Application Report
SPNA061 − August 2004

ADC Source Impedance

Jim Childers *TMS470 Microcontroller*

ABSTRACT

Unbuffered multiplexed ratiometric analog-to-digital converters have strict requirements on driving source impedance, which are not always obvious. This application report addresses the trade-offs between source impedance and sample rate. It includes examples using the TMS470R1x family of processors in the TSC5000 process node (F10/C10). With adjustments to maximum clock rates, this application note is also applicable to the GS30 process node (F05/C05).

Contents

1 Introduction . . . 2
2 Terms in This Document . . . 2
3 System Model . . . 3
4 ADC Input Model . . . 4
5 External Components . . . 4
6 Symptoms of High Source Impedance . . . 5
7 The Effect of Cext . . . 6
8 Calculating Cext . . . 8
9 Calculating Rsource . . . 10
10 Consequences of High Source Impedance . . . 12
11 Solutions . . . 12
12 Conclusions . . . 13
13 References . . . 13

List of Figures

Figure 1. ADC System . . . 3
Figure 2. Input Path of TMS470R1x Multiplexed, Unbuffered ADC . . . 4
Figure 3. Typical Circuit for External Components With ADC . . . 5
Figure 4. Adequate Impedance . . . 5
Figure 5. High Impedance . . . 5
Figure 6. Zoom In on Figure . . . 6
Figure 7. Settling time of Csamp vs. Cext, With Rsource = 1160 W . . . 7
Figure 8. Time-Domain Plot of Four SPICE Runs . . . 8

Figure 9. Rsource 10
Figure 10. Rsource vs. Cexternal 11

1 Introduction

Unbuffered multiplexed ratiometric ADCs are commonly used in microprocessors because of their simplicity of design and inherent absence of circuits that need trimming in production. Both multiplexed and unmultiplexed versions are commonly found in discrete form, and are probably the most common ADCs in existence. They have no internal buffer amplifiers to introduce input offset and gain errors, and no internal voltage references which might induce scaling errors.

Designers use these ADCs for a variety of low frequency applications. Over the last two decades, they have been included in virtually every microprocessor family from every company and have increased in conversion speed right along with the microprocessors that host them.

A disadvantage is that the sample capacitor within the ADC is directly charged by the external signal, and ever-increasing speed has made this a growing issue. While it may seem like a trivial problem to charge a 20 pF sample capacitor, at high conversion speeds it can be difficult to charge it to within 1/2 the least significant bit (LSB) in the allotted time.

Also, the charge left on the sample capacitor by the previous conversion of a channel can affect the accuracy of the channel currently being converted if inadequate settling time is allowed for a given source impedance. This phenomenon is referred to as channel-to-channel crosstalk.

2 Terms in This Document

ADCLK	The internal clock of the TMS470R1x ADC. The period of this clock is an integer multiple of the peripheral clock period, ICLK. This value is programmable from the ADC registers.
ICLK	The TMS470R1x peripheral clock which drives the ADC. The ICLK period is limited to a minimum of 40nS (maximum of 25 MHz).
Sample Time	The time during which the ADC's sample gate is open for charging the internal sample capacitor, Csamp. In the TMS470R1x ADC, this time is software selectable from the ADC registers as either 2, 8, 32, or 128 ADCLK periods.
Conversion Time	The time required for a single channel to be converted. It is the sum of the sample time plus ten ADCLK cycles; therefore, conversion time is either 12, 18, 42, or 138 ADCLK cycles long, depending on the value set for sample time for the TMS470R1x.
Group Conversion	In the TMS470R1x, a user-programmed autonomous sequential conversion of all selected channels in a group. Group conversions are set up and initiated by software. They may be programmed to run only once or continuously.
Group Cycle Time	Time measured from the start-of-conversion of channel [N] to the start of the next conversion of the same channel [N].
Channel Sample Frequency	This is the frequency at which a single channel is sampled and is equivalent to the reciprocal of the group cycle time.

3 System Model

To start with, we should examine the overall environment in which the ADC is used. A model of the ADC system should include everything from the sensor or signal source to the ADC input itself. Figure 1 partitions the system into four distinct blocks which we can discuss individually.

