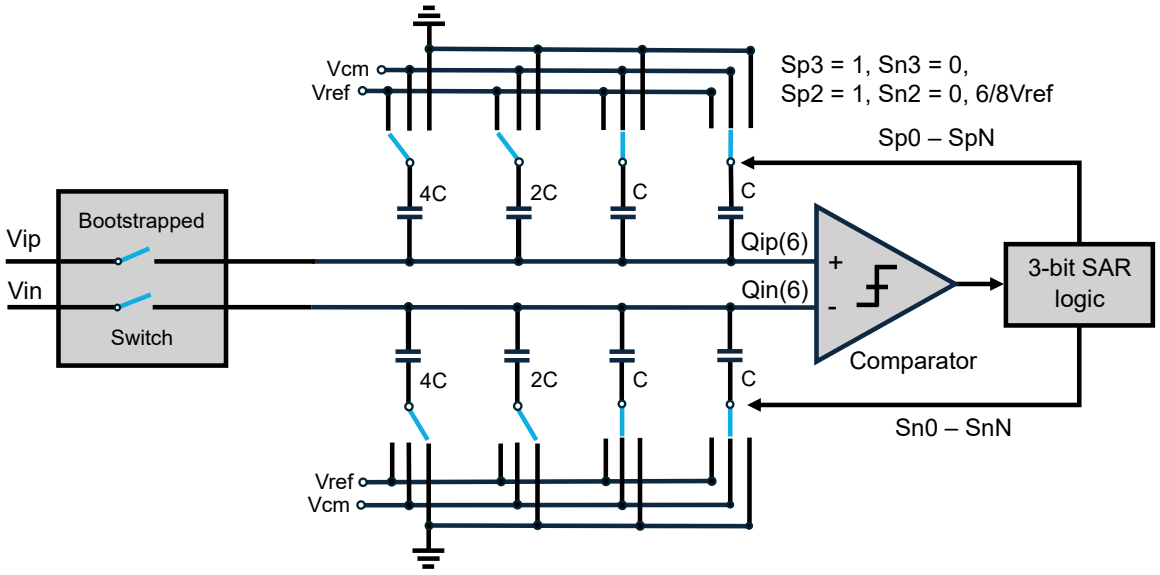

Figure 18. Full differential SAR ADC 6/8 Vref compare state

Figure 19 shows the full differential SAR ADC 7/8 Vref compare state. When the bootstrapped switch is open; and Sp1, Sp2, and Sp3 are connected to Vref; Sn1, Sn2, and Sn3 connect to GND. The quantity of electric charge is:

$$Qip(7) = (Vip(7) - Vcm) * C + (Vip(7) - Vref) * 7C \tag{67}$$

$$Qin(7) = (Vin(7) - Vcm) * C + Vin(7) * 7C \tag{68}$$

Using the principle of conservation of energy:

$$Qip(7) = Qip \tag{69}$$

$$Qin(7) = Qin \tag{70}$$

It implies:

$$Vip(7) = Vip - \frac{7}{8} * Vcm + \frac{7}{8} * Vref \tag{71}$$

$$Vin(7) = Vin - \frac{7}{8} * Vcm \tag{72}$$

The differential value is:

