

On the Programmability and Reconfigurability Capabilities of ADCs

Athon Zanicopoulos, J.A. Hegt and Arthur van Roermund
Mixed-signal Microelectronics Group, Eindhoven University of Technology,
Eindhoven, The Netherlands, email: a.zanicopoulos@tue.nl

Abstract—During the last years, there has been an evolution in wireless communication systems towards multifunction and multistandard terminals, supporting several standards, like GSM, Bluetooth, DECT, wireless LANs and UMTS. The implementation of a "universal terminal", which supports all these standards and allows wireless connectivity and roaming, becomes necessary for the reduction of the cost and the increase of the versatility, [1]. This paper presents the exploration of the Programmability and Reconfigurability (P/R) capabilities of the ADC architectures, keeping focus on ADCs that are commonly used in the telecommunication applications, i.e. Flash ADCs, Folding and Interpolating ADCs, Pipelined ADCs and $\Delta\Sigma$ ADCs. We present the recommended performance area of each architecture and we investigate their P/R capabilities. Our target is the coverage of a large part of the recommended performance area of each architecture, if not the whole, with only one programmable representative.

I. INTRODUCTION

Nowadays, we witness an exponential growth in the portable communication market. This growth dictates the need for low-cost, small-form and low-power receivers, since more and more systems use radio links. The consumer needs both the convenience of added connectivity and the benefit of the additional services. In parallel there is a fast growing number of standards like Wireless Local Area network (WLAN) and Wireless Personal Area Network (WPAN) that are based on IEEE 802.11x (x=a,b,d,g,e), Bluetooth, GSM, UMTS, GPRS, etc. The straightforward approach is to intergrade multiple links, one for each standard. The disadvantage of this approach is that the cost of the system can be high due to the large area needed to integrate all the individual receivers. Additionally, for every new standard, a new receiver should be implemented, increasing further the cost due to the longer design time. Therefore, the need for smaller and cheaper receivers dictates that we have to shift from the conventional single-standard receiver to a multistandard one. This means that a single receiver should be able to handle more than one standard.

The final solution to this problem may be the Software Radio, which essentially consists of an antenna, an Analog to Digital Converter (ADC) and a Digital Signal Processor (DSP). All the signal processing needed for a given standard is performed in software. The main bottleneck of this approach is the complexity and the very high power consumption required by the ADC.

The work is sponsored by Stichting Technische Wetenschappen.

Consequently, an intermediate solution might be the multistandard terminals that consist of a single receiver, but the signal processing is done partly in the analog domain and partly in the digital (software) domain, so called Software-defined Radio. According to Software-Defined Radio Forum (www.sdrforum.org): "Software-defined radio is a collection of hardware and software technologies that enable reconfigurable system architectures for wireless networks and user terminals." This solution requires that the analog hardware can change its properties after the fabrication and it can adjust its functionality according to the desired standard.

In the second section of this paper, we present the most promising receiver architecture concerning the multistandard terminals. The third section presents the relative to the Programmability and Reconfigurability of ADCs definitions and proposes the most suitable ADC architectures, depending on the standards that they should cover. In the fourth section we propose Programmable and Reconfigurable options for every ADC's architecture, according to cover wider range of performance specifications and in the final section we draw the conclusions of the study.

II. DIRECT CONVERSION RECEIVERS

It has been shown that the most promising candidate for a multistandard receiver architecture is the Direct Conversion architecture [1]. This architecture minimizes the necessary off-chip components, hence it reduces the cost, and provides the needed multi-mode flexibility by constructing the Low Pass Filters (LPF) variable. There are two variations of this architecture: *i*) The Direct Conversion receiver with Zero Intermediate Frequency (Zero-IF) and *ii*) the Direct Conversion receiver with Low-IF.

In the Direct Conversion receiver with Zero-IF (fig. 1) all the potential in-band channels are translated from the carrier directly to baseband using a single mixer stage, before any channel filtering is performed. All the energy from undesired channels is removed by on-chip filtering at baseband.

