# REALIZATION OF QUASI-CONTINUOUS DIGITAL FILTERS

F. Corthay, F. Pellandini

Institut de Microtechnique, Université de Neuchâtel, Rue A.-L. Breguet 2, CH-2000 NEUCHATEL, Switzerland

## RESUME

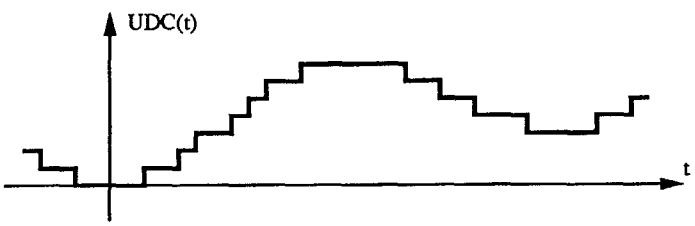
Les Filtres Numériques Quasi-Continus (FNQCs) sont des systèmes numériques, dont la structure peut être directement dérivée de celle de filtres analogiques. Ils se caractérisent par un traitement du signal réalisé de manière quasi-continue dans le temps à l'aide de signaux modulés en fréquence d'impulsions. Les opérateurs des FNQCs sont des compteurs-décompteurs et des multiplieurs à taux programmable. Disposant de ces deux types d'opérateurs, il est possible de synthétiser des filtres à boucles imbriquées similaires à des filtres RC actifs ou à des filtres à capacités commutées. L'analyse des effets dus à la quantification des variables d'état peut se faire à l'aide d'un filtre numérique classique, similaire au FNQC, par des programmes usuels. Cette étude permet de choisir le nombre de bits des compteurs-décompteurs et des multiplieurs à taux programmable et d'assembler, sur un circuit intégré VLSI, les tranches élémentaires nécessaires à la réalisation du FNQC.

## SUMMARY

Quasi-Continuous Digital Filters (QCDFs) are digital systems the structure of which can be derived from analog filters. They are characterized by a quasi-continuous representation of signals, due to the pulse frequency modulation of the variables. Up-down counters and rate multipliers are the basic operators of this class of filters. These operators allow the synthesis of leap-frog filters, similar to RC active filters or switched capacitor filters. The round off noise resulting from the state variable quantization can be analyzed by means of usual software, using a classical digital filter structurally similar to the QCDF. This analysis determines the required number of bits for the up-down counters and the rate multipliers. The actual operators are realized on a VLSI integrated circuit by the mere abutment of the required number of bit slices and assembled to build up the QCDF.

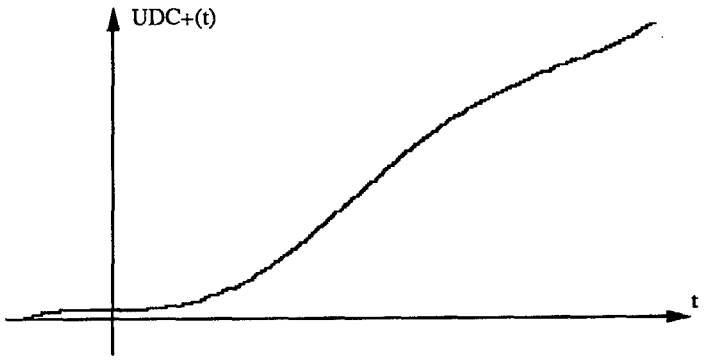
## I. Introduction

Quasi-Continuous Digital Filters (QCDFs) are characterized by a structure using simultaneously two types of binary signal representation: bit-parrallel numbers and Pulse Frequency Modulation (PFM) coded variables. Up-Down Counters (UDCs) transform PFM signals to bit-parallel signals, performing an integration by counting the incoming pulses. Rate Multipliers (RMs) weight their bit-parallel input and convert it to a PFM signal.




a) Bit-parallel output signal of an UDC


b) RM pulse frequency modulation of the signal given in fig. 1a

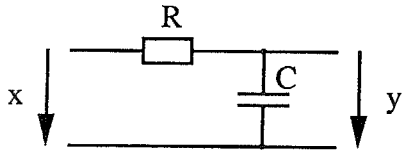



c) UDC integration of the signal given in fig. 1b

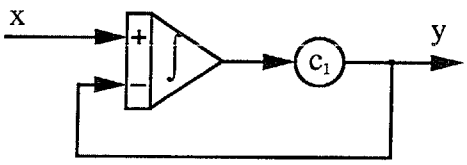
Figure 1 Characteristic signals of a QCDF.