$$Vip(7) - Vin(7) = \frac{7}{8} * Vref \tag{73}$$

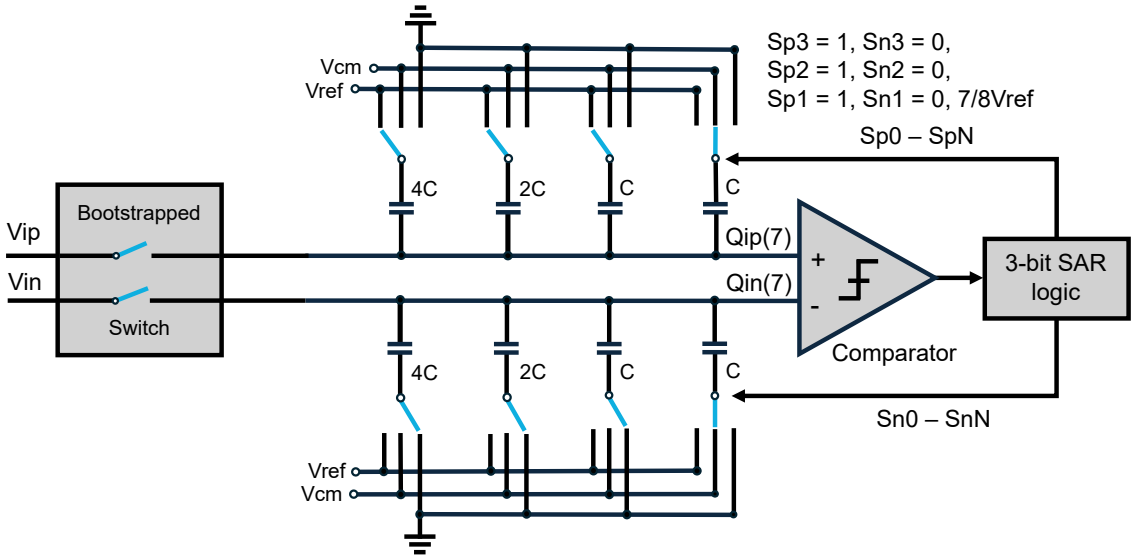

Figure 19. Full differential SAR ADC 7/8 Vref compare state

5 ADC static performance

When evaluating the ADC static performance, the following parameters must be evaluated:

- ADC sample precision
- ADC sample stability

Figure 20 shows the ADC quantizer input-output characteristics. In part (a) of Figure 20, differential nonlinearity (DNL) and integral nonlinearity (INL) are marked. The black line is the ideal code line and the red line is the actual code line.

DNL is the difference between a specified code bin width and the ideal code bin width. Its unit is LSB and normally the ADC specification gives the maximum value of this parameter. DNL is calculated as follows:

$$DNL(k) = \frac{W[k] - Q}{Q} \tag{74}$$

Where:

- W[k] = $\left(T[K+1] - T[K]\right)$
- Q: Ideal code bin width $Q = FSR / 2^N$

INL is the maximum deviation between ideal code line and actual code line. Its unit is LSB or percent of the full-scale range. INL can be calculated using Equation (75). This method is described in IEEE Std 1241–2000.

$$INL(k) = 100\% * \frac{\varepsilon(k)}{2^N * Q} = 100\% * \frac{\varepsilon(k)}{V_{FS}} \tag{75}$$

Where:

- $\mathrm{INL(k)}$: Integral nonlinearity at output code k
- $\varepsilon(k)$: Difference between $T[K]$ and ideal value of $T[K]$ computed from G and Vos
- Q: Ideal code bin width, expressed in input units
- V_{FS}: Full-scale range of the ADC in input units

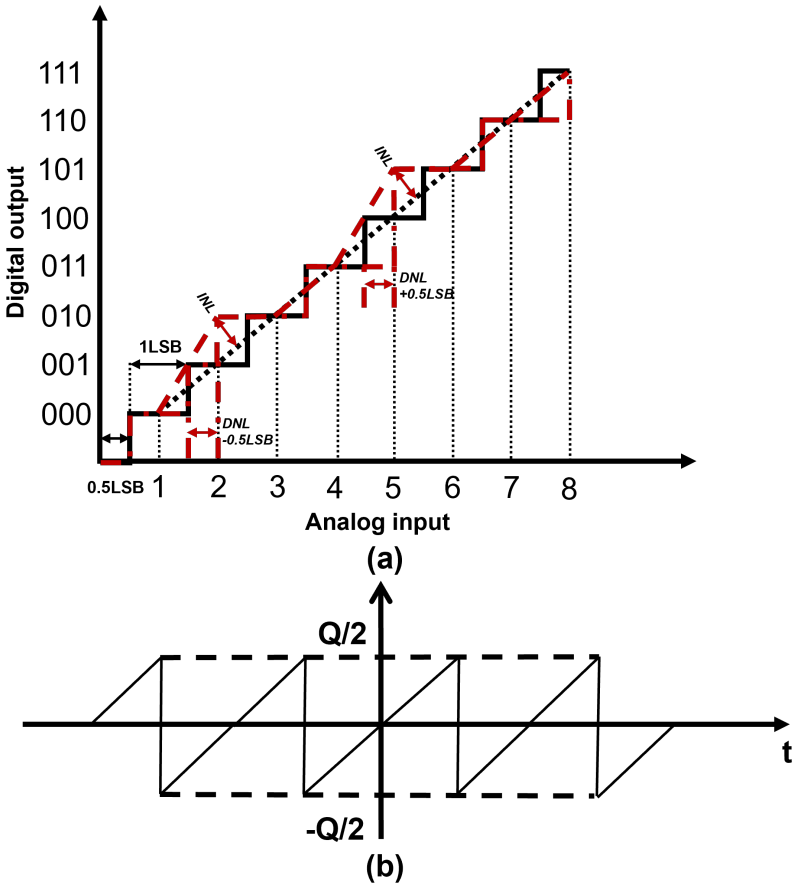

Figure 20. ADC quantizer input-output characteristics

6 ADC dynamic performance

When evaluating the dynamic performance of ADC, several parameters are used, for example:

- Signal-to-noise ratio (SNR)
- Signal-to-noise-and-distortion ratio (SINAD)
- Effective number of bits (ENOB)
- Total harmonic distortion (THD)
- Total harmonic distortion plus noise (THD+N)

To test ADC dynamic performance, usually a full-scale signal is used as the ADC input. The spectrum information is obtained using the discrete FFT (DFT) algorithm.

The value of the signal-to-noise ratio (SNR) parameter can be calculated as follows:

$$SNR = 20\log\left(\frac{Srms}{Nrms}\right) \tag{76}$$

Where:

- S_{rms}: Root mean square (RMS) of input signal
- N_{rms}: RMS of noise

If the input signal is defined as a full-scale sine wave:

$$Input_{Signal} = Q * 2^{N-1} * \sin(2\pi f t) \tag{77}$$

The S_{rms} can be defined as:

$$S_{rms} = \frac{Q * 2^{N-1}}{\sqrt{2}} \tag{78}$$

As shown in part (b) of Figure 20, for an ideal ADC, the noise only have quantization noise, which is in the range from $\frac{Q}{2}$ to $-\frac{Q}{2}$. Therefore, N_{rms} can be defined as:

$$N_{rms} = \frac{Q}{\sqrt{12}} \tag{79}$$

Using Equation (76), Equation (78), and Equation (79), the SNR can be defined as:

$$SNR = 20\log\left(\frac{\frac{Q * 2^{N-1}}{\sqrt{2}}}{\frac{Q}{\sqrt{12}}}\right) \tag{80}$$

Simplifying Equation (80) gives Equation (81) and Equation (82):

$$SNR = 20\log\left(\frac{2^{N-1}}{\sqrt{2}}\right) - 20\log\left(\frac{1}{\sqrt{12}}\right) \tag{81}$$

$$SNR = 6.02 * N + 1.76 \tag{82}$$

As shown in Figure 21, the SNR is full-scale input range minus noise floor. Therefore, $Noise\_Floor$ can be calculated as:

$$Noise\_Floor = 20\log\frac{S_{rms}}{F_{rms}} - SNR \tag{83}$$

Where the F_{rms} is defined as the ADC full-scale RMS value.

When the bandwidth of the input signal is smaller than the Nyquist bandwidth and the ADC output data is processed by a DFT algorithm, the FFT noise floor can be calculated using Equation (84). This method is described in IEEE Std 1241–2000.

$$Noise\_Floor = 10\log\frac{M}{2ENBW} - SNR \tag{84}$$

Where $ENBW$ is equal to the noise bandwidth of the window function, as defined below:

$$ENBW = \frac{M\sum_{k=0}^{M-1} W[k]^2}{\left(\sum_{k=0}^{M-1} W[k]\right)^2} \tag{85}$$

Where:

- M: ADC sample points
- k: Window coefficient index
- w[k]: Window coefficient

Figure 21 also mentions one more parameter, noise spectrum density. Some manufacturers provide this parameter as a reference for ADC performance evaluation. Its unit should be DbFS/Hz. Noise spectrum density is defined as:

$$NSD = 10log10(Fs/2) \tag{86}$$

Where Fs is the sample frequency.