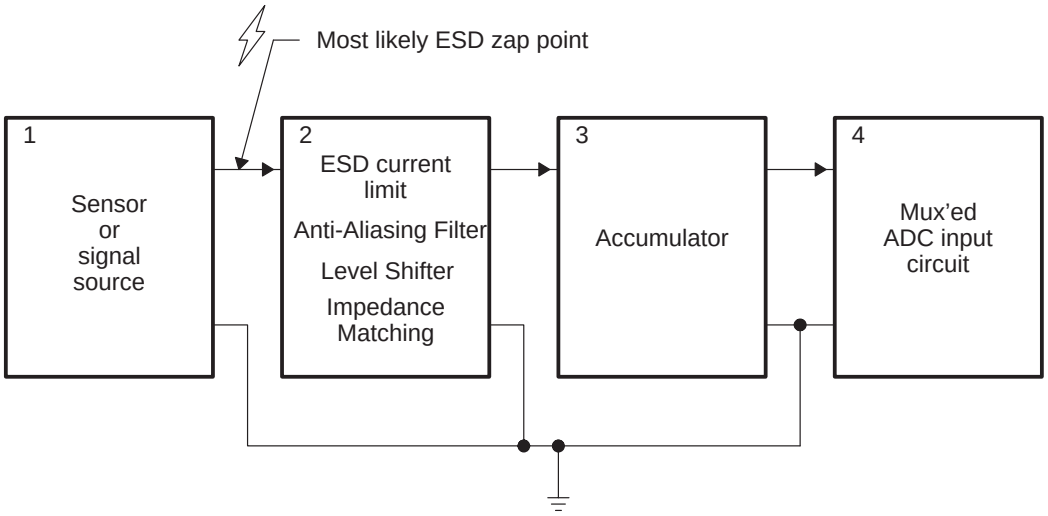

Figure 1. ADC System

Block 1: The sensor can be virtually anything from a sophisticated mass air flow sensor to a brick striking a piezoelectric crystal. As such, the source voltage can range from microvolts (as from a thermocouple) to several thousand volts (the brick/crystal). The source impedance and frequency can range similarly. With this in mind, we cannot say much about the source except that it clearly sets the requirements for the input of Block 2.

Block 2: This might best be described as a matching circuit. It has many simultaneous requirements to fulfill as noted in the figure.

It must maintain at least enough series resistance between the electrostatic discharge (ESD) entry point (if applicable) and the ADC input pin to protect the input from being damaged. For example, to pass the 4KV Contact Model ESD test, about 3000 Ω minimum resistance is required between the zap entry point and the ADC pin.

Any time something is digitized, it is essential that no information above the Nyquist frequency greater than a no effect level (−66dB for 10 bits) be introduced into the sampled signal. Once that noise is digitized, it is indistinguishable from the desired signal, so it must be small. Therefore, the cutoff frequency of a low-pass filter (anti-aliasing filter) must be strategically positioned between the desired maximum signal frequency, f, and the ADC's sampling frequency, fs. This filter is optional in some cases since some things do not change very fast, for example, the output of a thermistor.

A level shifter is often required to match the peak signal level of the input signal to the nominal 3.3 V swing of the ADC's input. This circuit may be as simple as two

resistors acting as a voltage divider, an active circuit like a voltage amplifier, or a sophisticated automatic gain control (AGC) circuit such as that used with a variable reluctance speed sensor.

Impedance matching is often necessary to match a higher impedance sensor or level shifter to the requirements of Block 3 or 4. The impedance requirement of Block 3 or 4 for a given channel is dictated by the sampling frequency, fs, of that channel. While the previous three items in Block 2 are generally well understood by designers; the true requirements for source impedance to the ADC inputs are sometimes misunderstood. Understanding the ADC's source impedance requirements is the focus of this application note.

Block 3: This block is optional depending on required speed, cost, and other factors. If it exists, it is simply a capacitor. We refer to it here as an *accumulator* because it accumulates charge in continuous time, which can then be charge-shared with the ADC's sample capacitor during the discrete-time sampling of that channel.

Block 4: This is the ADC. Since the ADC is a single converter time-multiplexed with up to sixteen input channels, it demands more attention at design time than if it was a converter-per-channel.

4 ADC Input Model

Starting with Block 4, Figure 2 shows a simplified model of the input path of the TMS470R1x's multiplexed, unbuffered ADC.