In an ideal QCDF, the PFM pulses can occur at any time, causing the counters to vary asynchronously. The name given to this class of filters has been derived from this basic property. QCDFs have been briefly presented at two conferences [Far84] [Far85] and more extensively in [Far86].

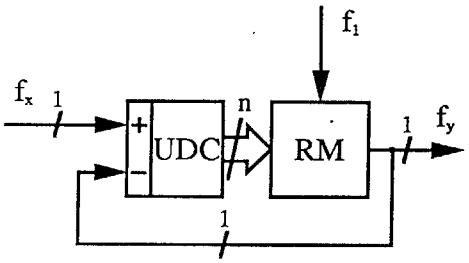
Figure 2 shows the QCDF realization of a first order lowpass filter and compares it to an analog computer implementation of the signal flowgraph of a RC filter. The figure shows the UDC acting as an integrator and the RM setting a filter coefficient.




a) Analog RC filter




b) Signal flowgraph realized by an analog computer




c) Quasi-Continuous Digital Filter (QCDF)

Figure 2 First order lowpass filter.

Following the signal path within the QCDF, it can be noticed that the QCDF input signal is PFM modulated. This signal coding is particularly interesting for applications where the input signal is directly available in this form, as it is the case for some sensors or for signal transmission in a noisy environment, since there is no need for an A/D converter. For analog input signals, a voltage to frequency converter replaces the A/D conversion.

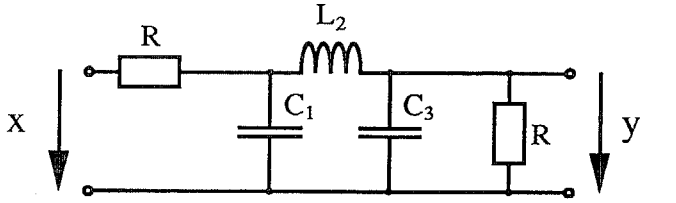
The state variables are also coded using PFM modulation. This reduces wiring costs and operator complexity, with the drawback that the signal bandwidth is low compared to the mean pulse frequency, the ratio being typically $2^n$, where n is the number of bits of the counters. For a given mean pulse frequency, this fact leads clearly to a tradeoff between signal bandwidth and quantization noise. QCDFs are best suited for the realization of selective all-pole lowpass filters. In this case, the filter synthesis, analysis and VLSI implementation are really simple and straightforward.

Finally, the output can be got obtained in a bit-parallel way, at the UDC output, or in a PFM form, at the output of the RM.

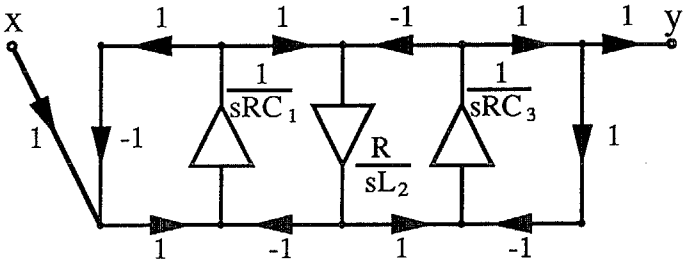
This paper describes the complete realization of all-pole QCDFs, by the example of a third order Butterworth lowpass, starting from the leap-frog synthesis (part II), followed by the analysis of the obtained filter (part III) and ending with the design of a layout for VLSI integration (part IV).

## II. QCDF synthesis in a leap-frog form

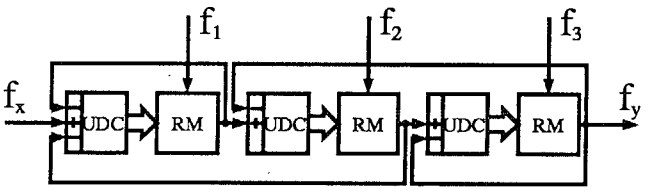
The design of a QCDF is based on an analog ladder reference filter, terminated by equal resistances, in order to inherit its low sensitivity to coefficient values. As for RC active filters [Gir70] or switched capacitor filters [Bah82], an analog prototype is designed to meet the desired filter response. From this lumped element network, we can extract a signal flowgraph which is exclusively made out of integrators and multipliers in the case of an all-pole (Butterworth or Chebyshev) realisation. These operators are replaced by UDCs and RMs in order to derive the corresponding QCDF:




a) Analog all-pole ladder filter




b) Signal flowgraph




c) Leap-frog QCDF

Figure 3 Third order lowpass filter synthesis.