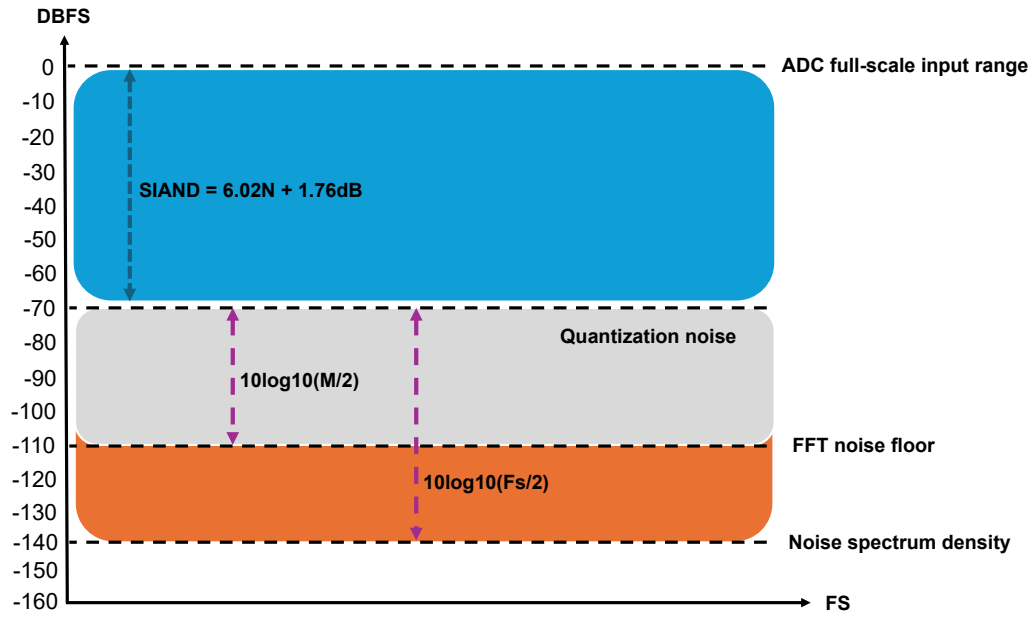

Figure 21. ADC DFT spectrum

The equations mentioned earlier in this section talk about SNR. These equations are based on an ideal ADC, which has only quantization noise. However, in reality, apart from quantization noise, an ADC also generates distortion during ADC conversions. Therefore, when evaluating ADC performance, use the signal-to-noise-and-distortion ratio (SINAD) parameter, which includes distortion.

SINAD is defined as follows:

$$SINAD = 20\log\left(\frac{Srms}{Nrms + Drms}\right) \tag{87}$$

The total harmonic distortion (THD) is defined as:

$$THD = 20\log\left(\frac{Srms}{Drms}\right) \tag{88}$$

The total harmonic distortion (THD) can also be calculated using Equation (89). This method is described in IEEE Std 1241–2000.

$$THD = 20\log\left(\frac{1}{M} * \sqrt{\sum_{h} Xavm(fh)^2}\right) \tag{89}$$

Where

- $Xavm(fh)$: Averaged magnitude of the component at the hth harmonic of the DFT of the ADC output data record
- M: Number of samples in the data record

Another important parameter used for evaluating the ADC dynamic performance is the effective number of bits (ENOB). This parameter is related to the SINAD and SNR parameters, which contain noise information. IEEE Std 1241–2000 defines the ENOB as:

$$ENOB = N - log_2\left(\frac{rms\ noise}{ideal\ rms\ quantization\ error}\right) = log_2\left(\frac{full\ scale\ range}{rms\ noise * \sqrt{12}}\right) \quad (90)$$

And the relationship between SINAD and ENOB as:

$$SINAD = 20*log\left(\sqrt{\frac{3}{2}} * \left(\frac{2*A}{V}\right) * 2^{ENOB}\right) \quad (91)$$

$$SINAD = 20*log\left(\sqrt{\frac{3}{2}}\right) + 20*log\left(\frac{2*A}{V}\right) + 20*\log\left(2^{ENOB}\right) \quad (92)$$

$$SINAD = 10*log\left(\frac{3}{2}\right) + 20*log\left(\frac{2*A}{V}\right) + 20*ENOB*\log(2) \quad (93)$$

$$SINAD = 1.76 + 6.02*ENOB + 20*log\left(\frac{2*A}{V}\right) \quad (94)$$

In Equation (94):

- A is the amplitude of the sine wave fitted to the output
- V is the full-scale range of the ADC under test

When the input signal is full scale, SINAD can be calculated as:

$$SINAD = 1.76 + 6.02*ENOB \quad (95)$$

Equation (82) and Equation (95) imply that SINAD is the same as SNR. It is so because the SINAD and ENOB equations are based on an ideal ADC system, which has only quantization noise. However, in reality, ADC also has distortion. Therefore, to get the ENOB value, first calculate the SINAD value using Equation (87) and then calculate ENOB.

