A novel digital-to-analog conversion technique using duty-cycle modulation

Bertrand Lonla Moffo, Jean Mbihi[(1)], Léandre Nneme Nneme and Martin Kom

Abstract—The novel digital-to-analog conversion technique studied in this paper, is founded on the digital duty-cycle modulation principle. It consists of a simple digital duty-cycle modulation program loaded in the flash memory of a microcontroller and an external analog low pass filter. Both pulse width and modulation period, simultaneously evolve according to the variation of the modulating input. Then, under the proposed modulation technique, high frequency harmonics with rank greater than a number *Nh*, are significantly eliminated from the spectrum of the resulting modulated wave. As a merit, the digital modulated wave to be filtered is computed directly and very fast from its discrete Fourier series model. Thus, the related digital modulation and generation routines do not require an intermediary n-bit processing hardware logic. The predicted and experimental results obtained when testing a prototyping digital-to-analog conversion workbench driven by a PIC30F6014A microcontroller, are presented and compared. These results show the feasibility and high quality of digital-to-analog conversion via digital duty-cycle modulation.

Keywords—Digital-to-analog conversion, digital duty-cycle modulation, fast Fourier series, analog filter, DsPIC30F6014A development kit.

I. Introduction

The digital-to-analog (D/A) conversion is an instrumentation function, allowing the transformation of a digital data flow into an analog signal using a mix of software and hardware processing resources. Although the first generations of D/A converters were implemented in single function chips, most modern D/A processes are embedded in microcontrollers ([1]-[4]). However, independently of the implementation technology, a variety of D/A conversion techniques have been invented over years, each of which being more suitable for a few application areas. These techniques differ greatly depending on a number of characteristics including the signal-to-noise ratio, the resolution, the input bandwidth, the output range, and the building cost.

Architecturally speaking, the DAC (D/A conversion) devices found in the market today, fall into two main classes known as direct and indirect DAC as presented in Fig. 1. More details about DAC architectures could be found in [5]-[6].

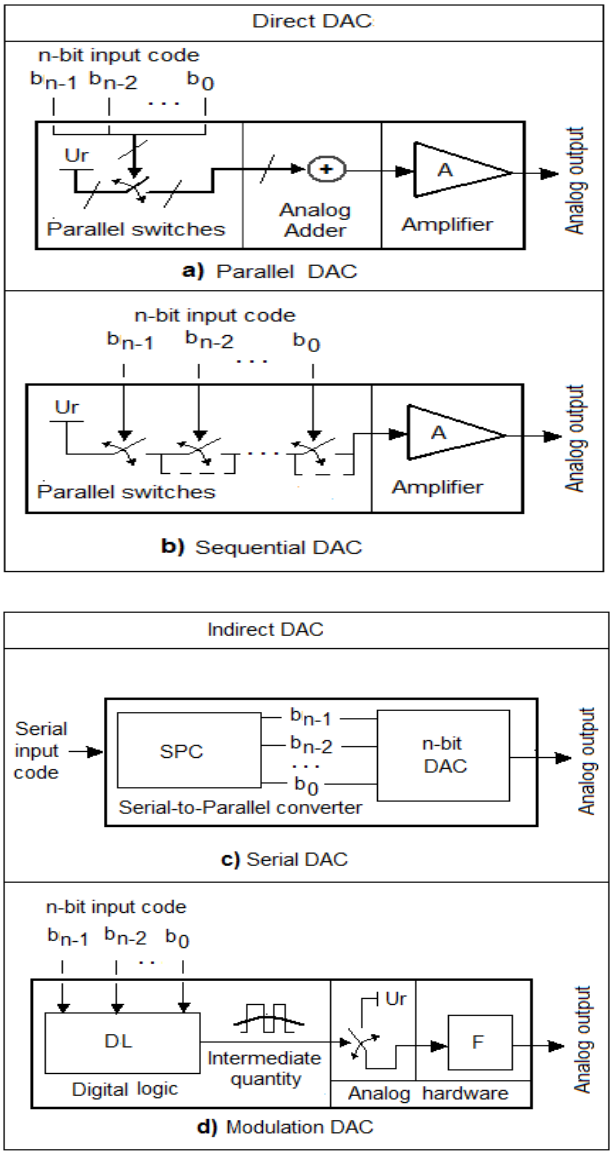


Fig. 1: DAC architecture types.

