A Full Featured Ultrasound NDE System in a Standard FPGA

C. FRITSCH, J. CAMACHO, A. IBAÑEZ, J. BRIZUELA,
Instituto de Automática Industrial, Madrid, Spain
R. GIACCHETTA, R. GONZÁLEZ, DASEL, S.L., Madrid, Spain

Abstract. This work presents a new NDE system based on a pipelined architecture built into a low-cost FPGA. Flaw detectability is improved by two means: first, by migrating some traditionally analog functions, like filtering and envelope extraction to the digital domain. Second, by including some Digital Signal Processing functions in the processing chain, mainly an effective EMI noise filter and a data compression algorithm which keeps all the flaw-relevant information. These algorithms coexist with conventional gating, TGC and other advanced ultrasound processing functions in a compact, USB driven instrument .

Introduction.

Ultrasonic Non Destructive Evaluation has become a standard practice for the integrity assessment of critical parts in many industrial fields: power generation, aeronautics, railways, civil engineering, food industries, etc. Inspections are carried out manually by expert operators or automatically, by means of scanners and intelligent software.

NDE instruments have been developed trying to simplify and improve their flaw detection capabilities. Currently, the increasing availability of very large scale integrated devices, particularly field programmable gate arrays (FPGAs) and digital signal processors (DSP), open the technological path to NDE systems with increased functionality and performance.

In particular, by migrating some functions from the analog to the digital domain, improved accuracy, repeatability and performance are obtained. Currently this can be achieved with simultaneous cost and power consumption savings.

Furthermore, along the years, many digital signal-processing methods have been devised, specifically directed towards the improvement of the information content of every A-scan. Their hardware implementation allows a higher throughput and an important data reduction by preserving only the relevant information, especially in automated NDE.

In this sense, gates provide information of peak amplitude and position within user-defined ranges, allowing for C and D-scan images. However, if only this information is kept, the impulsive EMI noise frequently found in industrial environments, yields false indications which hidden the true flaw echoes and reduce the system dynamic range. Conventional filtering is insufficient due to the broadband nature of the noise. To limit these problems, averaging or full A-scan recording to build B-scan images, less sensitive to impulsive noise, is commonly used. Both methods have their respective drawbacks, in the form of increased acquisition time or storage requirements.

This work presents a different approach, which is based on a pipelined architecture with multiple digital processing functions. Since all the processing functions are carried out in parallel, results are provided in real-time.

The processing chain includes an effective non-linear EMI-cancellation filter, followed by optimised RF filtering, accurate envelope extraction and a data compression algorithm that keeps all the flaw-relevant information. This combination provides a superior performance level with regard to speed, accuracy, range resolution, signal-to-noise ratio and storage requirements.

These processing functions have been integrated into a single, low-cost FPGA, together with other functions found in high-performance flaw detectors (multiple gates and alarms, TGC, pulser control, triggering, etc.). The paper focuses on the overall architecture, the algorithms implemented and their application to the field of ultrasonic NDE.

1. Background and architecture.

Space and energy savings are important for many applications, ranging from portable instruments to continuous monitoring with structural-embedded systems. Integration of the system into a single chip helps to achieve these goals, but it is required that this is not done at the expenses of performance. Modular design and pipelining are good choices to this purpose, as it was demonstrated by a former system in which different signal processing functions were performed in separate modules [1].

The availability of high-density programmable logic allows the integration of multiple processing modules into a single FPGA chip, while pipelining helps to carry out multiple DSP functions in parallel.

Figure 1 shows the described system, built in a 160x100 mm printed circuit board. Analog functions are reduced to a programmable high-voltage power supply (50-400V), and a pulser-amplifier circuit with active protection circuitry. Firing pulses are square-wave (30-1600 ns wide, 6.25 ns resolution) and can operate in burst mode.