The problem that associates with this architecture is that the Local Oscillator (LO) is at the same frequency as the RF carrier and this may lead to LO leakage either to the mixer or to the antenna. This results to a time-varying DC offset that reduces the dynamic range of the receiver. Hence, a DC offset calibration method should be employed.

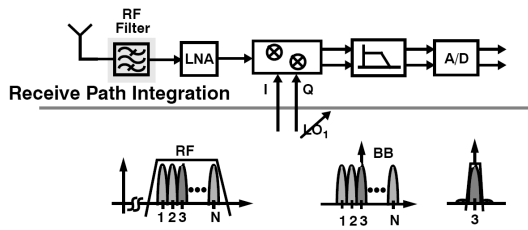


Fig. 1. Direct Conversion receiver with ZIF

In the Direct Conversion receiver with Low-IF (fig. 2) a similar conversion is performed, but now the single mixer is used to translate all the in-band channels to a low intermediate frequency.

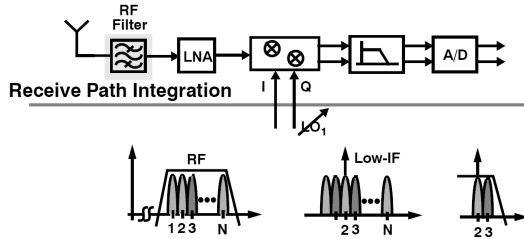


Fig. 2. Direct Conversion receiver with Low-IF

Typically, this IF is of the order of magnitude of one or two channel bandwidths [2]. One of the problems with the Low-IF receiver is that the desired carrier is down-converted to a low-IF and hence, an image-rejection technique should be applied. Additionally, this receiver architecture requires higher ADC performance with respect to bandwidth and resolution.

According to take a decision for which architecture is more suitable for a multistandard terminal, we have to take in account the standards that we want to implement. It has been proved [1] that the Zero-IF architecture is more suitable for some standards, like WLAN and UMTS, when the Low-IF is more suitable for some other standards, like GSM and Bluetooth. Therefore, a flexible receiver that implements both the Zero-IF and Low-IF seems the most promising solution.

It is clear that all the blocks of the multistandard terminal, RF filter, Low Noise Amplifier (LNA), Mixer, LPF and ADC should have more or less some Programmable and/or Reconfigurable (P/R) options. In the rest of this paper we concentrate on the P/R capabilities of the ADCs and we discuss the potential alternative implementation solutions.

III. PROGRAMMABILITY AND RECONFIGURABILITY IN ADCS

A. Definitions

Hereby we give some definitions concerning the programmability and reconfigurability (P/R) in the ADCs.

Performance space is the space produced by the performance merits *speed*, *accuracy*, *power consumption*. The *area* has been excluded from the performance merits, since in this study we concern about P/R ADCs and it is impossible to

change the area of an ADC after its fabrication. By using them as unity vectors, we can write:

$$\overrightarrow{Performance} = (a) \cdot \overrightarrow{speed} + (b) \cdot \overrightarrow{accuracy} + (c) \cdot \overrightarrow{power} \quad (1)$$

The weights a , b and c are factors that depend on the given ADC. Therefore, every ADC, programmable or not, can take place in the performance space, as it is depicted in figure 3.

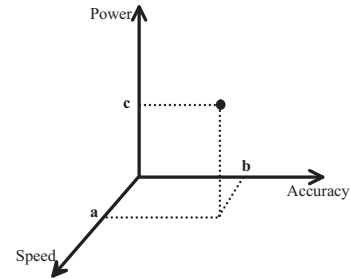


Fig. 3. Performance space

The points in the performance space can be continuous or discrete. By using the performance space, we can define the **Flexibility** as the ability of a system to place itself in a specific point of the performance space, under the action of a given set of *control signals*. These signals can be provided by the end-user based on *a priori* knowledge or can be generated by the system based on *a posteriori* knowledge. The term flexibility can be divided in two terms, namely *Programmability* and *Reconfigurability*. **Programmability** is the ability of a system to change at least one of its parameters (for example the bias current) and **Reconfigurability** is the ability of a system to change its configuration in architectural/circuit level.