7 MCX A3xx ADC features

An MCX A3xx part contains four single-ended ADC modules. An ADC module has the following features:

- Supports linear successive approximation algorithm
- Supports single-ended operation with 16-bit or 12-bit resolution
- Supports four hardware trigger sources with priority level configuration and eight entry conversion result FIFOs with configurable watermark and overflow detection
- Seven command buffers allow independent option selection and channel sequence scanning
- Supports automatic compare with "store on true" and "repeat until true" options
- Supports configurable analog input sample time
- Supports configurable speed options to accommodate operation in SoC low-power modes
- Supports interrupt, Direct Memory Access (DMA), or polled operation
- Supports linearity and gain adjustment calibration logic

MCX A3xx supports several low-power modes. Table 1 shows the state of an ADC module in different power modes.

Table 1. ADC state in different power modes

Module	Power mode			
	Sleep	Deep sleep	Power down	Deep power down
16-bit HS ADC0/1/2/3 Digital	On	Static/LP	Static	Off
16-bit HS ADC0/1/2/3 Analog	On/Off	On/Off	Static	Off

8 MCX A3xx ADC operation

This section provides information about the ADC clock source, reference voltage source, trigger, CMD configuration, compare function, and calibration.

8.1 ADC clock sources

The maximum ADC frequency is:

- 24 MHz in the Mid Drive (MD) mode, where VDD_CORE is 1.0 V
- 64 MHz in Over Drive (OD) mode, where VDD_CORE is 1.2 V

The ADC has four clock sources and one 4-bit clock divider, as shown in Figure 22.

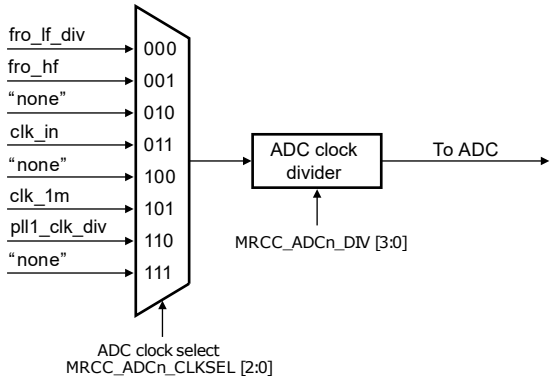

Figure 22. ADC clock sources and divider

In Figure 22:

- fro_lf_div: 12 MHz clock output from fro_lf_div
- fro_hf: Clock output from FRO180M (the frequency is controlled by SCG_FIRCCFG[FREQ_SEL])
- clk_in: Input clock from an external oscillator
- clk_1m: 1 MHz clock output from FRO12M
- pll1_clk_div: Clock from pll1_div output

8.2 ADC voltage reference options

The ADC voltage references can be selected by CFG[REFSEL] as follows:

- CFG[REFSEL] = 00: VREFH reference pin
- CFG[REFSEL] = 01: VREFI
- CFG[REFSEL] = 10: VDDA_ANA supply pin

CAUTION: *The VREFI function is multiplexed on the P2_7 I/O pin of the MCU. To avoid any impact on the ADC dynamic performance from the pins located near the P2_7 pin on the chip package, the toggle speed of such pins should be less than 1 kHz.*

8.3 ADC trigger inputs

For each ADC module, four hardware trigger sources are available. The trigger source can be selected in the INPUTMUX. The 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. For a detailed list of trigger sources and corresponding mux sort numbers, see the descriptions of the "ADC Trigger input connections (ADCx_TRIG0–ADCx_TRIG3)" registers in the "Input Multiplexing (INPUTMUX)" chapter of MCX A34x Reference Manual.

The execution of the ADC commands CMD1–CMD7 is initiated from the triggers. Each trigger can be generated from the software by setting the corresponding SWTRIG[SWTn] bit field. Alternatively, hardware triggers can be generated from asynchronous input sources at the periphery of the module. For example, to trigger a conversion periodically, use a Pulse Width Modulation (PWM) signal. 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]).

8.4 ADC CMD

The ADC performs conversion according to a CMD configuration, and loads the next CMD or recover to the IDLE state. A CMD can be triggered by a hardware trigger from INPUTMUX or a software trigger. The conversion results are stored in result FIFO along with the corresponding CMD configuration.