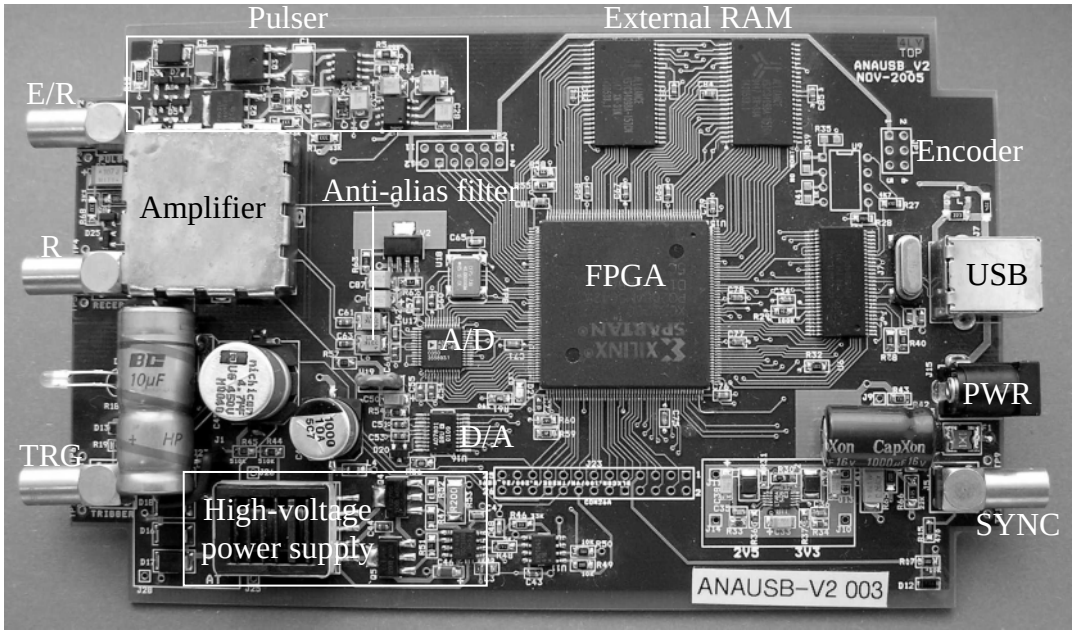

Fig. 1.- Picture of the system.

In the receiver path, a programmable attenuator and an ultra-low noise (0.74 $nV/Hz^{1/2}$) voltage-gain controlled amplifier provide differentially amplified signals to the analog-to-digital converter, operating at 80 MS/s. Only an anti-alias filter is used in the

analog section, which limits signal bandwidth to 30 MHz (-3 dB). A Digital-to-analog converter provides the Time-Gain-Correction voltage to the amplifier.

A single low-cost FPGA (XC2S200, Xilinx Inc.) receives data samples from the A/D, drives the TGC D/A and performs the control of the pulser and other circuits. In particular, it provides I/O to interfacing encoders, motors, alarm monitoring, external triggering, synchronisation and general purpose lines.

Inside the FPGA, a pipelined architecture is built to perform the different Digital Signal Processing functions that achieve the high performance of the system. Operational parameters are stored in the FPGA until new ones are programmed. Results are arranged in frames and sent to a external high-speed memory buffer (1 MB). A data frame typically provides the processed A-scan, origin, peak amplitude and position of 3 gates and encoder data together with synchronising and status information.

A chip interfaces the FPGA with an Universal Serial Bus v.2.0 (480 Mbit/s). Power consumption remains below 2.5W in normal operation, drawn from a single +5V rail. A host computer uses the USB to set up the required parameters and to recover information in data blocks. A block of data is composed of some integer number of frames. The application software extracts the different fields of information to the corresponding tasks.

2. EMI noise cancellation.

NDE equipment is frequently used in industrial and noisy environments. By far, impulsive EMI noise is more harmful than white or coloured noise since the filter response is indistinguishable from flaw indications. Operators ignore EMI spikes because they appear randomly in the A-scan screen.

However, in automatic inspections and, specially, when performing C or D scans, which are based on the signal amplitude or peak position, impulsive EMI noise can completely hidden the weakest signals, reducing the effective dynamic range of the system and the possibility of detecting flaws below the noise level.

An effective EMI noise cancellation filter is included at the beginning of the processing chain. It is based on the principle that, in a small set of consecutive acquisitions taken a short time interval apart, noise spikes appear at random positions while flaw indications remain at the same location.

