

ONE-BIT PROCESSING FOR REAL-TIME CONTROL

Xiaofeng Wu*, Roger Goodall*

*Department of Electronic & Electrical Engineering, Loughborough University, LE11 3TU, UK.

fax: +44 (0)1509 227014

email: {X.Wu, R.M.Goodall}@lboro.ac.uk

Keywords: $\Delta\Sigma$ modulation control system processing 1-bit signal processing system-on-chip pulse-density-modulation

Abstract

In this paper we present an important new concept for real-time control law implementation based on bit-serial approach. It is shown that control laws can be efficiently implemented by 1-bit processing through the use of $\Delta\Sigma$ modulation. After processing, the bitstream output can be directly applied to analogue interface according to pulse-density-modulation (PDM). This technique is quite promising for system-on-chip (SoC) applications as the system cost can be greatly reduced by bit-serial architectures and eliminating the decoding of a bitstream prior to processing.

1 Introduction

This paper investigates a highly-efficient signal processing algorithm and architecture for real-time control. There has been much work on bit-parallel processing, including one of the author's own work which has resulted in a targeted Control System Processor (CSP) architecture [6], but the approach taken here relates to 1-bit signal processing which is becoming increasingly common in the audio world but has not yet been applied for real-time control. One-bit processing almost certainly involves higher sampling frequencies, but results in significant reduction in computational complexity. The long-term aim of the research is to determine the overall benefits, particularly for implementing higher-order dynamic controllers; this paper describes the development of appropriate algorithms and architectures.

$\Delta\Sigma$ modulation has been rapidly gaining popularity in audio signal processing as an effective method for building high resolution A/D and D/A converters. $\Delta\Sigma$ A/D and D/A converters, illustrated in Figure. 1, can modulate the analogue input into a simple bitstream and demodulate the digital signal into analogue form at a very high frequency. From Fig. 1, the $\Delta\Sigma$ modulated signals are in the format of 1-bit. Thus, it is useful to consider signal processing directly without decimation filter and interpolation filter in an attempt to reduce the circuit complexity and save the valuable silicon area.

As $\Delta\Sigma$ modulation can be implemented in control systems easily, it is desirable to perform signal processing on $\Delta\Sigma$ modulated signals. However, it is essential to choose some suitable structure of digital controllers. Ideally there will be no multi-bit

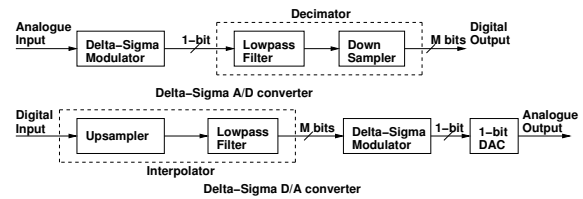


Figure 1: A/D and D/A converters based on $\Delta\Sigma$ modulation

multipliers as multipliers on DSPs and also processor solutions are a dominant factor in i.c. design and performance. In section 2.3, a controller structure is designed based on the modified δ -form. As the controller output after $\Delta\Sigma$ modulation is either 1 or -1, it is obvious that multiplication becomes a simple 'conditional sign switch', i.e. not a multiplier in the normal sense.

Signal conversion with $\Delta\Sigma$ modulation requires a very fast sampling frequency. It is sometimes argued that the sampling frequency cannot be too high otherwise numerical issues arise, but in fact this is not strictly correct [3]. It is usually necessary to sample perhaps 50-100 times the system bandwidth if the phase lags introduced by sampling and computation delays are to remain small, and this intrinsic requirement for fast sampling with real time control (which does not necessarily apply for other forms of signal processing) means that much more modest sampling frequency increases are needed in a relative sense. Therefore, because the signal is necessarily fast sampled, the system characteristics can approach those of high quality analogue processors in terms of phase responses and distortion effects, while retaining the advantages of digital processing techniques.

For 1-bit signal processing, it can be implemented into FPGAs easily with bit-serial architectures despite the bit-parallel processing still shows its advantages over traditional control system processing. The advantage of bit-serial architectures is that all of the bits pass through the same logic, resulting in a huge reduction in the required hardware. Typically, the bit serial approach requires $(1/n)^{th}$ of the hardware required for the equivalent n-bit parallel design. The price of this logic reduction is that the serial hardware takes n clock cycles to execute, while the equivalent parallel structure executes in one. However, the time-hardware product for the serial structures is often smaller than that for equivalent parallel ones because the logic delays between registers are generally significantly smaller [1].