The usefulness of a P/R ADC lies on the fact that it can be adjusted to possess more than one point in the performance space. That means one ADC can cover N points. On the contrary, a conventional (non-flexible) ADC possesses a specific point in the performance space, meaning that we need N systems to cover N points. In a flexible ADC, the only concern of the end-user is the position of the system in that space.

B. Identification of the performance space and the ADCs' architectures

According to decide which of the ADC architectures are more useful and worthy to add flexibility options to them, we should take into account the specification requirements of the mobile telecommunication standards with respect to the ADC. The following figure (fig. 4) presents the specifications and the standards from the ADC point of view.

We can observe that most of the mobile telecommunication standards are clustered in the range of resolutions between 8bits and 14bits. The figure below (fig. 5) gives us the speed and resolution area that can be optimally covered by each of the commonly used ADC's architectures.

By combining this two figures, we can identify the ADC architectures that are commonly used in mobile telecommunication applications and, hence, we should study their

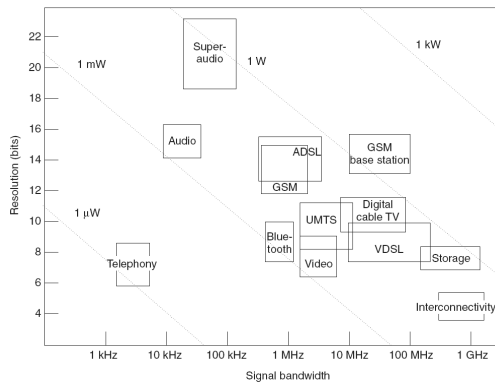


Fig. 4. Resolution vs Bandwidth for the telecom standards

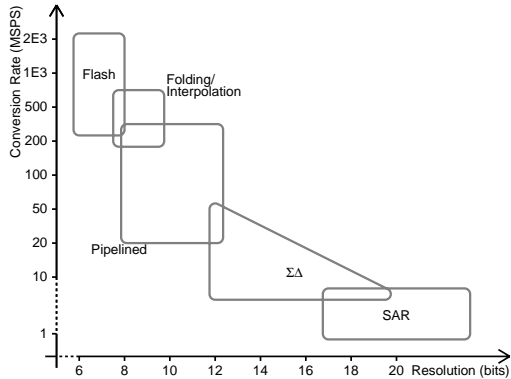


Fig. 5. Speed and resolution of the common ADC architectures

programmability and reconfigurability capabilities. They are, namely:

- Folding/Interpolating ADCs
- Pipelined ADCs
- $\Delta\Sigma$ ADCs

Additionally, it is worthy to include in our study the Flash ADC, since it is an essential building block of the aforementioned architectures.

IV. IMPLEMENTING PROGRAMMABILITY AND RECONFIGURABILITY IN ADCs

By identifying our target ADC architectures we reach the question of which functions of each architecture could be programmable and/or reconfigurable. The criteria according to choose flexible techniques are the maximum coverage of performance space by the same P/R ADC and the minimum performance penalty due to the introduction of these techniques.

A. Flash ADCs

The following figure (fig. 6) presents a typical flash ADC. It consists of a reference ladder, rows of preamplifiers and comparators and a encoding logic. The preamplifiers amplify the difference between input and reference voltage and the comparators resolve the thermometer code. For a N bits we need $2^N - 1$ of such rows. A common technique to reduce the number of the input preamplifiers and, hence, to reduce the

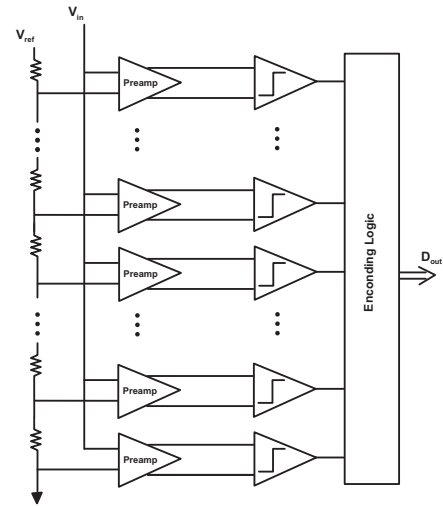


Fig. 6. Typical Flash ADC

input capacitance, is the use of *interpolation*. The following figure 7 depicts two methods of interpolations that essentially serve the same function.