Referring to figure 3, we notice that the filter coefficients are set by corresponding pulse frequencies ($f_1$, $f_2$, $f_3$). They can either be derived from the system clock or generated externally. In the latter case, it can be pointed out that the coefficients are entered via single wire connections. The frequency response of the QCDF is given by the coefficients and remains independent of the system clock frequency; this one has only to be high enough to process every input and coefficient pulse.

For a third order Butterworth lowpass filter with a normalized cutoff frequency of $f_c = 1/2\pi$, the value of the lumped elements ladder are given in table 1 according to [Tem77] while the QCDF coefficients are specified in table 2, where n is the number of bits of the counters.

| | |
|---|---|
| $C_1 = 1$ [F] | $f_1 = 1 \cdot 2^n \cdot f_c$ |
| $L_2 = 2$ [H] | $f_2 = 1/2 \cdot 2^n \cdot f_c$ |
| $C_3 = 1$ [F] | $f_3 = 1 \cdot 2^n \cdot f_c$ |

Table 1 Table 2

The preceding expressions show that the QCDFs are best suited for selective lowpass filtering. As we can see, the leap-frog synthesis of all-pole QCDFs is simple and straightforward; systems with transmission zeroes need more overhead and optimal solutions are presently studied.

## III. Analysis of the QCDF

Compared to the analog prototype, the QCDF suffers from three non-idealities, namely:

- frequeny warping due to sampling,
- quantisation noise

and

- operation of the rate multipliers.

In an ideal QCDF, the information is processed asynchronously throughout the filter. For a practical realization, the pulses of the state variables are synchronized to the system clock, and the outputs of the up-down counters are thus updated at this rate. Sampling is always associated with a z-transform mapping of the original analog to the sampled device transfer function. In our case, the sampling frequency is very high compared to the signal bandwidth so that the effects of the frequency warping can be merely ignored.

The QCDF state variables, which can be found at the output of the counters, are evidently quantized. The analysis of quantization noise is similar to what is done in digital filter theory. Rather than rewriting existing filter analysis software for our purpose, we can derive a classical digital filter equivalent to the QCDF which is then analyzed using existing tools [Cla84]. For this purpose, it has been shown that the UDCs have to be replaced by an accumulator followed by a scaling multiplier, and the RMs by a digital multiplier. As we want to analyze only the effects due to the quantization, we choose the filter sampling frequency arbitrarily high, typically of the same order of magnitude as the QCDF system clock. Figure 4 shows the digital filter equivalent to the first order lowpass of figure 2:

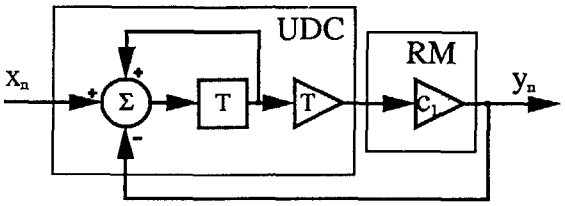



Figure 4 Digital filter, equivalent to the QCDF of figure 2.

Since the filter arithmetic is of unsigned type and overflow is treated by saturation to ensure stablility, the QCDF acts as a pseudopassive network, in the same way as wave digital filters [Fet86].

More difficult to analyze is the rate multiplier. It performs the multiplication of a pulse frequency modulated coefficient with a bit-parallel state variable in a manner similar to the techniques used for image dithering. The corresponding noise was taken into account in the filter analysis, but the time simulation and limit cycle detection had to be done with a dedicaced simulator.

The analysis of the classical filter equivalent to the third order Butterworth showed that the filter needs internally 11 bits to ensure the correctness of the 8 MSBs at the output. The step response simulation allows us to be even more optimistic about the quantization noise effects (figure 5).