In this paper we describe 1-bit signal processing algorithms and bit-serial architectures, which operate on $\Delta\Sigma$ modulated

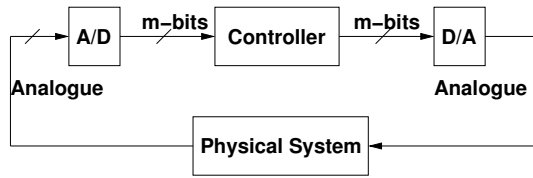


Figure 2: $\Delta\Sigma$ based control system

signals. The remainder is organized as follows. Section 2 introduces the principles of 1-bit signal processing including $\Delta\Sigma$ modulation, δ -operator and controller structures. Section 3 discusses the basic blocks of bit-serial architecture. Section 4 describes an application of digital controller implementation using $\Delta\Sigma$ modulation. Section 5 concludes and discusses future research directions.

2 $\Delta\Sigma$ based control system Design

1-bit signal processing has been explored in audio industry for the last decade. Companies like Sony, Philips, etc. have already applied this technique into digital recording systems. In control, however, it is still unknown as control engineers are used to bit-parallel processing. But bit-parallel processing shows its limits in system-on-chip applications. It is especially true when the ambition is to integrate the whole control system into one FPGA as A/D, D/A converters and multi-bit multipliers consume a lot of silicon area. In this case, 1-bit signal processing seems to be an alternative good solution. The question is how it works?

Fig. 2 shows the scheme of $\Delta\Sigma$ based control system. The A/D converter can be replaced by a simple $\Delta\Sigma$ modulator with the output of 1-bit signal. The controller is modified for 1-bit signal processing. The output of controller is two-level (1/-1) signals, hence a series of pulses. The physical system can be controlled by the density of the pulses, which is so called PDM. So the D/A converter can often be removed in the $\Delta\Sigma$ based control systems, although in some cases simple analogue filtering may be needed.

2.1 $\Delta\Sigma$ Modulation

$\Delta\Sigma$ modulation is an algorithm by which analogue and digital signals are coded in a low resolution and high sampling rate format. The simplest $\Delta\Sigma$ modulator is the first-order implementation in Fig. 3 (a). The quantizer works as a comparator, from which the output is a single bit 1 or -1. The relationship between the input and output can be described as follow: The bitstream out is subdivided into intervals of L bit. The number M of '1'-states and N of '-1'-states contained in an interval are counted. The average output value can be computed as

$$\bar{O} = (M - N)/L. \quad (1)$$

As the input is constrained between -1 and 1, the U_{out} should

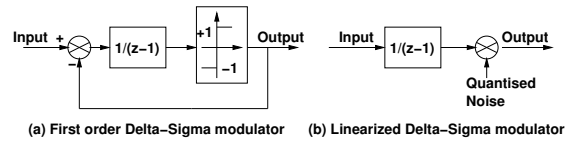


Figure 3: $\Delta\Sigma$ modulator

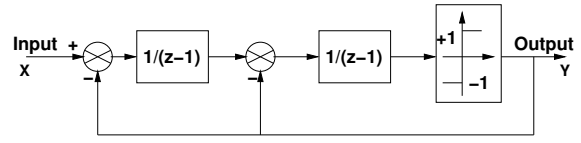


Figure 4: The digital second order $\Delta\Sigma$ modulator

equal to U_{in} when U_{out} is computed as

$$U_{out} = K \cdot out, \quad (2)$$

where K is a scaling value.

The control system containing $\Delta\Sigma$ modulation can be thought of as a non-linear one. With the uniform quantizer in the loop, it is surprisingly difficult to analyze such a control system precisely. Notice the fact that the output bitstream will definitely contain noise due to the quantization. For the purpose of analyzing conveniently, here, we can make a simple assumption that the $\Delta\Sigma$ modulator introduces quantization noise into the loop [7]. Hence, the modulator can be modelled as a linear approximation shown in Fig. 3 (b).

Although many $\Delta\Sigma$ modulators are possible, in this paper the control system will be based on a second-order modulator shown in Fig. 4. Here, the input and output signals are multi-bit and one-bit respectively.