(1) Jean Mbihi *is the corresponding author*. He is with the Research laboratory of industrial computer science and automation, engineering, in the advanced teachers training college for technical education of the University of Douala, Cameroun (e-mail: mbihidr@yahoo.fr).

Bertrand Moffo is with the Research laboratory of industrial computer science and automation engineering, in the advanced teachers training college for technical education of the University of Douala, Cameroun (e-mail: moffolb@yahoo.fr)

Léandre Nneme Nneme is with the Research laboratory of industrial computer science and automation engineering, in the advanced teachers training college for technical education of the University of Douala, Cameroun (e-mail: leandren@gmail.com).

Martin Kom is with the Research laboratory of biomedical electronics, in the National Polytechnics High school of the University of Yaounde I, Cameroun (E-mail: kommar@yahoo.fr).

In direct DAC, the bits of the input code are processed simultaneously (case of parallel DAC in figure 1(a)), or sequentially (case of sequential DAC in figure 1(b)), using a controlled switching logic under a reference signal Ur. As an implication, the analog output results from mixing the multilevel internal signals produced. Although parallel DAC run fast, the related n-bit code to be converted simultaneously might lack synchronization and become unreliable under high frequency inputs. Conversely, the merit of sequential DAC relies on the serial processing of information for the sake of better reliability at the expend of a lower bandwidth.

In indirect DAC, the n-bit input code is transformed into an intermediary quantity, before being translated into an analog output. The class of indirect DAC consists of serial and modulation DAC. In the case of serial DAC (see figure 1(c)), the intermediary quantity is a custom serial code, in which case the original n-bit code should be translated into a target serial code by an external host processor, before been processed via a serial-to-DAC transform with an n-bit internal processing logic [7]. In the case of modulation DAC (see figure 1(d)), the value of the n-bit input is encapsulated in a modulated signal, then a terminal analog filter is used to extract the original modulating signal. Unlike serial DAC, modulation DAC are widely used and implemented from PWM (pulse width modulation) technique, in modern PICs (programmable integrated circuits) and even in CPLD (Complex programmable logic devices). The typical examples of such devices are PIC12F683, PIC17C42, PIC16C620, PIC18F87J60 and CPLD-Max II and TMS320F280x. In addition, the great challenge brought by a pioneering work [8], has been the extension of the standard single output DPWM-based technique to multiple output signals, using a PIC18F8720 programmable integrated circuit connected to a host PC computer via a RS232 communication link.

PWM-based DAC are characterized on by a an internal n-bit input code, which is processed usually using a micro-hardware, consisting of logic gates, flip-flops, n-bit registers, m-bit up/down counters and more. Thus, although PWM-based DAC have been commonly used as a tradition in instrumentation engineering, they still have a number of weaknesses including : a) the n-bit internal processing architecture per single channel, b) the complexity of the processing hardware logic, and c) the greedy frequency spectrum of the modulated wave to be filtered, due to a fixed modulation frequency for all modulation values. Thus, a well tested novel indirect DAC structure, founded on duty-cycle modulation (DCM), is studied in depth in this paper. The DDCM (digital DCM) principle is a discrete version of an analog instrumentation technique initiated in [9], and could be taught off as the reversible process of the class of DCM-based ADC (analog-to-digital conversion) systems recently initiated in [10]-[12]. In Section II, the principle of DDM-based DAC is presented, whereas its numerical analysis is conducted in Section III, in order to present the predicted results and related performance. Finally, section IV is devoted to both experimental studies and results obtained when implementing the first realization of the novel DDCM-based DAC using a DsPIC30F6014A development kit.

II. Principle of DDCM-based DAC

The proposed DDCM-based DAC scheme is presented in Fig. 2. The whole conversion process with digital modulating input x is presented in Fig. 2(a). Viewed at discrete time k, the first stage consists of a DDCM routine with output $x_M(k)$ as shown in Fig. 2(b). Then, the resulting DDCM wave $x_M(k)$, is transformed into a sequence of 1-bit codes $b_m(k)$ as indicated in Fig. 2(c). Furthermore, $b_m(k)$ is written to the 1-bit output register of a microcontroller, and the resulting analog switching wave $b_m(t)$ shown in Fig. 2(d), is processed by an external low pas filter in order to reconstruct the analog version ***xa*** of the digital modulating input.