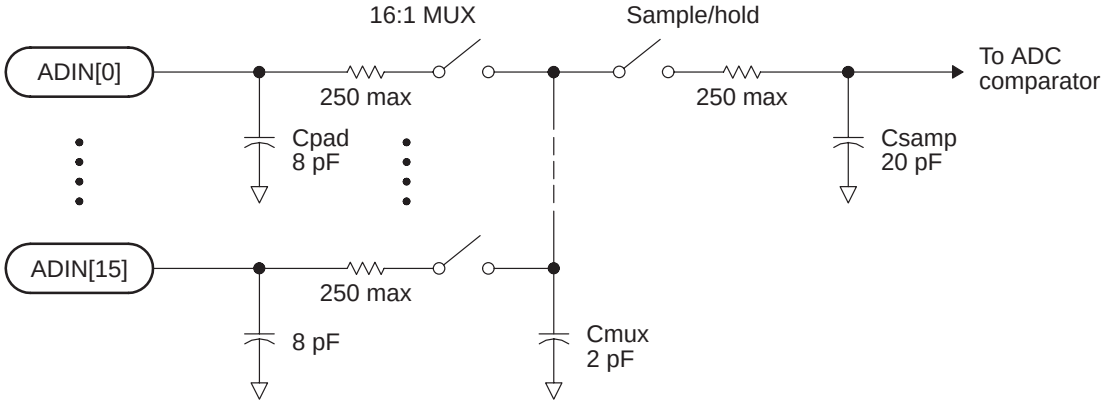

Figure 2. Input Path of TMS470R1x Multiplexed, Unbuffered ADC

There are two CMOS switches in the path between the ADINx pin and the sample capacitor, Csamp. The first is a 16-to-1 multiplexer that selects the channel to be converted. The second is the sample-and-hold gate that is controlled by the ADC's successive approximation state machine.

5 External Components

As we discussed for Blocks 2 and 3, it is common practice to add external components to the ADINx pins that scale and filter the signal from the analog source. These components are determined by the requirements set by Blocks 1 and 4. A fairly typical circuit is shown in Figure 3.

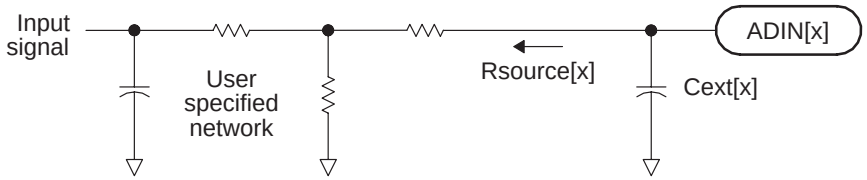

Figure 3. Typical Circuit for External Components With ADC

Generally, most designers place a large capacitor (Block 3) from the ADC pin to ground (Cext[x] in Figure 3). This capacitor is used to lower the source impedance of the channel as seen by the ADC so that the internal 20 pF sample capacitor can be charged quickly. As noted, this is a charge-sharing process between Cext and Csamp (refer to Figure 2 and Figure 3), whose RC time constant is primarily determined by the maximum ADC input resistance (500 Ω) and maximum sample capacitance (20 pF) of the ADC.

What is obvious about the above figure is that as Rsource is increased, the cutoff frequency created by Rsource and Cext will be lowered. What may not be obvious is that the values of Rsource and Cext are dictated by an individual channel's sample frequency, *not by the desired cutoff frequency* for that channel. This is simply a side effect of multiplexing the ADC inputs.

6 Symptoms of High Source Impedance

In Figure 4, Figure 5, and Figure 6, we fake oscilloscope pictures to demonstrate the effect of high source impedance. For example, let us say there are two channels that you are converting in sequence with a 10 µS group cycle time. The first channel has a 100 Hz square wave on it, and the second channel is a DC signal.

For the first oscillograph, shown in Figure 4, let us assume the impedance is adequate for the chosen sample frequency. In other words, this is what you expect to see on the two ADIN pins when everything goes right:

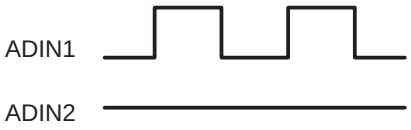

Figure 4. Adequate Impedance

Now let us increase the impedance by 2 or 3 orders of magnitude. After all, these two signals are very low frequencies, 100 Hz and DC. Why *should* they have a low impedance? Figure 5 shows the results.

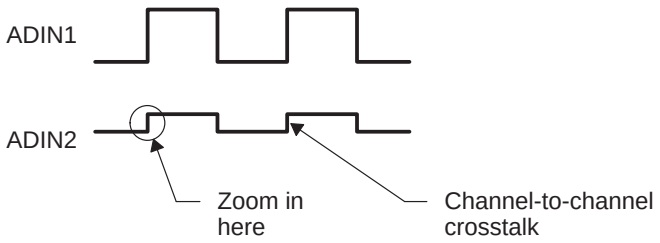

Figure 5. High Impedance

The first waveform has lost some bandwidth, so the corners are not quite so square. Perhaps this is not an issue if we are only interested in its min and max values.