Now we consider the quasi-linear analysis of the second order $\Delta\Sigma$ modulator. From Fig. 3 (b) and Fig. 4, the output signal, Y , can be shown in the z transfer function as

$$Y(z) = z^{-1}X(z) + \frac{(z-1)^2}{z^2}E(z). \quad (3)$$

Thus, the output signal, $Y(z)$ is equal to the delayed input signal plus quantized noise as in Fig. 3 (b). The noise transfer function of the modulator is equal to

$$N(z) = \frac{(z-1)^2}{z^2}. \quad (4)$$

Obviously, the $\Delta\Sigma$ modulator is a high-pass filter for the quantized noise. By choosing a sufficiently high sampling frequency, the noise is shaped to lie outside the frequency range of the interest.

2.2 δ -operator

As discussed before, signal conversion with $\Delta\Sigma$ modulation requires a very fast sampling frequency. However, a very fast sampling frequency may result in long word-lengths for both

coefficients and variables within the controller, primarily because the differences between successive values of the input and output become increasingly small.

The discrete state-space form is a natural way of expressing the implementation equations for digital controllers. Transforming other expressions, e.g. continuous and discrete transfer functions, into this form is rather straightforward. The discrete state-space equations can be expressed as follows:

$$\begin{aligned} zX &= A_z X + B_z U; \\ Y &= C_z X + D_z U. \end{aligned} \tag{5}$$

However, there are particular problems of coefficient sensitivity with these conventional forms of control system processing using the shift operator z [8], a feature that becomes particularly critical with the high sample rates required for 1-bit signal processing. It has been recognized that the alternative forms using δ -operator overcomes a number of these problems since a difference operator is more like a derivative, resembling the continuous operator d/dt . Thus, the state-space equation becomes

$$\begin{aligned} \delta X &= A_\delta X + B_\delta U; \\ Y &= C_\delta X + D_\delta U. \end{aligned} \tag{6}$$

The δ -operator can be defined as

$$\delta = z - 1. \tag{7}$$

Some researchers also defined δ -operator as

$$\delta = \frac{z - 1}{T}, \tag{8}$$

in which T is the sampling period. This definition shows that there is a unification between discrete and continuous time since $\delta \rightarrow s$ (s is the Laplace operator) as $T \rightarrow 0$ [9]. However, it is in fact just a difference of scaling factor between these two definitions, and the first one is more direct since no multiplication is involved. Fig. 5 shows the diagram of δ -operator according to the first definition. It can be represented as

$$\delta^{-1} = \frac{z^{-1}}{1 - z^{-1}}. \tag{9}$$

Obviously the δ -operator is easy to implement as the equation

$$y(k + 1) = x(k) + y(k), \tag{10}$$

where k is the sample number, although it is possible to create a new state-space equation with a new set of states and a state matrix A_δ which is structured to give δ -operator arithmetic [5].

2.3 Controller Structure

Another important issue is to choose a suitable controller structure in order to reduce the circuit complexity for applications of system-on-chip. First let us see a canonic δ form [4], illustrated in Fig. 6, a particular feature being the coefficients on

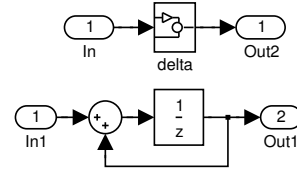


Figure 5: The δ -operator

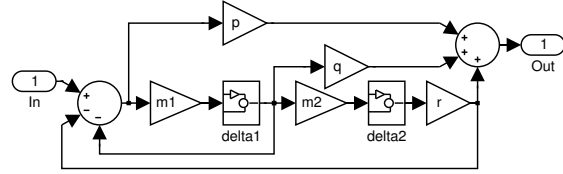


Figure 6: A second order structure for filters using the δ -operator

the "forward path" of the computations. This structure is definitely a perfect one for multi-bit processors because it gives low sensitivity coefficients and excellent scaling properties for the internal variables. However, there are many multi-bit multipliers involved in this structure. For the single bit processor, a simple structure is preferred to implement the $\Delta\Sigma$ modulation and perform 1-bit signal processing, in which the coefficients are moved from the feed-forward back to the feedback path, i.e. a more conventional canonic form. Fig. 7 shows the scheme.

As the output is in multi-bit format, it is necessary to remodulate the signal into 1-bit format. This can be accomplished easily by inserting a $\Delta\Sigma$ modulator after the output. Thus it shows that there are 5 'conditional sign switches' in the modified structure compared to 5 multi-bit multipliers in Fig. 6. Other structures are possible for $\Delta\Sigma$ signals processing [2], but this structure is the most simple and straightforward one. In this paper the controller will be implemented in this scheme.