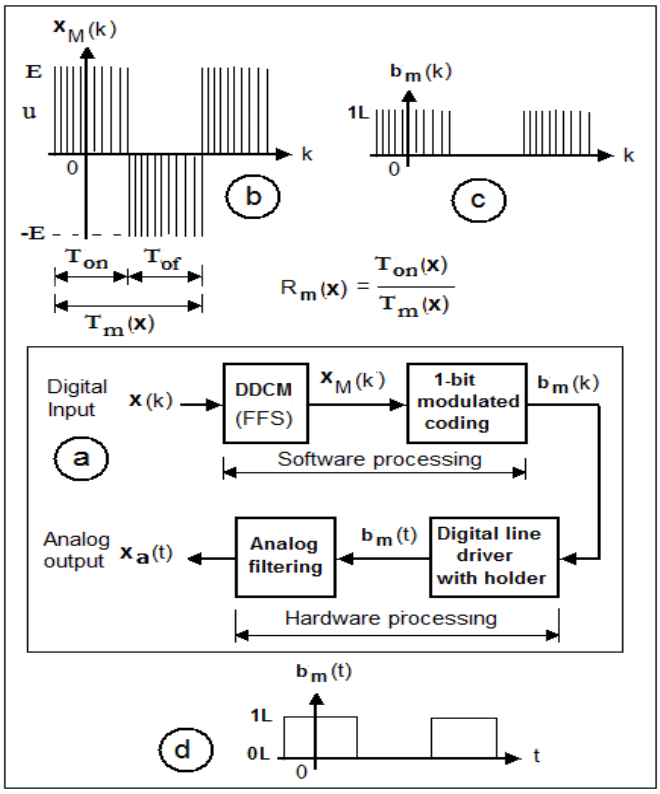


Fig. 2 : The DDCM-based D/A conversion scheme.

Thus, the DDCM wave $x_M(k)$, is characterized on by the duty-cycle function,

$$R_m(x,k) = T_{on}(x,k) / T_m(x,k) \tag{1}$$

and its discrete Fourier series is given by :

$$x_M(x,k) = \underbrace{(2R_m(x)-1)E}_{\text{Low frequency term}} + \underbrace{\left(\frac{4E}{\pi}\right)\sum_{n=1}^{\infty}\left(\frac{\sin(n\pi R_m(x))}{n}\cos\left(2\pi n\frac{(kT)}{T_m(x)}\right)\right)}_{\text{High frequency terms}} \tag{2}$$

where $T_{on}(.)$ and $T_{of}(.)$ stand for the ON/OFF time for which $x_M(.)$ is E or $-E$, $T_m(.) = T_{on}(.) + T_{of}(.)$ being the modulation period, with related frequency $f_m(.) = 1/T_m(.)$. It is important to mention that the relation (2) related to DCM is valid also for a standard PWM. However, the difference between the two is dictated by their respective duty-cycle structure (1). Following

our prior works on single and multichannel ADC (analog-to-digital conversion) via duty-cycle modulation [10]-[12], the duty-cycle $R_m(.)$ and the related modulating frequency $f_m(.) = 1/T_m(.)$ in (2) are given under an admissible modulating range by:

$$R_m(x) = \frac{\frac{\alpha_1(1-\alpha_1)}{E(1-\alpha_1^2)}}{\log\left(\frac{1+\alpha_1}{1-\alpha_1}\right)} x + \frac{1}{2} \quad (3)$$

$$f_m(x) = \frac{1}{T_m(x)} = \frac{1}{\tau\ Log\left(\frac{(\alpha_2 x)^2 - ((1+\alpha_1)E)^2}{(\alpha_2 x)^2 - ((\alpha_1 - 1)E)^2}\right)} \quad (4)$$