The detailed ADC conversion behavior can be configured in the CMD register:

- CMDLx[MODE]: Selects the resolution of the conversion.
- CMDLx[ADCH]: Selects the input channel.
- CMDHx[NEXT]: Selects the next CMD to be executed. If this area is 0, the ADC returns to the IDLE state, and waits for the next trigger event.
- CMDHx[LOOP]: Selects how many times this command is executed repeatedly. The command executes LOOP+1 times, and a result is stored in the FIFO every time.
- CMDHx[AVGS]: Selects how many ADC conversions are averaged to create a result. The conversion executes two hardware average times, but only one average result is stored in the FIFO.
- CMDHx[STS]: Selects the sample time. The minimum sample time is 3.5 ADCK cycles if STS = 0. Total sample time is 3.5 + 2*sample time of ADCK cycles. STS has a non-zero value.
- CMDHx[LWI]: If this bit is set, the input channel must be increased when the command is executed in the next loop.
- CMDHx[WAIT_TRIG]: If this bit is set, the next command pointed by the current command cannot be executed until the active trigger is asserted again.
- CMDHx[CMPEN]: Configures the compare function:
  - When CMPEN = 10b, the compare function is enabled and the ADC results are stored in the FIFO. Then, the compare result is true. The loop count is increased after command execution regardless of the compare result.
  - When CMPEN = 11b, the command is executed repeatedly until the compare result is true.

8.5 Compare function

For each command, seven compare value (CV) registers CV1–CV7 are available. Every two CV registers can be separated with two 16-bit areas:

- Compare Value Low (CVL)
- Compare Value High (CVH)

The compare result is true if the conversion result is greater than the CVH or less than the CVL.

8.6 ADC calibration

The ADC module has offset calibration and ADC calibration functions that must be executed in ADC initialization to achieve more accuracy. Calibration must be run after any reset and before a conversion is executed. Averaging multiple conversions can achieve improved accuracy during calibration routines. You are recommended to set CTRL[CAL_AVGS] for a minimum of 256 averaging during calibration steps.

The Offset Trim (OFSTRIM) register is used to trim the ADC comparator offset voltage. The ADC supports an offset calibration function, which updates the OFSTRIM register automatically. Initiate offset calibration by setting CTRL[CALOFS]. Poll the STAT[CAL_RDY] flag. When STAT[CAL_RDY] is asserted, the offset calibration function is executed and the OFSTRIM register is updated.

The ADC includes a hardware calibration logic, which performs the calibration of the converter for gain error and linearity error correction of the raw conversion result. The ADC supports a calibration function, which updates the CAL_GAR registers automatically. The calibration function also updates GCC0[GAIN_CAL]. A software calculation is required to derive GCR0[GCALR] from GCC0[GAIN_CAL].

Perform these steps to prepare the calibration setup:

1. Execute offset calibration steps. The OFSTRIM register is used during calibration to trim the comparator offset voltage.
2. Initiate the calibration routine by writing 1 to CTRL[CAL_REQ]:
   a. CTRL[CAL_REQ] remains 1 until the ADC accepts the CAL routine.
   b. After ADC acceptance, CTRL[CAL_REQ] becomes 0 automatically.
3. Poll the GCR0[RDY] flag. When it is asserted, the hardware-controlled calibration operation is performed, and CAL_GAR and GCC0[GAIN_CAL] registers are updated. The updated value in GCC0[GAIN_CAL] is needed for further software processing described in the following steps.
4. Read GCC0[GAIN_CAL], which is a 16-bit signed value. It is used for calculating the gain adjustment.
5. Calculate the gain adjustment as follows:
   Gain adjustment = (131072)/(131072-GCC0[GAIN_CAL])
   The result is a floating-point value between 0 and 2.
6. Convert the floating-point value into its integer component (0 or 1) and its fractional component rounded off to 16 bits.
7. Store the integer and fractional components in the GCALR bits (16–0) of the Gain Calculation Result (GCR0) register:
   a. Store the integer value in GCR0[GCALR[16]].
   b. Store the fractional component in GCR0[GCALR[15:0]].
8. After storing the result from the gain adjustment calculation in GCR0[GCALR], set the GCR0[RDY] flag to indicate that it is valid.

After completing the above steps, the calibration sequence is completed and the STAT[CAL_RDY] flag is set. The STAT[CAL_RDY] flag remains set until the user resets the system or requests a new calibration sequence.

When STAT[CAL_RDY] is set, the ADC is configured to run in calibrated mode. Each conversion uses a combination of linearity and gain calibration results to correct Successive Approximation Register (SAR) data. Calibration conversion latency is required to process each sample. However, due to the pipelined nature of the data and control sequences, a conversion can happen without experiencing this calibration delay.

9 MCX A3xx ADC calculation tool

The purpose of the ADC calculation tool is to define the maximum sampling rates achieved, depending on the input signal impedance characteristics. To sample the input voltage accurately, the source resistance and ADC sample time must be chosen appropriately. The ADC simplified circuit is shown in Figure 23.