To this purpose, an order statistics non-linear filtering process [2] applied to 2 or 3 consecutive A-scans is followed. The acquisitions are taken at intervals of 256 μs (this represents a displacement of 0.25 mm at a scanning speed of 1000 mm/s).

Following this procedure, for a set of M consecutive A-scans of length L samples, with x_{ik} being the sample k of A-scan i, every result sample r_k is obtained as:

$$r_k = \Psi_{i=1}^{M}(x_{ik}) \qquad 1 \le k \le L \tag{1}$$

where $\Psi(\cdot)$ represents the non-linear filtering function applied to the M A-scans. Here, the *minimabs* function that selects the nearest to zero value has been chosen:

$$\begin{aligned} \Psi(a,b) &= a \quad for\ |a| \le |b| \\ \Psi(a,b) &= b \quad for\ |a| > |b| \end{aligned} \tag{2}$$

where $|\cdot|$ represents the absolute value. Then,

$$\begin{aligned} r_{1k} &= x_{1k} \quad i = 1, \quad 1 \le k \le L \\ r_{ik} &= \Psi(x_{ik}, r_{i-1,k}) \quad i > 1, \quad 1 \le k \le L \end{aligned} \tag{3}$$

where Ψ represents the minimabs function, which is just a binary operator.

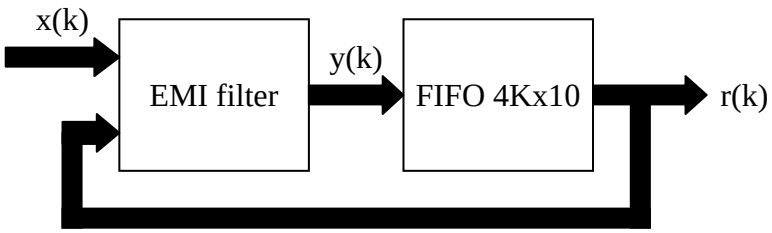

Figure 2.- EMI filtering.

The architecture for the EMI filter is shown in Fig. 2. The process starts with the EMI-filter disabled to store a single A-scan in the FIFO, which performs the first part of Eq. (3). Then, the subsequent A-scans are filtered with partial results stored in the FIFO: following the second part of Eq. (3), the *minimabs* function is applied to every A-scan sample (x_{ik}) and to the corresponding partial result ($r_{i-1,k}$). Note that the FIFO always keeps a constant number L of data once it has been filled with the first A-scan.

The EMI filtering hardware is independent of the number M of A-scans processed but, in practice, there is little gain if M exceeds 3 or 4. Moreover, since this is a non-linear filtering function, there is a risk of destroying useful information if this figure is made too large. The EMI noise cancellation is especially effective for A, B, C and D scans as shown in Fig. 3. Here, the B-scan of an aluminium test-piece with several flat bottom holes is obtained in a highly noisy environment. The effect of the EMI filter is apparent.

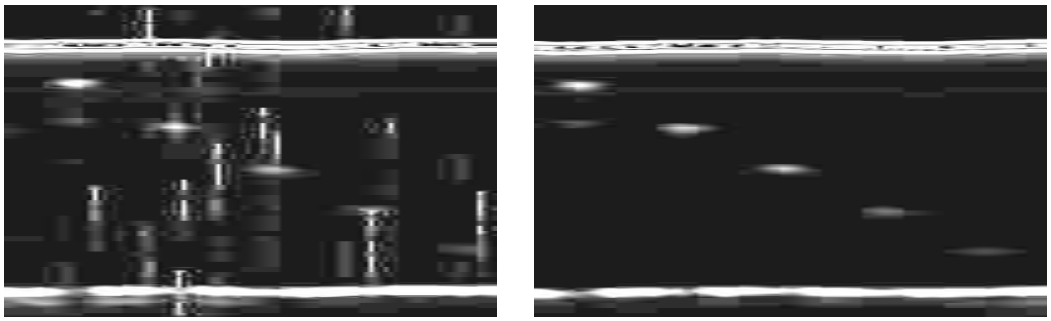

Fig. 3.- B-scan of an aluminium test-piece with several FBH. Left: EMI filter disabled, Right: EMI filter enabled. Transducer 10 MHz.