But the second waveform has picked up crosstalk from the previous channel. If it was intended to be a DC level, this signal has been rendered almost useless. What is the source of the problem? Earlier we mentioned that the RC time constant set by Rsource*Cext is a function of channel sample frequency, not signal frequency. Here is how you can see this phenomenon in action. Figure 6 is a zoom-in on Figure 5 on the lower trace's rising edge:

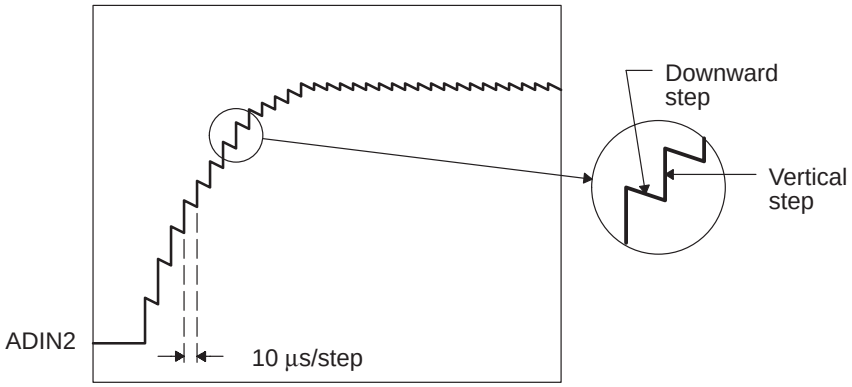

Figure 6. Zoom In on Figure 5

You should see a saw-tooth pattern that makes up the transition edge. Actually in this example, the ratio of sample-frequency to signal-frequency is 1000:1 so probably there are a hundred or so steps rather than the dozen shown. As shown, the step spacing will be at the channel cycle rate. In this case, 10 µS. The vertical steps are caused by residue from the conversion of ADIN1 left on Csamp thus creating a very small undesirable offset in Cext[2] during the conversion of ADIN2. The downward steps are caused by the source for ADIN2 attempting to recover the error via the source impedance. The difference between these two step amounts is the error that accumulates with time. As the error voltage across Rsource accumulates with each cycle, the error step becomes smaller until the vertical upward step and the downward step cancel each other.

On the oscilloscope, when zoomed out enough to see the 100 Hz waveform, the 100 KHz sampling artifacts of the crosstalk are completely invisible, and the basic exponential shape (in Figure 6) looks like a clean squarewave. In the general case, crosstalk on the current channel looks like a vertically scaled image of the previous channel's waveform.

7 The Effect of Cext

Now let us look at Block 3 and the rationale for selecting Cext, or for that matter even having Cext. To do this, we can use a SPICE model to try several values of Cext and measure the time at which the voltage on Csamp settles to within 1/2 LSB of the exact value (assuming 10 bits). Plotting a curve of settling time versus Cext may tell us something about the nature of Cext.

Figure 7 is a graph plotted from multiple runs of SPICE assuming two channels of the TMS470R1x in continuous conversion mode. A mid-range value of 1160 Ω was chosen for Rsource.

The leftmost point on the graph corresponds to a circuit with Cext = 0.1 pF (effectively zero). As Cext is increased, it can be seen that the required settling time gets worse until Cext is around 20 nF. Then there is a sharp roll-off in settling time until Cext is around 41 nF, at which point the slope of the graph settles to near zero. This graph clearly illustrates that there is an optimum value of Cext. In the next section, *Calculating Cext*, we will show how this optimum value is derived, but for now we need to understand the shape of the curve in Figure 7.