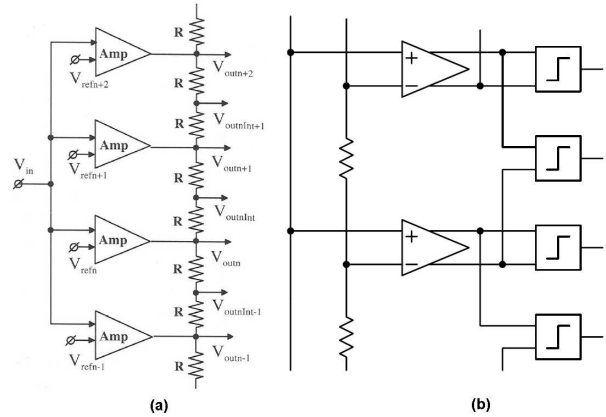


Fig. 7. (a) Resistive and (b) latch interpolation

How extensive is the interpolation is quantified by the *Interpolation Factor*, (F_{INT}). Interpolation factor of two, means that we double the signals after the preamplifiers using interpolation. In such a way we increase the resolution of the ADC by one bit. Additionally to the reconfigurable interpolation we need to implement programmable gain preamplifiers, since unsaturated analog outputs are necessary for the proper operation of the latches. By doubling the number of latches the speed reduces approximately by a factor of two. Therefore, in the Flash ADC architecture we can identify the use of reconfigurable interpolation and programmable preamplifiers' gain as techniques that can lead to a P/R ADC [3].

B. Folding/Interpolating ADCs

The Folding/Interpolating ADCs have been proposed [4] as an alternative to flash architecture, according to alleviate the exponential increasing complexity and the high input

capacitance for resolutions higher than 6 or 8 bits. A typical representative of this architectures, which resolves 8 bits, is given in the following figure (fig. 8).

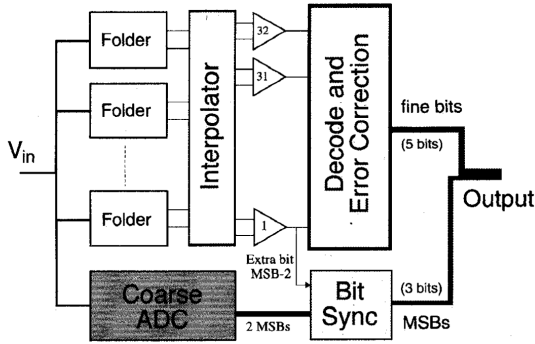


Fig. 8. Folding/Interpolating ADC

The distinctive blocks in this figure are the folding and interpolating blocks that perform an analog signal processing operation. The main advantage of the folding technique is the reduction of the number of the comparators. The interpolation technique, as in the case of the flash ADC, is used according to generate more signals without the use of more folders. The coarse ADC and the decoding block are consist by flash ADCs.

The figure below (fig. 9) shows a typical folding CMOS circuit.

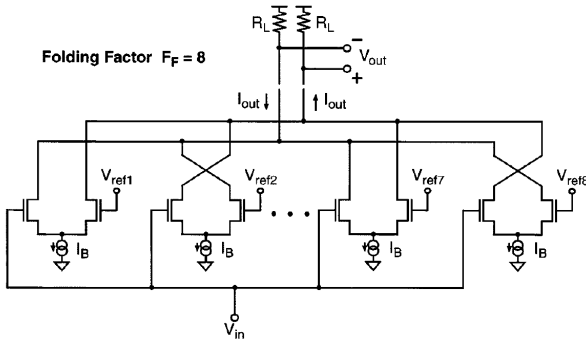


Fig. 9. CMOS Folding circuit

The level of folding is quantified by the *Folding Factor* (F_F), which in the figure above is equal to 8. Given the folding factor, the number of folding blocks (N_{FB}) and the interpolation factor (F_{INT}), we can prove a formula (eq. 2) that connects these quantities with the total resolution (N) of the ADC.