In (3) and (4), $\alpha_1 = 1-\alpha_2$ is a design parameter, whereas E stands for the amplitude of the modulation wave, τ being a given time constant. In addition, it is easy to show from the derivative where of (4) that $f_m(x)$ is convex over the modulating space with maximum for $x = 0$ volt (or $R_m(x) = 0.5$ equivalently). In order to understand the mayor merit of the DCM technique, it is necessary to compute and compare the normalized spectra of both DCM and PWM waves computed from (2), given a fixed PWM frequency equal to $f_m(0)$ in (4). While for DCM technique, $R_m(x)$ is computed from (3) given (4), the equivalent quantity is equal to $R_m^{Pwm} = -(1/(2\,u^{osc}))\ x + 0.5$ in the PWM case, where u^{osc} stands for the amplitude of the triangular oscillator from which the PWM wave might be generated. For the sake of numerical simulation needs, the parameters used are as follows : $\tau = 33$ µs, a1 = 0.2495, a2 = 0.7505, $E = 9$ volts, and $u^{osc} = 6$ volts. The respective spectra obtained are presented in Fig. 3. The quality of the PWM spectrum shown in Fig. 3(a), is widely better than that of the standard PWM observed on Fig. 3(b). The potential discovery behind this results is that, the fact of using a constant modulation frequency over the whole modulating space as in a standard PWM, is a great weakness with negative effects on the spectral behavior of the modulated wave to be filtered. Conversely, the Fourier series (2) given (3) and (4) could be thought off as a virtual FFS (Fast Fourier Series) transform from Fig. 3(b) to Fig. 3(a). Thus, under DCM control, the first 10 harmonics of (2) are sufficient for a direct and very rapid reconstruction of the discrete DCM wave to be filtered. This potential discovery indicates that under DCM, a real time processing of (2) could be very fast while it might be intractable from standard PWM controls.

III. Numerical Analysis of DDCM-based DAC

The numerical simulation process conducted in this Section using Matlab software, could be thought off as a virtual demonstration of the DDCM-based DAC described in figure 2. The parameters used for the simulation of DDCM are E = 9 volts, $f_m(0) = f_{pwm} = 29$ kHz in (3), a1 = 0.2495 = 1-a2, $\tau = 33$ µs, $N_h = 10$ (10 first 10 harmonics), $f_e = 1/Tech = 500$ kHz (sampling frequency), $f_{clock} = 20$ MHz

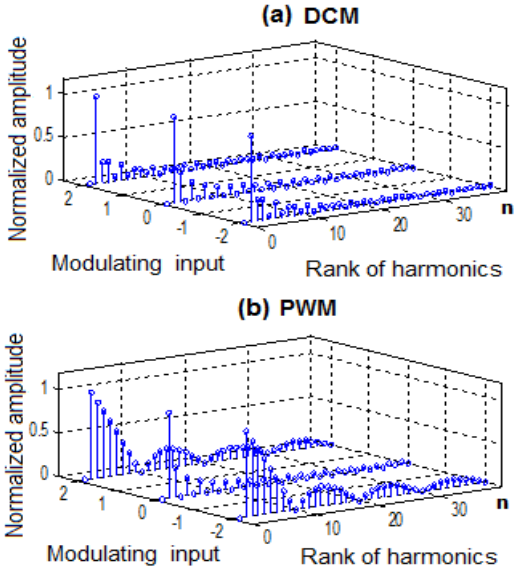


Fig. 3: Normalized spectrum DCM and PWM waves.

(clock source), $Np = 4$ or 5 (number of modulating periods required for a each simulation sequence), $x_m = 4$ volts (amplitude of the modulating wave to be processed), and $f_{BW} = 200$ Hz (modulating bandwidth). In addition, the the low-pass filter considered in this study for analog demodulation is characterized on by a transfer function :

$$G(s) = \frac{12764}{0.0002754\,s^2 + 3.692s + 10000} \quad (5)$$

a static gain $K_f = 1.2764$, and a cut off frequency $f_c = 1$ kHz. The same set of data will be used further for a real time implementation of a prototyping DDCM-based DAC using a DsPIC30F6014A development kit.

The digital simulation results obtained under a sampling frequency $f_e = 500$ kHz, are presented in Fig. 4 for different types of modulating signals : 100 Hz sine wave (Fig. 4(a)), 200 Hz sine wave (Fig. 4(b)), 100 Hz triangle (Fig. 4(c)) and 200 Hz triangular wave (Fig. 4(d)). In each case, the graph of the DDCM wave $\boldsymbol{x}_M(k)$ resulting from the FFS (2) given (3) and (4) is shown, followed by the comparison of the graphs of the modulating signal x and that of the overall response $\boldsymbol{x}_a$ available at the filter output. In all cases, the overall analog output $\boldsymbol{x}_a$ is a rigorous copy of the modulating $\boldsymbol{x}$. Hence, at this stage of the analysis, the feasibility of the novel DDCM-based DAC is proven, at least in the virtual word. In addition, a few parameters required for the performance analysis are denoted as follows : f_{clock} (main clock frequency), $f_m(x)$ (DCM frequency), A_m (amplitude of the DCM signal) and A_{rip} (amplitude of the analog output ripple). Given these parameters, the performance of the DDCM-based DAC is dictated by the following uncertain quantities : $\Delta_m(x)$ (DCM voltage uncertainty), $\Delta_{rip}(x)$ (ripple voltage uncertainty), $\Delta(x)$ (overall uncertainty) and $N(x)$ (DAC resolution in bits).