3. Linear filtering.

Once the impulsive EMI noise has been cancelled, linear filtering is a very important function to reduce unwanted noise (white or coloured). It is usually carried out in the analog domain, with first, second or (seldom) third order filters. These circuits use several capacitors, inductors and selection logic, which require a large amount of real estate and components of tight tolerances. Furthermore, only a small set of fixed low-pass and high-pass frequencies can be chosen by the user. Analog filters show a non-flat frequency response in the band-pass, large transition bands and non-linear phase.

On the other hand, filtering in the digital domain nearly removes these limitations. FIR filters are linear in phase and provide arbitrarily defined cut-off frequencies within the Nyquist band. With a high order enough, response in the passband can be done flat and

transition bands narrow, with a high attenuation in the band-stop. Furthermore, linear filtering is applied *after* the impulsive noise has been removed.

The cost is the involved hardware. An order-N FIR filtering performs the convolution between the incoming data sequence *x(k)* and a set of N+1 coefficients *c(i)*:

$$y(k) = \sum_{i=0}^{N} x(k-i)c(i) \qquad (4)$$

The challenge is that *N* multiplications and additions have to be carried out for every output sample. However high processing rates can be achieved, using distributed arithmetic to perform these specific operations in FPGAs [3].

In this work, a programmable 63-coefficient FIR filter producing a filtered result every 50 ns has been implemented. Coefficients and data are 10-bits wide; full precision results are 25-bits wide. Figure 4 shows the filter frequency response for two filters: narrowband (4 to 6 MHz) and wideband (2 to 8 MHz) for a sampling frequency f_S = 80 MHz. The ratio of the pass-band to the stop-band attenuation is about 50 dB or more.

The filter output is conveniently scaled to keep the response constant independently of the filter bandwidth. This is achieved by computing:

$$z(k) = \frac{G}{1024} y(k) \qquad (5)$$

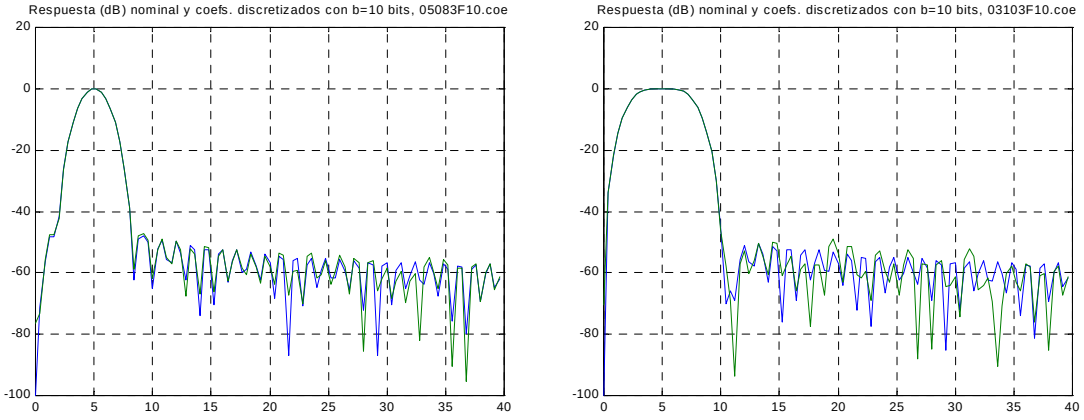

Fig. 4.- Filter response. Left: narrowband (4 to 6 MHz), right: wideband (2 to 8 MHz)

Figure 5 shows the filter arrangement with *G* being the scaling factor, between 0 and 1023. Overflow conditions are managed by saturation so that the output values are limited to fit a symmetrical output range around zero.

The RF filter can perform any arbitrary filtering function, simply by reprogramming the coefficient set. This way, besides band-pass filtering, deconvolution (Wiener filtering, for instance), cross-correlation (pulse stretching) and other well documented DSP functions useful for NDE applications can be implemented.