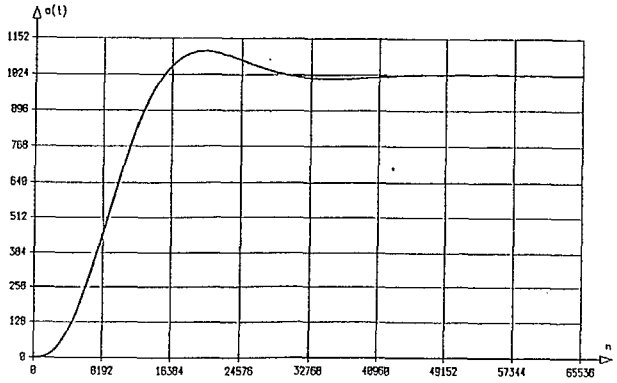
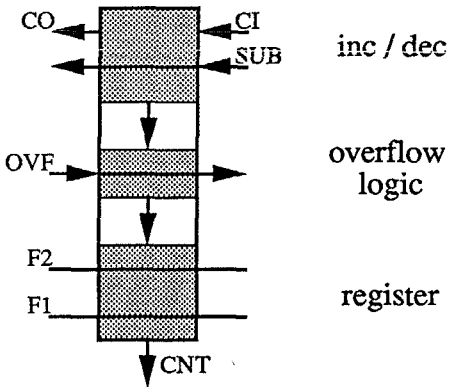


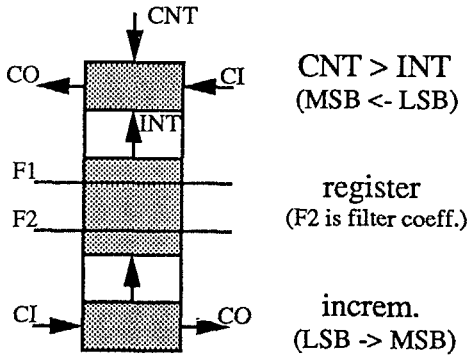

Figure 5 Step response of a third order Butterworth QCDF.

## IV. Integrated circuit implementation

The basic QCDFs building blocks have been designed for a CMOS 2µm technology (VLSI Tech. Inc. cmn20a). They consist of Up-Down Counter (UDC) and Rate Multiplier (RM) bit slices using static carry ripple-through logic. The UDC bit slice is made out of an incrementer/decrementer, a saturation overflow logic and a register driven by the two system clock phases F1 and F2. The RM consists of a bit reversed (having its LSB facing the RM input MSB) free running counter driven by the filter coefficient and a comparator which sends out a pulse when the RM input (CNT) is greater than the value read from the bit reversed internal counter (INT).




Up-down counter bit slice




rate multiplier bit slice

Figure 6 Up-down counter and rate multiplier bit slice.

One UDC-RM bit slice counts 112 transistors. The placement of their connectors allows to build up an n bit operator by the mere abutment of n bit slices. The UDC overflow condition (OVF) is the carry out (CO) of the MSB slice. The achieved up-down counters and rate multipliers can also be abutted by pairs.

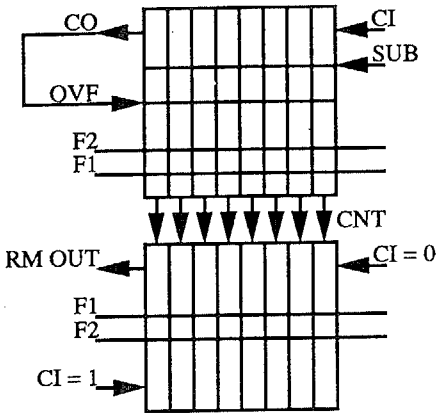



Figure 7 8-bit up-down counter and rate multiplier.

Finally, an intermediate slice connects the operator pairs and distributes the pulse frequency modulated state variables to the neighbouring counters in the leap-frog manner. Figure 8 presents the floorplan of the third order 11 bit QCDF of figure 3:

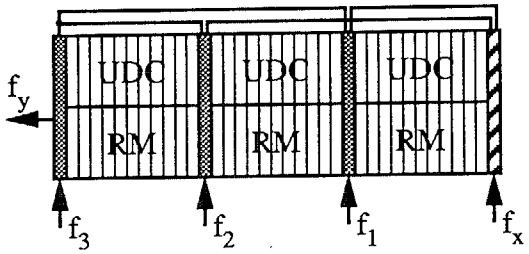



Figure 8 Floorplan of a third order 11-bit lowpass QCDF.