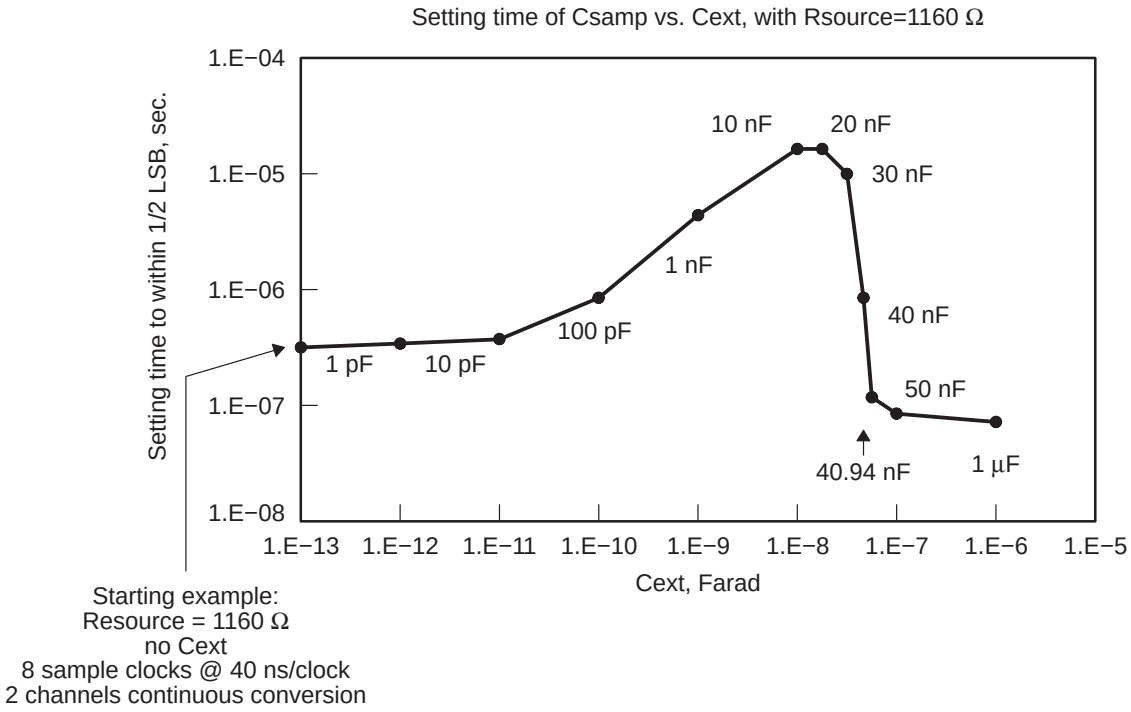

Figure 7. Settling time of Csamp vs. Cext, With Rsource = 1160 Ω

Figure 8 is a hand-drawn time-domain plot of four SPICE runs like that made for the graph in Figure 7. These may help demonstrate the reason for the sudden drop in settling time as Cext increases. However, the waveforms have such hugh scale differences that not even a log-log graph does an adequate job of placing them on the same plot, so there is some graphic license taken in Figure 8.

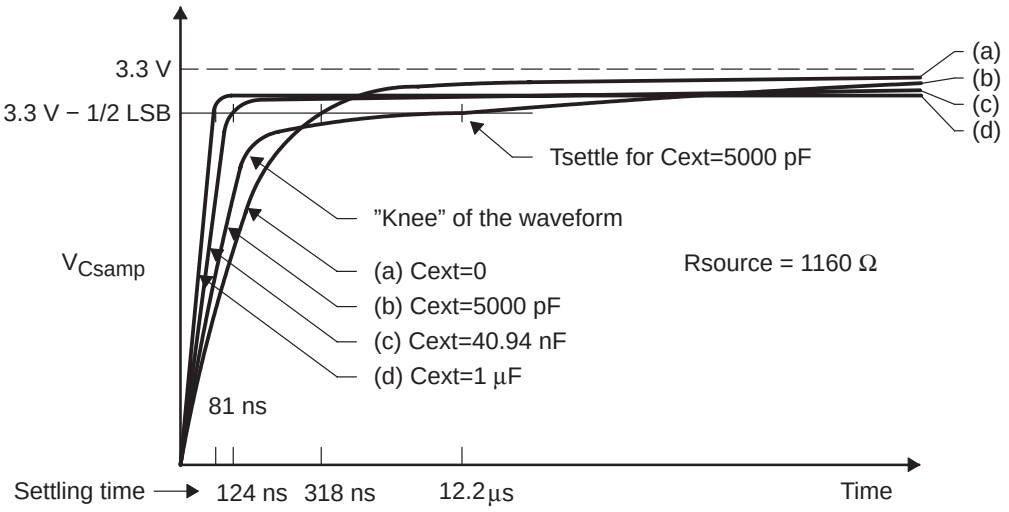

Figure 8. Time-Domain Plot of Four SPICE Runs

The object is to get Csamp charged to within 1/2 LSB of 3.3 V before declaring Csamp settled. Note that as Cext becomes much larger than Csamp, the curve looks more like two straight lines joined by a knee. As Cext gets larger, the knee gets sharper.