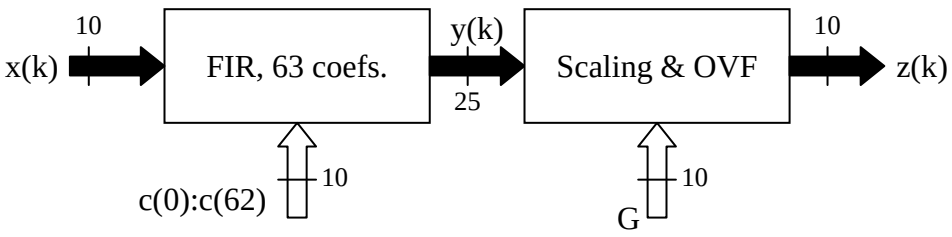

Fig. 5.- Structure of the FIR filter with scaling and overflow handling.

4. Envelope extraction.

Most NDE flaw detectors and imaging systems are based on the envelope of the signal rather than in the RF itself. The envelope is commonly used to assess the flaw size with regard to indications of standard flat bottom holes (FBH). The linearity and conformance to theoretical envelope of the detector are, thus, critical in quantitative NDE [4].

The envelope is usually detected in the analog domain by means of: a) rectification and filtering or b) synchronous AM detection. However, filtering the high-frequency components while keeping undistorted the baseband information is not optimal, and most flaw detectors provide a small set of *smoothing* filters to set a trade-off among resolution and high-frequency rejection.

In the digital domain there are several techniques which have been used for this purpose. The universally recognised as more accurate involves computing the absolute value of the Hilbert transform [5], but this is almost restricted to software approaches (a FFT and an IFFT or a specialised FIR Hilbert filter are required, besides a modulus extraction operation).

For real-time applications, some other approaches have been suggested in the past [6], but they require a considerable amount of memory for look-up tables which do not fit in the storage capabilities of current FPGAs.

The approach followed here applies full rectification and filtering, but with several advantages derived from the digital implementation. For signals of angular frequency ω, the full-rectified wave is:

$$y(t) = A(t)\left|\cos(\omega t)\right| \tag{6}$$

The series development yields:

$$y(t) \approx A(t)\left(\frac{2}{\pi} + \frac{4}{3\pi}\cos 2\omega t - \frac{4}{15\pi}\cos 4\omega t + \ldots\right) \tag{7}$$

Considering the frequency response of a moving average filter of order P:

$$\left|H(f)\right| = \left|\frac{sin(\pi f P)}{P sin(\pi f)}\right| \tag{8}$$

which presents zeros at frequencies $f \cdot P = k$, with $k = \pm 1, \pm 2, \pm 3, \ldots$ Then, to null out frequencies $2 \cdot f_R$, $4 \cdot f_R$, $6 \cdot f_R$, ... corresponding to the even harmonics of the received signal of fundamental frequency f_R, it is required that:

$$P = k\frac{f_S}{2 f_R} \quad k = 1, 2, \ldots \tag{9}$$

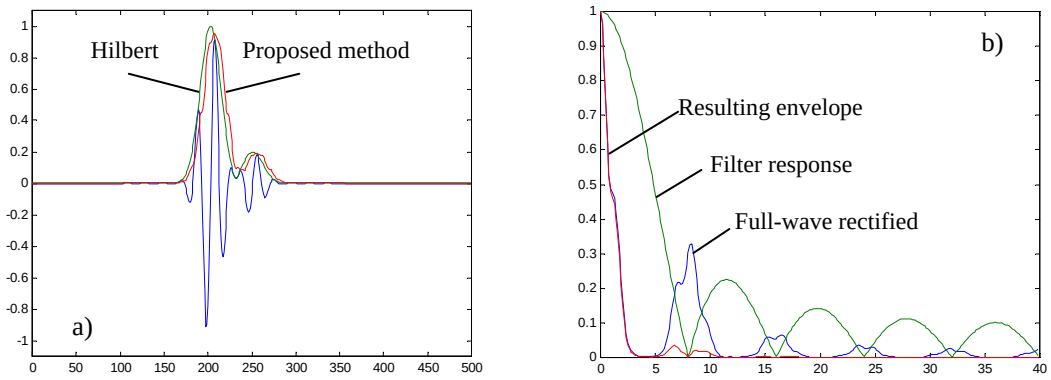

Fig. 6.- Envelope extraction. a) Absolute value of the Hilbert transform vs. the proposed method; b) involved spectra.