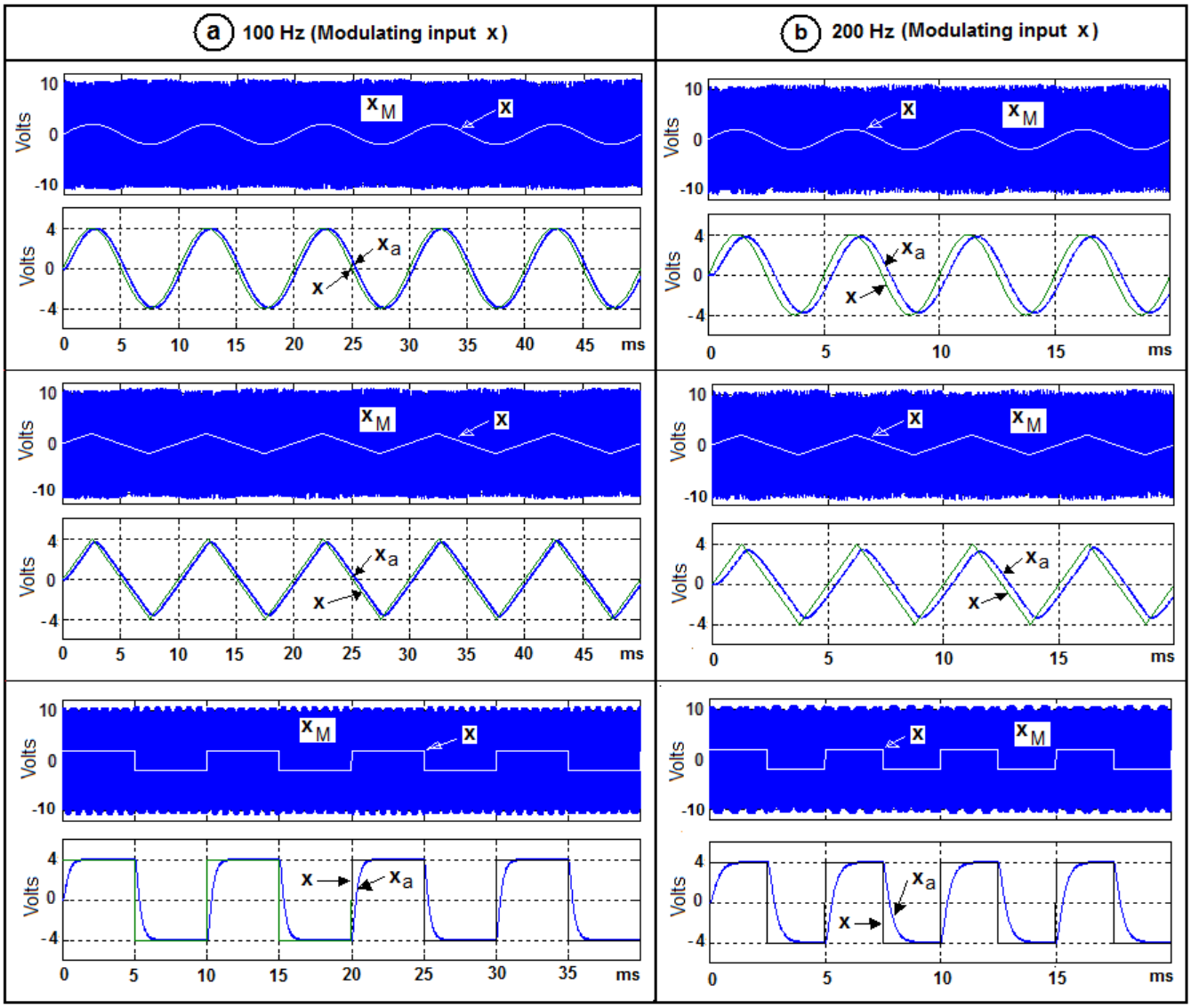


Fig. 4: Numerical simulation results for DDCM-based D/A conversion.