Note: *To calculate the sampling time or the source impedance, use the MCX A3xx ADC calculation tool available at ADC_Specs_Calculator_MCXA.xlsx.*

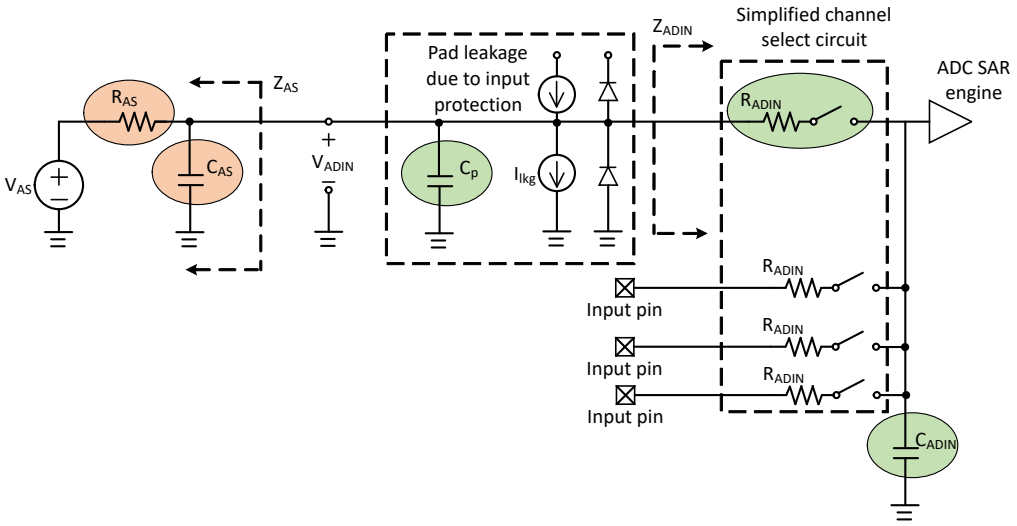

Figure 23. Simplified ADC circuit diagram

Equation (96) provides the required sample time for a fixed analog source resistance (R_{AS}):

$$T_{SMP_{REQ}} = B*\ln(2)*\left[R_{AS}*(C_{AS}+C_P+C_{ADIN})+(R_{AS}+R_{ADIN})*C_{ADIN}\right] \quad (96)$$

Where:

- B: The desired accuracy in bits
- R_{AS}: Input impedance value
- C_{AS}: Source capacitance
- C_P: Parasitic capacitance of pad/package
- C_{ADIN}: Input capacitance
- R_{ADIN}: Input resistance

The actual sample time is determined by the ADC input clock frequency (F_{ADCK}) and Sample Time Select (STS) bits in the ADC command register, which selects the number of sample cycles:

$$T_{SMP} = \frac{CYC_{SMP}}{F_{ADCK}} \quad (97)$$

Where:

- CYC_{SMP} is programmable from 3.5 to 131.5 cycles.
- T_{SMP} must be configured as greater than or equal to T_{SMP_REQ}.

If you set $T_{SMP} > T_{SMP\_REQ}$ and calculate R_{AS}, you can find the maximum source resistance required to sample at the desired accuracy:

$$R_{AS} < \frac{\frac{CYC_{SMP}}{F_{ADCK}*B*\ln(2)} - R_{ADIN}*C_{ADIN}}{C_{AS} + C_{P+}2*C_{ADIN}} \quad (98)$$

Equation (98) assumes that the input voltage can change up to full-scale (Vrefh-Vrefl) between subsequent conversions. If the maximum input voltage change between subsequent conversions is known and less than

Vrefh-Vrefl, the value of B can be adjusted. For example, if the maximum input voltage change is known ((Vrefh-Vrefl)/4), then to get 1/2 LSB sample accuracy in 12-bit mode, set B = 11, instead of setting B = 13.

The MCX A3xx ADC calculation tool has two sections. The first section shown in Figure 24 specifies the required sample time for a fixed R_{AS}. The user can input the source resistance R_{AS}, source capacitance C_{AS} (both are external components), the resolution B, and the value of the acceptable sampling error accuracy modifier of the ADC.