$$2^N = F_F \cdot N_{FB} \cdot F_{INT} \quad (2)$$

Therefore, by implementing these quantities programmable and/or reconfigurable, we have the ability to effect the ADC's resolution. For example, we can implement an 8 bit Folding/Interpolating ADC with $F_F=8$, $N_{FB}=4$ and $F_{INT}=8$, ($8 \cdot 4 \cdot 8 = 2^8$). By increasing the F_{INT} to 16 and using eight

instead of four folding blocks, we can achieve a resolution of 10 bits ($8 \cdot 8 \cdot 16 = 2^{10}$). The folding and/or the interpolation can be carried out in more than one step, in a cascade way, [5]. This does not influence the generality of the formula and the conclusions of this study.

Additionally, the aforementioned P/R options are necessary to be coupled with P/R flash converters (coarse and decoding blocks), employing techniques that has been described in the previous section.

C. Pipelined ADCs

The pipelined ADC architecture is one of the most successful architecture for telecommunication applications (fig. 10). It is an excellent combination of high-speed operation with low-power capabilities.

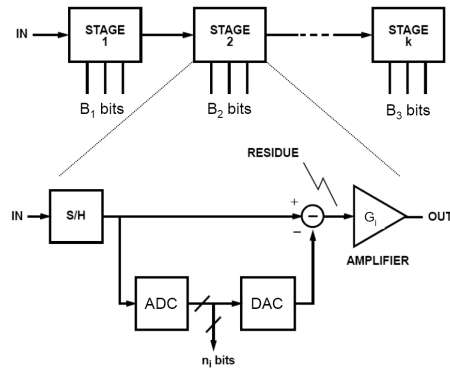


Fig. 10. Pipelined ADC

Furthermore, it provides high modularity and regularity that can be used according to implement P/R ADCs.

In [6] a flexible ADC is proposed, based on identical building blocks and an interconnection network (fig. 11). For the sake of clarity the figure depicts a 6x6 matrix, but its size can vary depending on the applications that it should cover. This design exploits the modularity of the pipelined architecture, along with its property that the number of the components grows linearly with the resolution.

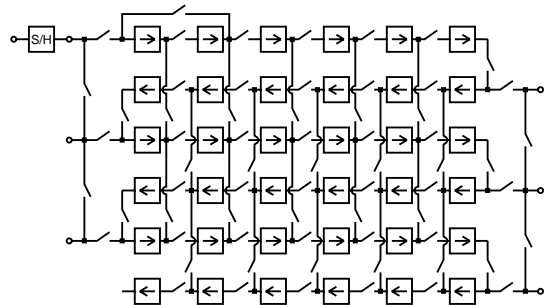


Fig. 11. Flexible ADC matrix

The key idea is the use of a number of building blocks in series and/or in parallel. By adding or removing building blocks, using switches, we can increase or decrease, respectively, the resolution. Using this matrix we can implement

several configurations, aiming for different performance requirements (speed/accuracy), namely: *i*) standalone ADC, *ii*) Time-Interleaved ADC, *iii*) Cyclic ADC and *iv*) Pipelined ADC employing averaging [6].

According to use effectively this reconfigurable matrix, we should employ several programmable techniques. The bias current of the building blocks' amplifiers can be self-adjustable, proportional to: *a*) the position of the block in the pipelined chain and *b*) the sampling frequency, as it has been proposed in [7]. Similar techniques can be applied to the sub-ADCs. Moreover, the building blocks that are not in use during the operation of a given configuration can be switched-off to save power.

D. $\Delta\Sigma$ ADCs

The $\Delta\Sigma$ ADC architectures offer means to exchange resolution in time for that in amplitude so as to avoid the design of complicate precision analog circuits [8]. This property becomes even more attractive for the future technologies, where the amplitude resolution is reducing in favor of the time resolution.