These quantities are defined as follows [13] :

$$\Delta_m(x) = \frac{f_m(x)}{f_{clock}} \tag{6}$$

$$\Delta_r(x) = \frac{A_{rip}}{A_m} \tag{7}$$

$$\Delta_m(x) = \frac{f_m(x)}{f_{clock}(x)} + \frac{A_{ripple}(x)}{A_m(x)} \tag{8}$$

$$N_m = -\log_2\left(\frac{f_m(x)}{f_{clock}(x)} + \frac{A_{rip}(x)}{A_m}\right) \tag{9}$$

Table 1 presents a summary of the numerical analysis results, obtained from computing uncertain quantities (6)-(9) for $R_m(x)$ in (3) and $f_m(x)$ in (4) for $f_{clock} = 20\ Mhz$, $A_m = E = 9\ V$, $x \in [-6.70\ \ 6.70]$ with fs = 100 Hz.

Table 1 : Numerical analysis of the performance of DDCM-based DAC.

f_{clock} = 20 MHz, A_m = 9 V and fs = 100 Hz for x						
x (Volt)	$R_m(x)$	$f_m(x)$ (kHz)	Δ_{dcm} (10^{-3})	Δ_{rip} (10^{-3})	Δ_m (10^{-3})	N_m (bits)
-6.70	0.2013	18.888	0.944	2.6	3.544	8.140
-6.00	0.2343	21.153	1.100	2.3	3.400	8.200
-5.00	0.2802	23.850	1.200	2.0	3.200	8.277
-4.00	0.3248	25.997	1.300	1.9	3.200	8.288
-2.0	0.4128	28.800	1.400	1.7	3.100	8.288
0.0	0.5000	29.723	1.500	1.6	3.100	8.288
2.0	0.5872	28.800	1.400	1.7	3.100	8.288
4.0	0.6752	25.997	1.300	1.9	3.200	8.277
5.0	0.7199	23.850	1.200	2.0	3.200	8.277
6.0	0.7657	21.153	1.100	2.3	3.400	8.200
6.70	0.7987	18.888	0.944	2.6	3.500	8.140

The last colon of Table 1 indicates that the effective overall (input-output) resolution of the DDCM-based D/A conversion technique, is 8 bits for any admissible modulating input x.

This information is important since an 8-bit resolution within the DCM amplitude range [-*E* *E*] with *E* = 9 volts, appears to be suitable for most low precision applications in instrumentation engineering.

The real time DDCM program could be implemented using one of the algorithms shown in Fig. 5, where $C_0(x) = (2\ R_m(x) - 1)\ E$ and $C_n(x) = (4\ E/\pi)\ \sin(n\ \pi\ R_m(x))/n$. In figure 5(a), the DDCM signals is computed offline, before being used sequentially in real time mode. This first mode requires a storage memory with higher capacity. In Fig. 5(b), as soon as each sample of the DDCM is computed, it is