The minimum value of P is obtained for k = 1. Figure 6a shows two overlapping echo signals from a 4 MHz transducer, sampled at 80 MHz, together with the obtained envelope by the absolute value of the Hilbert transform and by the proposed method with P=80/2·4 = 10. Differences are negligible.

In Fig. 6b the frequency spectrum of the full-wave rectified signal, the frequency response of the filter and the spectrum of the resulting waveform are shown. Note that the spectra around the even harmonics in the rectified signal coincide with zeros in the frequency response of the filter. As a result, these harmonics become highly attenuated, while the envelope spectrum is kept. The moving average filter computes:

$$y(k) = \frac{1}{P}\sum_{i=0}^{P-1} x(k-i) \qquad (10)$$

This function can be implemented with a P-coefficient FIR filter, with all the coefficients set to 1/P, but a recursive implementation is more efficient. From (10):

$$Y(k) = Py(k) = Y(k-1) + x(k) - x(k-P) \qquad (11)$$

which is computed in real-time with the circuit shown in Fig. 7. First, the absolute value of the incoming sequence *xa(k)* is obtained and, then, passed thorough the filter, implemented following Eq. (11) as it can be easily verified. The output *Y(k-1)* is scaled by a number that is a function of the P value.

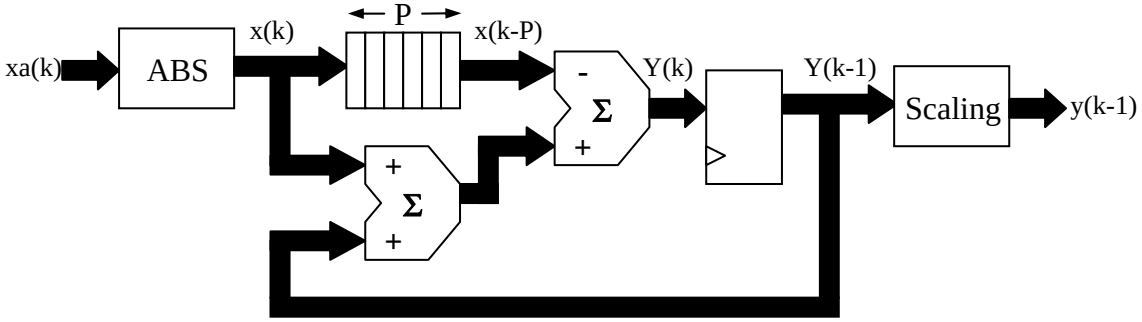

Fig. 7.- Structure for the envelope extraction.

5. Data compression.

Automated NDE of large parts produce a huge volume of information. Traditionally, gating has been used to reduce the information to just peak amplitude and position on every A-scan. Although C- and D-scans can be produced from this data set, no further analysis is possible. B-scan imaging is preferred, but this would require recording of all A-scan data.

Decimation is frequently used to decrease the data volume. However, if A-scans are decimated more than indicated by the Nyquist criterion, flaw indications can be lost, so that the data reduction gets rather limited.

In this work we have implemented a different data-compression algorithm, which is simple and quite effective. Data reduction rates are programmable and can be very high (above 100:1), without loss of peak amplitude and relative position in the resulting A-scan.

To this purpose, given a compression factor F (i.e., obtaining a data reduction of F:1), every A-scan of length L samples is decomposed in a set of G contiguous sequences {S_1, S_2, ..., S_G} every one containing 2F samples:

$$G = \left|\frac{L}{2F}\right|_{\uparrow} \qquad (12)$$

where $\lceil \cdot \rceil_{\uparrow}$ is the rounding function to infinity; the last sequence might have less than $2F$ samples. The sequence i is:

$$S_i = \{x_A, x_{A+1}, ..., x_{A+2F-1}\} \qquad A = 2F(i-1)+1 \qquad i = 1,2,...G \tag{13}$$

From this sequence the following values are computed:

$$\begin{aligned} p_i &= min\{S_i\} \qquad q_i = max\{S_i\} \\ ip &= index(p_i) \qquad iq = index(q_i) \end{aligned} \tag{14}$$

where $index(\cdot)$ is the index within the sequence. Then, the sequence S_i (length $2F$) is transformed in the reduced sequence R_i (lenght 2) by means:

$$\begin{aligned} &If \quad ip \le iq, \quad R_i = \{p_i, q_i\} \\ &otherwise \quad R_i = \{q_i, p_i\} \end{aligned} \tag{15}$$

This way a compression $2F$:2 or F:1 is achieved. The important feature is that the maximum and minimum amplitude within every sequence and their relative positions are kept.