The ADC conversion can be performed in one step, by one noise shaper, or in more steps. For example, the following figure 12 depicts two cascaded noise shapers, implementing an architecture called MASH (multi-stage noise-shaping). The final output is formed by a linear combination of the individual noise shapers' outputs.

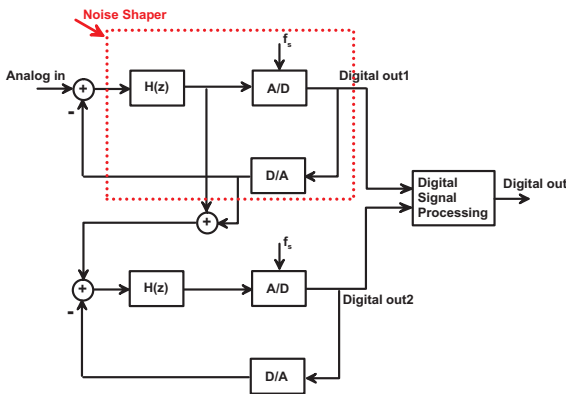


Fig. 12. MASH ADC

A noise shaper consists of a loop filter, $H(z)$, an ADC and a Digital to Analog Converter (DAC) in the feedback loop. In the $\Delta\Sigma$ ADC architecture we can identify four different P/R options, namely:

- 1) The OSR (Oversampling Ratio).
- 2) The order of the loop filter.
- 3) The number of bits in the ADC and DAC.
- 4) The number of cascaded noise shapers in MASH structures.

The following figure (fig. 13) shows the achievable performance for converters with a single loop [9]. $OnBm$ is an n^{th} order noise shaper with m -bit ADC and DAC. The graph shows clear that by increasing the OSR, the SNR increases

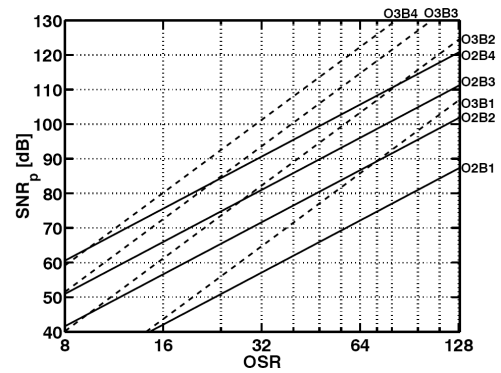


Fig. 13. Single- and multi-bit, single loop $\Delta\Sigma$ ADCs

too. The interest point is that the rise rate is steeper if we use higher order loop filter, since we have more aggressive noise shaping, but only for high OSRs (≤ 16). Furthermore, using more bits in the ADCs and DACs we expect improvement in dynamic range), or, alternately, for a given dynamic range an increase in bandwidth since we can reduce the OSR. These improvements stem from the higher allowed overload levels in comparison to the single bit noise shaper. Additionally, one more benefit of using multibit ADCs and DACs is that they improve the stability of the noise shaper.

Therefore, by using such graphs that covers three out of four of the above enumerated P/R options, we can decide the architecture and the combination of parameters that are needed to fulfill the performance requirements of a given multistandard terminal. On the top of these options, we have the potential of using more than one noise shaper in a cascaded way (in a similar way as in the pipelined ADC case). The big benefit is that we can achieve high order shaping by using unconditional stable loop filters, which is particular important when the stability is a major concern.

By employing the aforementioned options to implement P/R $\Delta\Sigma$ ADCs, we have to take into serious consideration the existing trade-offs.

- 1) For n additional bits of accuracy, we have to increased the OSR by 2^{2n} . Therefore, for high OSR we have to compromise the analog BW of the converter or we reach the technological limit in speed/accuracy.
- 2) Increasing the order of the loop filter, we compromise the stability resulting in serious degradation of the SNR.
- 3) Increasing the number of bits in the ADC we start having problems with the DAC's accuracy, since, while the ADC lies in the forward path and hence is less sensitive, the DAC operates in the feedback path. The linearity of the DAC should be at least as good as the linearity of the whole $\Delta\Sigma$ converter.
- 4) The number of cascaded noise shapers is limited by the required matching between the circuitry that implements the noise shapers (e.g. OpAmps). This happens because the noise cancellation occurs in the analog domain. In general, we cascade no more than two or three noise shapers.