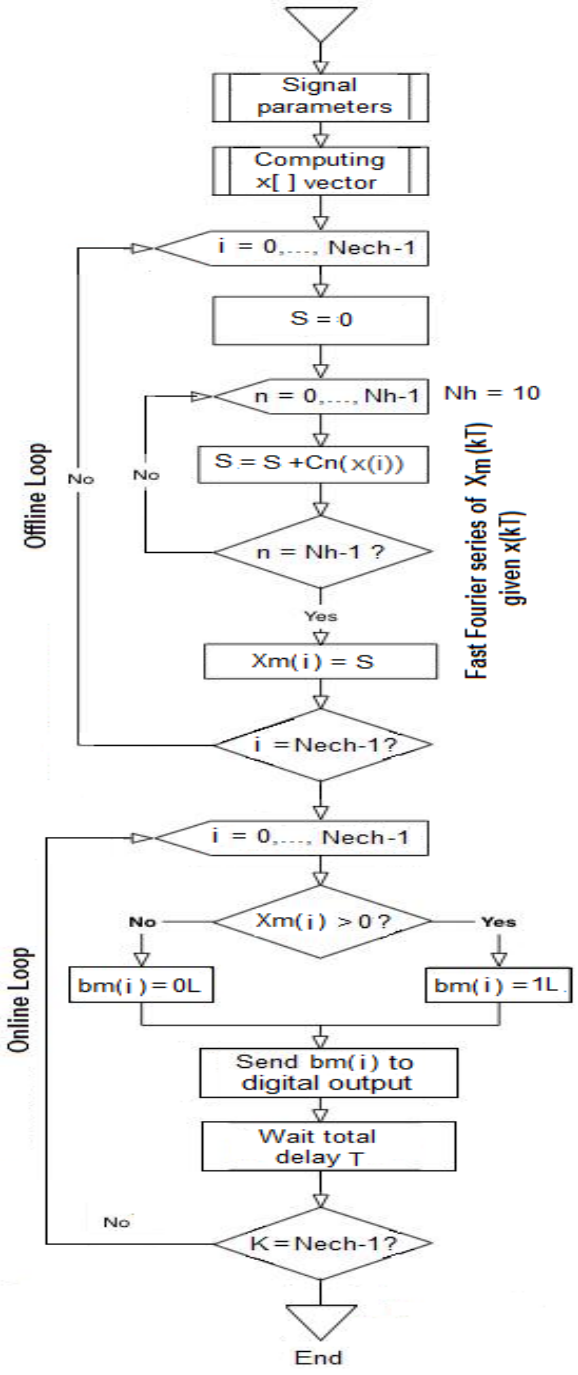


Fig. 5 (**a) :** Offline DDCM algorithm.

.

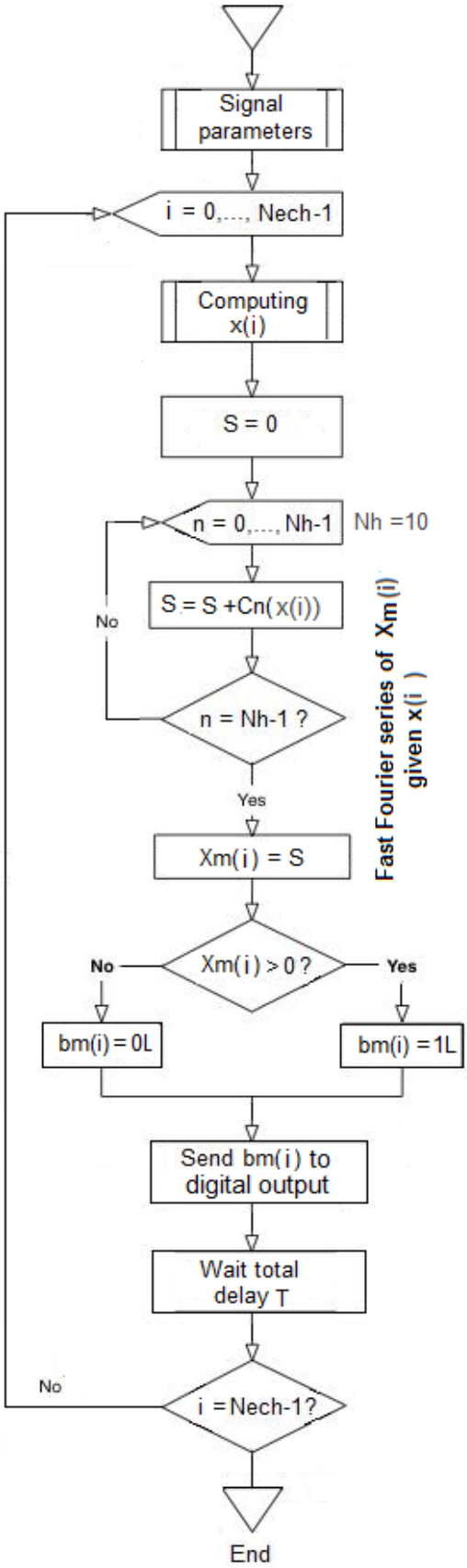


Fig. 5 (**b) :** Offline DDCM algorithm.

immediately applied to the 1-bit output register of the microcontroller; before being filtered. The second mode might be more suitable in a digital feedback control loop. However, it requires a high frequency clock source. In this section, the algorithm presented in Fig. 5(a) is used for real time implementation of DDCM routines. The source codes of DDCM routines, are provided in the Appendix for interested readers in C programming language.

IV. EXPERIMENTAL TEST OF DDCM-BASED DAC

The experimental workbench used built to test the realistic nature of the proposed DDCM-based DAC is presented in Fig. 6. It consist of : a PC with the installed Microchip MPLAB X IDE and XC16 compiler, a DsPIC30F6014A development Kit (with a 20 MHz clock source) connected to a PC via an USB uploaded link, an external analog filter with transfer function (5), and a LW2042C digital storage oscilloscope with 40 MHz bandwidth for the measurement and monitoring of the analog output *x*a.