3 Implementations

It has shown great advantages to implement $\Delta\Sigma$ -based control systems with bit-parallel architectures. However, for reducing the complexity of circuits and the price of chips further, the bit-

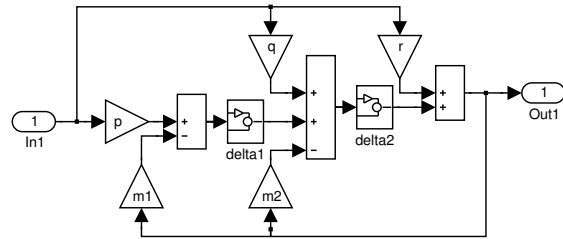


Figure 7: A second order structure with the modified δ form

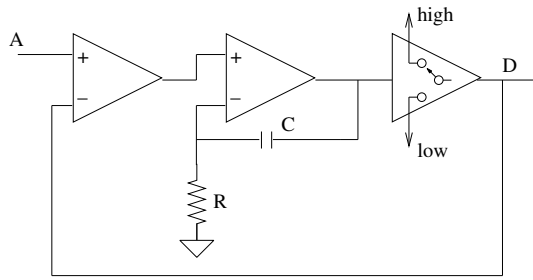


Figure 8: $\Delta\Sigma$ modulator using RC lowpass as integrator

serial approach is a more efficient solution. This is especially true when designing system-on-chips with FPGA. Using a bit serial architecture, it is frequently possible to pack a relatively complex function into a single FPGA. A throughput improvement may even be realized over an equivalent parallel structure implemented in FPGAs [1].

Using 1-bit signal processing, it is quite straightforward to implement the control system into bit serial-architecture. From the modified controller structure (Fig. 7), multiplication happens only in forward route and feedback route. As we referred before, multiplication is only a ‘conditional sign switch’ because the input and output signals are either 1 or -1. Coefficients are small because of the nature of δ operator and high sampling frequency. They can be stored in a simple floating format with typically 6 bits of mantissa and 5 bits of exponent [10]. Thus, the total operations involved are addition, negation, shift and delay. Because most operations are accompanied by an add operation, a shift-add element is promoted here.

3.1 1-bit A/D conversion

1-bit A/D conversion is a first step of control system processing, which performs the conversion of analogue signals to a bitstream. Here, the A/D converter is replaced by a simple $\Delta\Sigma$ modulator. Fig. 8 shows a first order $\Delta\Sigma$ modulator for analogue to digital conversion. It applies an RC-lowpass filter as an integrator and a simple comparator. The output switches to logic high when the input is greater than 0V, and to logic low when the input is less than 0V. High-order $\Delta\Sigma$ modulators can be expanded from this model.

3.2 Basic bit-serial blocks

A bit serial adder is constructed using a full adder with registers on both its carry and sum outputs. The registered carry output is wired back to the carry input. Fig. 9 (a) illustrates its schematic. In operation, the two words are simultaneously shifted least significant bit first into the remaining two inputs. The carry out from the addition of each bit is stored and then used in the summation of the next bit. The carry flip-flops must be cleared before each new data word so that the previous word’s carry is not added to the current addition.

Bit-serial 2s complement is required to compute the 2s comple-

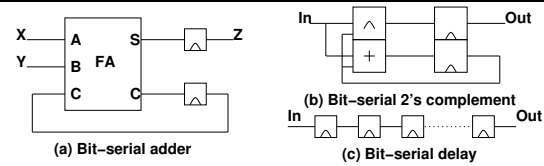


Figure 9: Basic bit-serial blocks

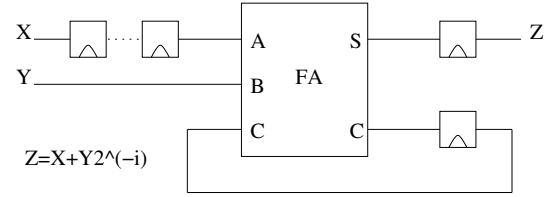


Figure 10: bit serial shift-adder

ment of the input. A bit-serial 2s complement circuit is shown in Fig. 9 (b). The input should be presented least significant bit first. The carry flip-flop must be reset before each input word.

The delay is useful for aligning words. It is simply a D flip-flop inserted into the data path for each bit of delay desired. Word delays are constructed from a string of bit delays equal in length to the number of bits in the word. A sample layout of a word delay is shown in Fig. 9 (c).