In literature can be found numerous techniques that ease one or more of the above mentioned problems. For example, according to improve the linearity of the DAC we can employ Dynamic Element Matching (DEM) [10] or Digital Correction [11]. Likewise, we can use adaptive digital correction [12], [13] to solve the mismatch problems of the MASH ADCs

V. CONCLUSIONS

In this paper we review the receiver architectures that are suitable for multistandard use and we highlight the need for Programmable and/or Reconfigurable building blocks. Since our focus point is the Analog-to-Digital Converter, we identify the ADC's architectures that are most commonly used in the telecommunication applications, depending on the requirements, and we conclude that the P/R ADCs become a necessity in the modern multistandard terminals, since they can lead to lower power and smaller area. We propose a set of ADC's architectures that can cover the wanted performance space and finally, we identified the P/R options for each one of the architectures.

REFERENCES

- [1] M. Brandolini, P. Rossi, D. Manstretta, and F. Svelto, "Toward Multistandard Mobile Terminal - Fully Integrated Receiver Requirements and Architectures," *IEEE Trans. Microwave Theory Tech.*, vol. 53, no. 3, pp. 1026–1038, Mar. 2005.
- [2] J. C. Rudell et al., "Recent developments in high integration multistandard CMOS transceivers for personal communication systems," in *Proc. IEEE Low Power Electronics and Design, International Symposium on*, Aug. 1998.
- [3] P. Setty, J. Barner, J. Plany, H. Burger, and J. Sonntag, "A 5.75b 350MS/s or 6.75b 150MS/s Reconfigurable Flash ADC for a PRML Read Channel," in *Proc. IEEE ISSCC*, vol. 1, 1998, pp. 148–148.
- [4] van de Grift et al., "An 8bit video ADC Incorporating Folding and Interpolation Techniques," *IEEE J. Solid-State Circuits*, vol. 22, no. 6, pp. 944–953, Dec. 1987.
- [5] K. Bult et al., "An Embedded 240mW 10b 50MS/s CMOS ADC in $1\mu m^2$," *IEEE J. Solid-State Circuits*, vol. 32, no. 12, pp. 1887–1895, Dec. 1997.
- [6] A. Zanicopoulos, P. Harpe, J. A. Hegt, and A. van Roermund, "A Flexible ADC Approach for Mixed-signal SoC Platforms," in *Proc. IEEE ISCAS*, 2005.
- [7] —, "Self-Adjusting Bias Current Technique in Flexible ADCs for Mixed-signal SoC Platforms," in *Proc. IEE Advanced A/D and D/A Conversion techniques and their applications (ADDA'05)*, Limerick, 2005.
- [8] S. R. Northworthy, R. Schreier, and G. C. Temes, *Delta-Sigma Converters: Theory, Design and Simulation*. IEEE Press, 1997.
- [9] Y. Geerts, M. Steyart, and W. Sansen, "Circuit design aspects of multi-bit delta-sigma converters," in *Proc. Advances in Analog Circuit Design (AACD)*, 2002.
- [10] L. R. Carley, "A noise shaping coder topology for 15+ bit converters," *IEEE J. Solid-State Circuits*, vol. 24.
- [11] M. Sarhang-Nejad and G. C. Temes, "A high-resolution multibit $\Delta\Sigma$ adc with digital correction and relaxed amplifier requirements," *IEEE J. Solid-State Circuits*, vol. 28, no. 6, pp. 648–660, June 1993.
- [12] G. Cauwenberghs et al., "Adaptive digital correction of analog errors in MASH ADC's - Part I: Off-line and blind on-line calibration," *IEEE Trans. Circuits Syst. II*, vol. 47, no. 7, pp. 621–628, July 2000.
- [13] P. Kiss et al., "Adaptive digital correction of analog errors in MASH ADC's - Part II: Correction using test-signal injection," *IEEE Trans. Circuits Syst. II*, vol. 47, no. 7, pp. 629–638, July 2000.