Figure 8 shows an example, where a decimation by a factor F=16 of an A-scan of length L= 4000 samples, produces loss of data (those marked 1, 2 and 3 in a), while using the described algorithm (in b), all the indications are kept with their exact amplitude and relative position. In both cases, the resulting A-scan has 250 samples.

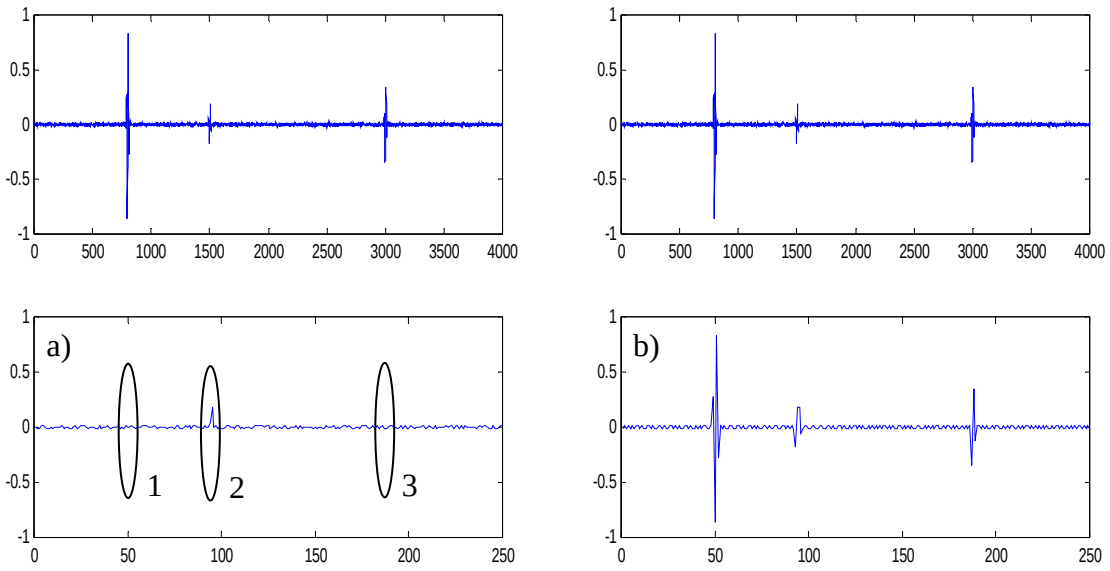

Fig. 8. A-scan compression 16:1: a) By decimation, b) By the proposed method.

6. Overall architecture.

Figure 9 summarises the overall architecture, including the already considered functions and other that will be briefly presented.