3.3 Shift-add element

Fig. 10 illustrates a schematic of the shift-add element. The 2^{-i} shift is accomplished using delays to realign the bits in X relative to the bits in Y . The data is presented *LSB* first. In order to shift Y i bits to the right, the bits in X must arrive at the processing element i bit times before the corresponding bit in X . By inserting the delay on the X path, X is shifted to the left relative to Y which is the as Y being shifted to the right relative to X . Notice the carry from the full adder is registered (effectively a 1-bit shift) and returned to the carry input of the adder. This is an implementation of the ripple transitive form of the adder. The carry flip-flops must be cleared before before each new data word.

4 A $\Delta\Sigma$ control system application

We have discussed $\Delta\Sigma$ modulation and developed a suitable controller structure before. Now let us demonstrate a d.c. motor control system, which controls the position of a rotating load with flexibility in the drive shaft. Fig. 11 shows the motor model.

A 4th order command-tracking controller has been designed including a PI, a phase advance and a notch filter to minimise the effect of the resonance caused by the flexibility. Thus, the overall control scheme can be illustrated as Fig. 12. Any sampling and computation delays introduced by the discrete controller are critical because both gain and phase margins are important

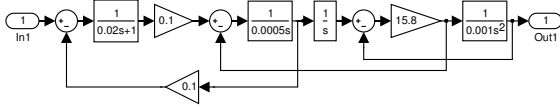


Figure 11: D.C motor physical model

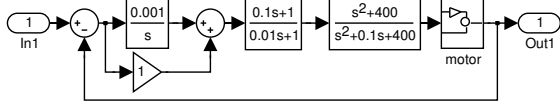


Figure 12: An overall control scheme

Feedforward coefficients	1kHz	10kHz
p_0	$4.0e^{-11}$	$4.0e^{-15}$
p_1	$4.0e^{-5}$	$4.0e^{-8}$
p_2	$4.0e^{-3}$	$4.0e^{-5}$
p_3	$1.0e^{-1}$	$1.0e^{-2}$
p_4	$1.0e^{+1}$	$1.0e^{+1}$
Feedback coefficients	1kHz	10kHz
q_0	0	0
q_1	$4.0e^{-5}$	$4.0e^{-8}$
q_2	$4.1e^{-4}$	$4.1e^{-6}$
q_3	$1.0e^{-1}$	$2.22e^{-2}$

Table 1: Controller coefficients at 1KHz and 10KHz

in the design.

In the δ form with the modified canonic structure (Fig. 13) the transfer function becomes

$$\frac{Y}{X} = \frac{p_4\delta^4 + p_3\delta^3 + p_2\delta^2 + p_1\delta + p_0}{\delta^4 + q_0\delta^3 + q_1\delta^2 + q_2\delta + q_3}, \quad (11)$$

where $p_0 \dots p_4$ are feedforward coefficients and $q_0 \dots q_3$ are feedback coefficients of the controller. These coefficients are calculated according to the sampling frequency. The higher the sampling frequency, the smaller the coefficients (see Table 1).

The procedure of the control system processing can be described as follows. Firstly the analogue signals (command and motor position) are sampled by a simple $\Delta\Sigma$ modulator and give a bitstream output. Then the signals feed into the digital controller and cause an update of the state variables so they are ready for next sample:

$$\begin{aligned} x_1 &= p_0u - q_3y1 + x_1; \\ x_2 &= p_1u + x_1 - q_2y1 + x_2; \\ x_3 &= p_2u + x_2 - q_3y1 + x_3; \\ x_4 &= p_3u + x_3 + x_4; \\ y &= p_4u + x_4, \end{aligned} \quad (12)$$

in which $y1$ is the modulated 1-bit signal of y . Then $y1$ can be directly output using pulse-density-modulation (PDM) technique to drive the motor.

To implement this controller into an FPGA, there are 8 switch logics for 'conditional sign switch', 7 bit-serial adders, 11 bit-serial shift-adders. Obviously, the whole system architecture is

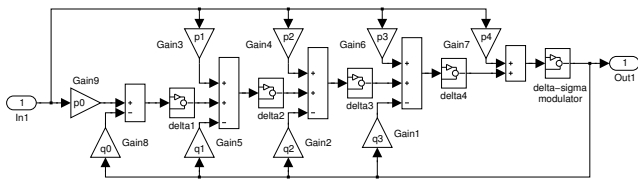


Figure 13: Modified controller structure

quite simple compared to 8 multi-bit multipliers and 13 multi-bit adders. The architecture should be more simple because of the elimination of A/D and D/A converters.