## V. Achieved results

The density of the basic slices is greater than 4'000 MOS/mm$^2$. The third order 11-bit lowpass filter has been realized as shown in figure 8. It contains 3879 transistors and uses 0.6 x 1.85 mm$^2$ = 1.11 mm$^2$, which gives us a global density of 3'500 MOS/mm$^2$. The circuit has been integrated and successfully tested up to 10 MHz clock frequency.

The filter layout, corresponding to the floorplan of figure 8, is displayed in figure 9.

With a clock rate of 10 MHz, an 11 bit QCDF has a maximal cutoff frequency of 5 kHz. The observed step response matches the simulation results of figure 5.

## VI. Conclusion

QCDFs show to be of interest for selective lowpass filters, where the input is given as a PFM signal. The synthesis and analysis steps are performed by well known analog and digital signal processing methods. Due to the filter regularity, the VLSI implementation is easy and the achieved transistor density is high.

## VII. Acknowlegements

This work was supported by the *Commission pour l'Encouragement de la Recherche Scientifique* (CERS n° 1506), Bern, Switzerland and by *ASULAB S. A.* , Neuchâtel, Switzerland.

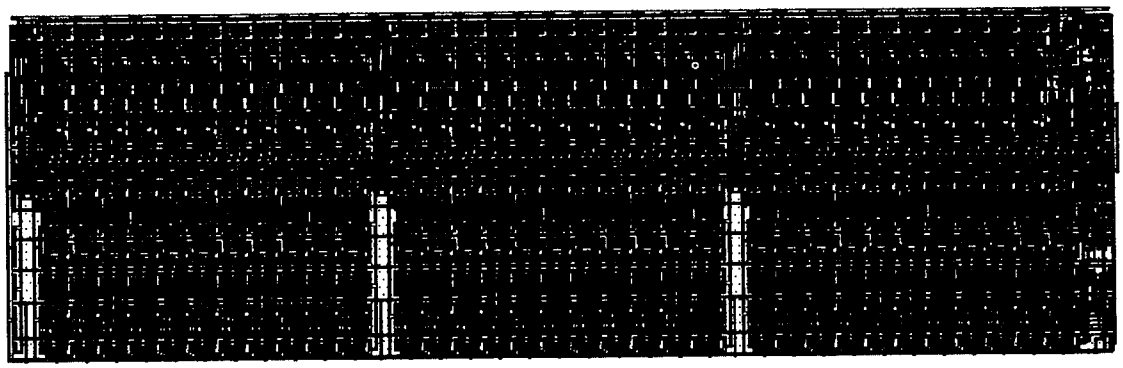


Figure 9 Layout of a third order lowpass QCDF.

## VIII. References

[Gir70] F. E. J. Girling, F. E. Good, "*Active filters 12 - The leapfrog or active-ladder synthesis*", Wireless World, Vol. 76, pp 341-345, July 1970.

F. E. J. Girling, F. E. Good, "*Active filters 13 - Applications of the active-ladder synthesis*", Wireless World, Vol. 76, pp 445-450, Sept. 1970.

[Tem77] G. C. Temes, J. W. LaPatra, "*Circuit synthesis and design*", International Student Edition, McGraw-Hill, Tokyo, Japan, 1977.

[Bah82] H. Baher, S. O. Scalan, "*Stability and exact synthesis of low-pass switched-capacitor filters*", IEEE Trans. on Circuits and Systems, Vol. CAS-29, No 7, pp 488-492, July 1982.

[Cla84] L. Claesen, H. De Man, J. Vandewalle, "*DIGEST: a digital filter evaluation and simulation tool for MOS VLSI filter implementation*", IEEE J. of Solid-State Circuits, Vol. SC-19, No. 3, June 1984.

[Far84] P.-A. Farine, F. Pellandini, "*On the behaviour of quasi-continuous digital filters*", Int. Conf. on Digital Signal Processing, pp 27-31, Florence, Italy, September 1984.

[Far85] P.-A. Farine, F. Pellandini, "*Technique de filtrage numérique quasi-continu. Avantages et limitations*", Dixième colloque sur le traitement du signal et ses applications, pp 449-454, GRETSI, Nice, France, 1985.

[Far86] P.-A. Farine, "*Filtrage numérique caractérisé par un traitement quasi-continu des signaux*", Thèse de doctorat, Université de Neuchâtel, Suisse, 1986.

[Fet86] A. Fettweis, "*Wave digital filters: theory and practice*", Proc. IEEE, Vol. 74, No. 2, pp 270-327, Feb. 1986.