Below are the dependencies of each of the curves above:

(a) With Cext = 0, there is effectively no discernible knee. The curve is a simple RC made up of (Rsource + Rmux)*Csamp.

(b, c, d) For the other three curves, charge sharing jumps the voltage up to the knee with an RC time constant dominated by Rmux*Csamp. Then from the knee on, the RC time constant is dominated by Rsource*Cext.

(c, d) As the knee rises above the 3.3 V−1/2 LSB line with increasing Cext, there is a rapid reduction in time required for Csamp to settle, because being above the 1/2 LSB line is the very definition of settled.

The vertical segment is dominated by charge sharing between Cext and Csamp, while the horizontal segment is dominated by Cext recovering via Rsource. If, after charge-sharing, the knee falls short of the 3.3 V−1/2 LSB line, then it can take a very long time to finish charging Cext and Csamp the rest of the way. However, if the knee occurs *at* or *above* the line, then charging Csamp is already done, and all that remains is to complete the recharging of Cext. However, once Csamp is settled, the A-to-D conversion can proceed and we have a full group cycle time to complete recharging Cext.

8 Calculating Cext

Let us look at why Cext should be greater than or equal to about 41nF. To do this we need to examine the charge sharing between Cext and Csamp. Note that we will use only Csamp (20pF), not Cpad + Cmux + Csamp (30pF) for this calculation since Cpad (8pF) capacitance is in parallel with Cext, and Cmux (2pF) is negligible (refer to the ADC model earlier).

Recalling that *conservation of charge* says something like “the total charge *before* sharing is equal to the total charge *after* sharing”:

Before charge sharing,

$$Qsamp = Csamp \times Vsamp \text{ and } Qext = Cext \times Vext \quad (1)$$

Conservation of charge,

$$Qfinal = Qsamp + Qext \quad (2)$$

Capacitors in parallel add,

$$Ctotal = Csamp + Cext \quad (3)$$

After charge sharing,

$$Qfinal = Ctotal \times Vfinal \quad (4)$$

Substituting [1], [2], and [3] into [4],

$$(Qext + Qsamp) = (Cext + Csamp) \times Vfinal \quad (5)$$

Solving for Vfinal,

$$Vfinal = \frac{Qext + Qsamp}{Cext + Qsamp}, \quad (6)$$

Substituting [1] into [6],

$$Vfinal = \frac{Cext \cdot Vext + Csamp \cdot Vsamp}{Cext + Qsamp} \quad (7)$$

where Vfinal is the voltage remaining on Cext after charge sharing with Csamp.

It can be seen that for a 10-bit ADC settling to within 1/2 LSB, Vfinal would have to be:

$$Vfinal = \left(Vin - \frac{Vin}{2^{(10+1)}}\right) = 0.9995117 \times Vin \quad (8)$$

where Vin is the desired input signal value.

In a nominal 3.3 V system, this amounts to a worst-case value of 1.62 mV.

Assume that Csamp is discharged, Cext holds the value Vin, and Vfinal must end up at 0.9995117 x Vin (that is, to within 1/2 LSB of Vin):

From equations 7 and 8,

$$Vin \times \left(1 - \frac{1}{2^{10+1}}\right) = \frac{(Cext \cdot Vin + Csamp \cdot 0)}{(Cext + Csamp)} \quad (9)$$

$$(Cext + Csamp) \times \left(1 - \frac{1}{2048}\right) = Cext \quad (10)$$

Solving for Cext,

$$Cext = 2047 \times Csamp \quad (11)$$

For TMS470R1x ADC,

$$Cext = 2047 \times 20\ pF = 40.94\ nF \quad (12)$$

This is a minimum value of Cext. Larger values are all right, but have no effect on sample time and limited effect on group cycle time. This is discussed further later.

Having Cext < 2047*Csamp requires that Csamp be charged entirely during the *sample time* rather than during the *group cycle time*. So Cext < 2047*Csamp is essentially a different mode of operation from Cext > 2047*Csamp.

From the standpoint of speed and Rsource requirements, you are actually better off with no external cap if Cext < 2047*Csamp (assuming there is no anti-aliasing filter). This was shown dramatically in the graphs in Figure 4 through Figure 6.

9 Calculating Rsource

The total resistance feeding the external capacitor, Cext, is called Rsource. Namely, it is the Thevenin equivalent resistance of the driving source as viewed by Cext.