ADC Sample Time Calculator		
To calculate the required sample time given fixed R_{AS}		
Input (Enter value in yellow highlighted cells)		
R_{AS}	100	Ω
C_{AS}	150.00	pF
Resolution	12	bits
Accuracy Modifier	1	bits
B	10	bits
C_P	2.00	pF
C_{ADIN}	1.92	pF
Output		
Direct input channel	**R_{ADIN} (kΩ)**	**T_{SMP_REQ} (ns)**
V_{DDA}>1.8	1.65	129.98
V_{DDA}>3.0	1.35	125.99
Muxed input channel	**R_{ADIN} (kΩ)**	**T_{SMP_REQ} (ns)**
V_{DDA}>1.8	7.00	241.41
V_{DDA}>3.0	2.01	161.72

Figure 24. ADC sample time calculator

The second section of the tool shown in Figure 25 provides the maximum source resistance based on the sample time and the ADC frequency used. In the first section, the user can change the yellow highlighted cells according to the parameters being used. Ensure that the CYC_{SMP_USED} is greater than CYC_{SMP_MIN}. Then, the maximum source resistance and conversion rate can be generated. The user can use the data to design a sample circuit and initialize the ADC with appropriate parameters.

ADC Sampling Frequency and R_{AS} Calculator						
To calculate maximum R_{AS} with given sample time and ADC input clock frequency						
Input (Enter value in yellow highlighted cells)						
Resolution	12	bits	CMDH[STS]	0		
C_{AS}	10.00	pF	CMDH[AVGS]	0		
ADCnCLKDIV[DIV]	3		CFG2[HS]	1		
Accuracy Modifier	0	bits	CFG2[HSEXTRA]	0		
B	10	bits	CFG2[TUNE]	1		
ADC clock source frequency	192	MHz	CMDLa[MODE]	0		
F_{ADCK}	48	MHz	ADCK cycles/conversion	16.0		
C_P	2.00	pF				
C_{ADIN}	1.92	pF				
Output						
Direct input channel	**R_{ADIN} (kΩ)**	**CYC_{SMP_MIN}**	**CYC_{SMP_USER}**	**T_{SMP} (ns)**	**R_{AS_MAX} (Ω)**	**ADC sampling frequency (Msamples/s)**
V_{DDA}>1.8	1.65	1.1	3.5	72.92	464.12	3.00
V_{DDA}>2.1	1.35	0.9	3.5	72.92	500.48	3.00
Muxed input channel	**R_{ADIN} (kΩ)**	**CYC_{SMP_MIN}**	**CYC_{SMP_USER}**	**T_{SMP} (ns)**	**R_{AS_MAX} (Ω)**	**ADC sampling frequency (Msamples/s)**
V_{DDA}>1.8	7.00	4.5	3.5	72.92	ERR	3.00
V_{DDA}>2.1	2.01	1.3	3.5	72.92	420.48	3.00

Figure 25. ADC sample frequency and R_{AS} calculator

The following are some additional parameters used in the ADC calculation tool:

- F_{ADCK}: ADC input clock frequency
- CYC_{SMP_MIN}: Minimum sample cycles required for $T_{SMP} > T_{SMP\_REQ}$
- CYC_{SMP_USER}: Sample cycles set by the user using CMDHn[STS]
- T_{SMP}: Sample time set by the user

10 Acronyms

Table 2 lists the acronyms used in this document.

Table 2. Acronyms

Acronym	Description
ADC	Analog-to-Digital Converter
CV	Compare value
CVH	Compare Value High
CVL	Compare Value Low
DAC	Digital-to-Analog Converter
DFT	Discrete FFT
DMA	Direct Memory Access
DNL	Differential nonlinearity
ENOB	Effective number of bits

Table 2. Acronyms...*continued*

Acronym	Description
FFT	Fast Fourier Transform
FIFO	First in, first out
INL	Integral nonlinearity
IP	Intellectual property
MD	Mid Drive
MS/s	Megasamples per second
OD	Over Drive
PWM	Pulse Width Modulation
RMS	Root mean square
SAR	Successive Approximation Register
S/H	Sample and hold
SINAD	Signal-to-noise-and-distortion ratio
SNR	Signal-to-noise ratio
STS	Sample Time Select
THD	Total harmonic distortion
THD+N	Total harmonic distortion plus noise

11 Revision history

Table 3 summarizes the revisions to this document.

Table 3. Revision history

Document ID	Release date	Description
AN14679 v.1.0	15 July 2025	Initial public 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 https://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 document 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.

HTML publications — An HTML version, if available, of this document is provided as a courtesy. Definitive information is contained in the applicable document in PDF format. If there is a discrepancy between the HTML document and the PDF document, the PDF document has priority.

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.

NXP B.V. — NXP B.V. is not an operating company and it does not distribute or sell 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.

Contents