Fig. 14 shows simulation results of the step response of the closed-loop system at a sampling frequency of 10kHz. Graph (a) gives results of continuous model, discrete model with multi-bit processing and discrete model with 1-bit processing. The difference between the results is very slight. Graph (b) is an expanded particular of the response, and we can see that there are small differences in the residual oscillation of the flexible mode, but these are within 0.1%. This means the 1-bit signal processing is applicable for real-time control.

5 Conclusion

One-bit signal processing has been shown applicable for real-time control. With bit-serial architecture, it is also simple and easy to implement. This technique is quite promising for system-on-chip solutions for embedded controllers as the valuable silicon area can be saved to reduce the chip cost.

The dynamic range of this type controller largely depends on the sampling frequency and the noise shaping ability of the modulator. For the future research, as the sampling frequency cannot be increased indefinitely, it is still necessary to improve the structure of $\Delta\Sigma$ modulators. At the same time, the $\Delta\Sigma$ based control system is expected to be implemented practically for a Magnetic Levitation (MAGLEV) demonstrator, where a magnetically-suspended vehicle will be controlled by a single chip.

6 Acknowledgement

This work is being undertaken with funding from EPSRC, the UK research council, grant number GR/R38002/01.

References

- [1] Raymond J. Andraka. FIR filter fits in an FPGA using a bit-serial approach. In *Proc. 3rd PLD Conference*, 1993.
- [2] J.A.S. Angus and S. Draper. An improved method for

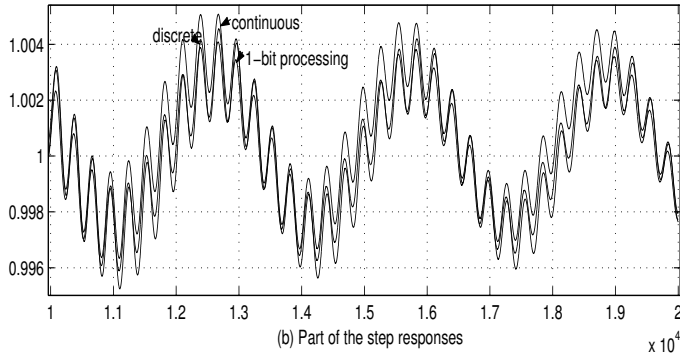
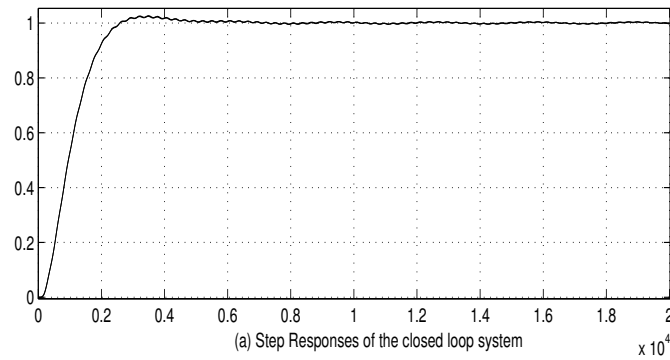


Figure 14: Simulation results of step response

[10] Rene A. Cumplido Parra. *On the design and implementation of a control system processor*. PhD thesis, Loughborough University, 2001.

directly filtering $\sigma - \delta$ audio signals. In *Proc. AES 104th convention*, Amsterdam, The Netherlands, 1998.

- [3] R.M. Goodall. Sampling and computation delays in real-time control. *Unpublished memorandum*, October 2002.
- [4] R.M. Goodall and D.S. Brown. High speed digital controllers using an 8-bit microprocessor. *Software and Microsystems*, 4:109–116, 1985.
- [5] R.M. Goodall and B.J. Donoghue. Very high sampling rate digital filters using the δ operator. *IEE Proceedings-G*, 140(3), 1993.
- [6] R.M. Goodall, S.R. Jones, R. Cumplido-Parra, F. Mitchell, and S. Bateman. A control system processor architecture for complex LTI controllers. In *Proc. 6th IFAC Workshop AARTC 2000*, pages 167–172, Palma de Mallorca, Spain, May 1998.
- [7] D.A. Johns and D.M. Lewis. Design and analysis of Delta-Sigma based IIR filters. In *IEEE transactions on circuits and systems-II: analog and digital signal processing*, volume 40, April 1993.
- [8] B. Liu. Effect of finite wordlength on the accuracy of digital filters — a review. In *IEEE Transactions. Circuit Theory*, volume CT-18, pages 670–677, 1971.
- [9] R.H. Middleton and G.C. Goodwin. *Digital control and estimation — a unified approach*. Prentice Hall, USA, 1990.