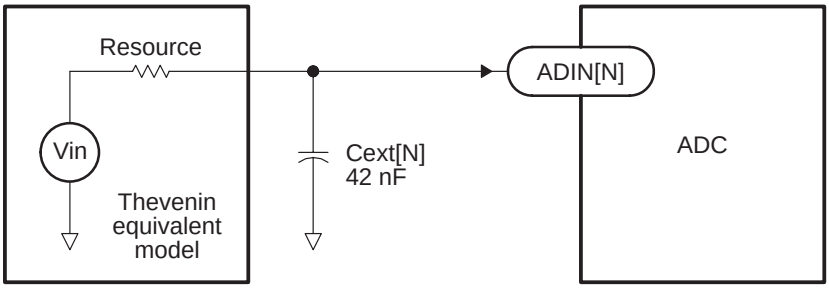

Figure 9. Rsource

The time constant required for an RC circuit to settle to within 1/2 LSB with 10 bits of resolution is:

$$\gamma = \ln(2^{(10+1)}) = 7.62 \text{ time constants} \quad (13)$$

Given a group cycle time, Tcyc, the value of Rsouce required to replenish the charge depleted from Cext is given by the relationship:

$$Rsource < \frac{Tcyc}{\gamma \times Cext} \quad (14)$$

As an example, assume that you are running only two channels in continuous mode with a 25 MHz ICLK, the ADCLK divider is set to divide-by-one (ADCLK = 40 nS), and the number of sample clocks is set to two. This means we have a conversion time of 12 clock cycles per channel (12 = 2 + 10), which is 24 cycles for two channels (Tcyc or group cycle time); therefore, Tcyc = 960 nS.

If we have followed the rules and made Cext at least 40.94 nF, then the source resistance from equation 14 is:

$$\text{Rsource} < \frac{960\ \text{ns}}{7.62 \times 40.94\ \text{nf}} = 3.08\Omega \quad \textbf{(15)}$$

Intuitively, this sounds very low and thus suspect; however, look at the voltage across the charging resistor, Rsource. One side is at Vin while the other side must be maintained within Vin ± 1/2 LSB to avoid error. So the maximum voltage drop across the resistor is limited to only 1/2 LSB. As noted above, 1/2 LSB amounts to 1.62 mV. You cannot charge much, even through a 3Ω resistor, with a voltage difference of only 1.62 mV.

Earlier we mentioned that making Cext >> 2047*Csamp has limited benefit. To bear this out, the graph in Figure 10 compares Rsource versus Cext. It shows Cext increased from 2047*Csamp (the minimum) to about three orders of magnitude larger.

As Cext increases exponentially, Rsource improves asymptotically to a maximum of 7.62 times the resistance of when Cext was a minimum (Cext_min = 2047*Csamp). The number 7.62 should be familiar as we developed it in equation (13). This graph makes it clear that there is little reason to increase Cext beyond 10*Cext_min.

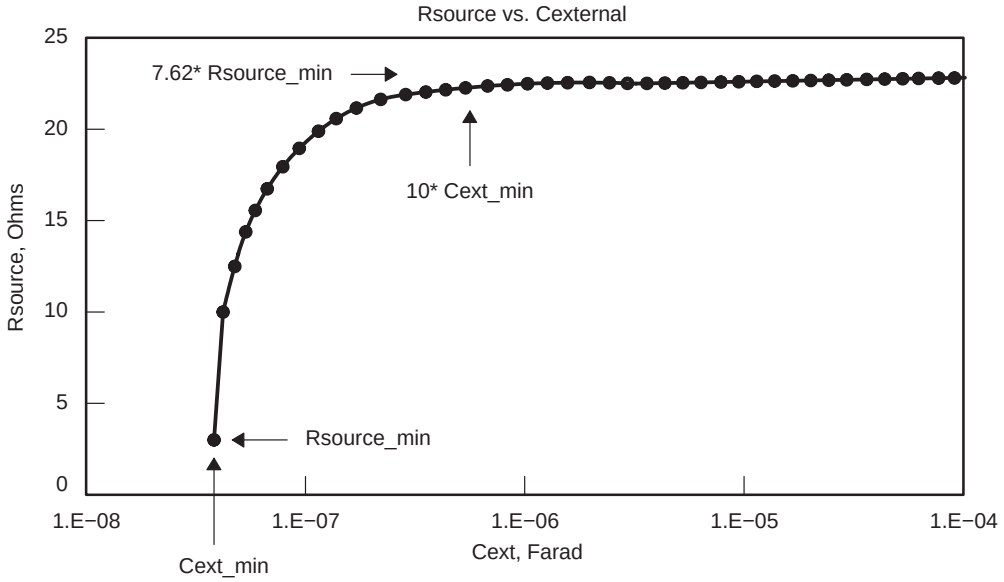

Figure 10. Rsource vs. Cexternal

For reference, in Figure 10, each incremental point on the horizontal axis is a 15% increase in Cext.