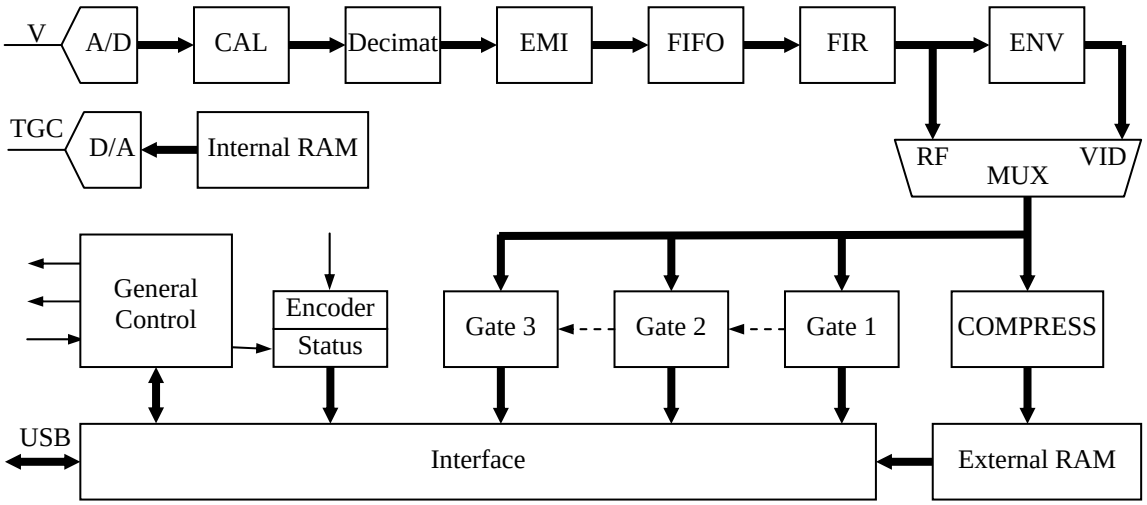

Fig. 9.- Overall architecture and data path.

First, the offset is removed from acquired samples in a calibration block (CAL), which also detects the signal level for "echo-start" acquisition. Then, a decimation process allows to optionally increasing the acquired depth. Following it, the EMI filter and associated FIFO removes impulsive noise.

Linear filtering is carried out by the programmable 63-tap filter, adjusting the bandwidth to that of the transducer. This way, the signal-to-noise ratio is improved while keeping the transducer axial resolution. Envelope extraction (ENV) follows and a multiplexer selects among RF or video mode.

There are three gates which can be defined independent or linked, positive or negative and with programmable alarm thresholds. Gating is performed in parallel with the compression process. This way, detected peaks in the gates have the resolution of the sampling period (12.5 ns without decimation, less than 0.02 mm in steel for shear waves), independently of the compression factor.

The interface block collects information from several parts of the system building a *frame* for every acquisition and processing cycle. The frames may or may not include the processed A-scan following the user needs, but always contains information of the start, peak amplitude and position on every gate, encoder count, status word, and synchronisation data. Frames are sent thorough the USB port to the host computer for further processing, display and storage.

Conclusions.

Advanced ultrasound NDE equipment can be now developed using the features and high performance of available high-density programmable logic, mainly FPGAs. This allows migrating functions traditionally performed by analog circuitry, like filtering and video

detection, to the digital domain, with several advantages in cost, space and power consumption.

On the other hand, new and known algorithms to improve the information content of the A-scans can be also integrated in the same device, fully achieving the concept of "system on a chip". The availability of denser devices will allow the inclusion of a higher degree of functionality and performance.

Pipelining is a good choice for these applications, which provide a modular way to incorporate new functions. Furthermore, re-configurable devices allow an easy upgrade of the system and its adaptation to special applications.

Acknowledgements.

This work has been supported by project DPI-2004-06470 of the Spanish Ministry of Education and Science and by a research contract between CSIC and DASEL, S.L. We also want to acknowledge the work of J.C. Liébana and D. Videgaín in setting-up all the prototypes. Finally, we acknowledge the patent application carried out by the OTT of CSIC about this subject (Pat. nr. 200601187, 9 May 2006).

References.

[1] C. Fritsch et al. : "A Pipelined Architecture for High Speed Automated NDE", Proc. IEEE Ultrasonics Symposium, Seattle (WA), pp. 833-836, Nov. 1995

[2] I.Pitas, A.N Venetsanopoulos: “Non linear digital filters”, Kluwer Academic Publishers, Massachusetts, 1995.

[3] L. Mintzer: "FIR Filters with Field-Programmable Gate Arrays", The Journal of VLSI Signal Processing, 6, 2, Springer Verlag, pp. 119-127, Aug. 1993.

[4] A. S. Birks, R. E. Green: "Nondestructive Testing Handbook", 2nd. Edition, vol. 7, ASNT, 1991.

[5] A.V. Oppenheim, R. W. Schafer: "Digital Signal Processing", Prentice-Hall, 1975.

[6] C. Fritsch, A. Ibañez, M. Parrilla: "A Digital Envelope Detection Filter for Real-Time Operation", IEEE Trans. on Instrumentation and Measurement, 48, 6, 1287-1293, 1999.