10 Consequences of High Source Impedance

If the impedance is too high, the most obvious consequence is response time. That is, the measured value will not be what you thought it should be.

The second, more insidious consequence of high Rsource is channel-to-channel crosstalk. As stated before, this is charge transferred from one channel to the next, and accumulated over many conversion cycles. With crosstalk, each channel disturbs the *next* channel in the group to be converted. In a 16 channel group conversion, channel 0 disturbs channel 1, channel 1 disturbs channel 2, ..., channel 15 disturbs channel 0. This phenomenon is due to the residue of charge on Csamp after converting channel [N] which contaminates channel [N+1].

If the source impedance is low enough, this effect is less than 1/2 LSB, so who cares. But as Rsource increases, more and more of the contaminated signal is allowed to accumulate on channel [N+1]'s external capacitor, $Cext_{[N+1]}$. With each conversion loop, $Cext_{[N+1]}$ is booted (up or down) by a few microvolts that accumulate. Eventually channel [N+1] stabilizes with a new (but incorrect) value.

11 Solutions

Practically, there are only a few possible solutions to this problem:

1. Lower the source impedance, Rsource. This can be done in a variety of ways depending on the type of signal source. One good way to lower the Rsource is to use an op amp. This is an almost universal solution since you can match any impedance to any other impedance with proper op amp trickery. However, of course there is the cost, complexity, board area, etc. Sometimes a different interface circuit will help, or the sensor might be designed with a lower output impedance.

2. Accept a lower resolution result. For each bit of resolution lost, the required settling time is cut in half. For many applications, it is not necessary to know a level to within 3.2 mV (1 LSB @ 10 bits). Accepting a lower resolution result may require only software changes, or possibly no changes at all.

3. Break up the channels into two or more groups: channels that can tolerate a slower sample rate in one group, and channels that must be converted faster in another group. This is only software for the slower channels. The higher speed channels may require hardware to reduce impedance.

4. Slow down ADCLK to provide more settling time. This should be only a software change, but may be more extensive than reprogramming the ADCLK divisor value.

Some experts have suggested that the ADC should be modified to discharge Csamp to ground between conversions; however, this would only hide the problem. In this case, the crosstalk would always be from a known value (zero), but would result in a gain error that *varies with changes in source impedance*, which in turn may vary with sensor position, creating nonlinearities.

12 Conclusions

Unbuffered multiplexed ratiometric ADCs are excellent in terms of cost and produceability, but careful consideration must be used when designing with them to obtain the expected results.

Most notable are the following points:

- Source impedance issues can be easily diagnosed by examining the waveforms at the ADINx pins while the ADC is running in a continuous loop.
- In most cases, the best speed/impedance results will be obtained by including Cext if the proper value is selected.
- Given a specified number of bits of resolution, Cext can be calculated independent of frequency and source impedance.
- The product of Rsource*Cext is a function of sampling frequency, not input filter bandwidth.
- Several remedies exist for correcting designs that exhibit excessive source impedance.

13 References

1. *SE470R1B31B 16/32-Bit RISC Flash Microcontroller Datasheet*, (literature number SPN064).
2. *Advanced CMOS Logic Data Book*, (literature number SCAD001C).

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third-party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

Following are URLs where you can obtain information on other Texas Instruments products and application solutions:

Products		**Applications**	
Amplifiers	amplifier.ti.com	Audio	www.ti.com/audio
Data Converters	dataconverter.ti.com	Automotive	www.ti.com/automotive
DSP	dsp.ti.com	Broadband	www.ti.com/broadband
Interface	interface.ti.com	Digital Control	www.ti.com/digitalcontrol
Logic	logic.ti.com	Military	www.ti.com/military
Power Mgmt	power.ti.com	Optical Networking	www.ti.com/opticalnetwork
Microcontrollers	microcontroller.ti.com	Security	www.ti.com/security
		Telephony	www.ti.com/telephony
		Video & Imaging	www.ti.com/video
		Wireless	www.ti.com/wireless

Mailing Address: Texas Instruments
Post Office Box 655303 Dallas, Texas 75265