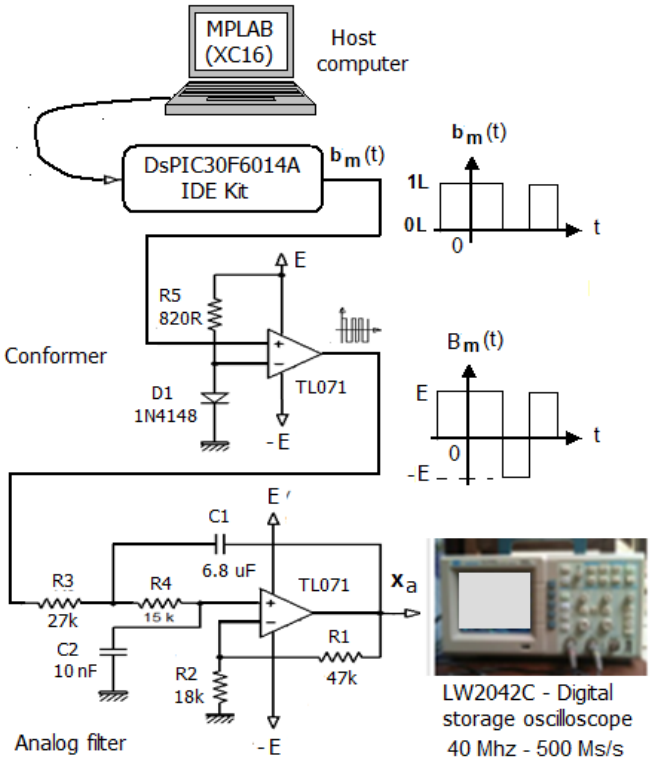
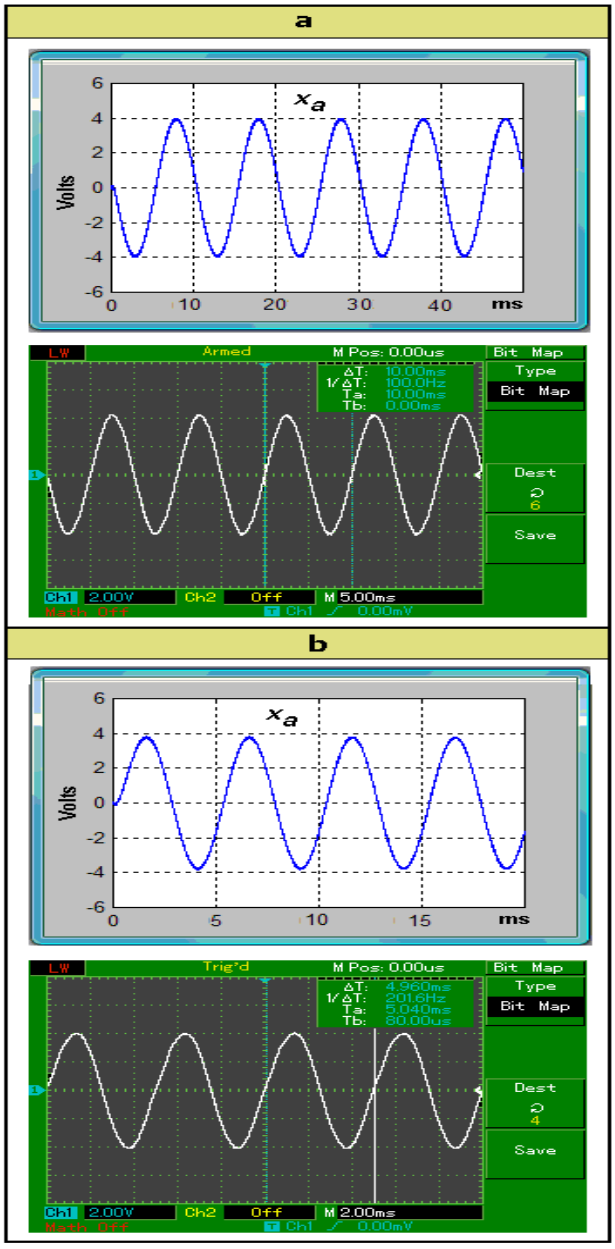


Fig. 6: Workbench for DDCM-based DAC.

Following the set of data used in previous numerical simulations, the sampling loop is driven by a 500 kHz Timer, while the FFS loop is limited to the first $N_h = 10$ harmonics in (2). It is important to recall that the tive filter used in figure 6 might be replaced by a second order RC filter as it is usually the case for most low precision PWM-based DAC [13].

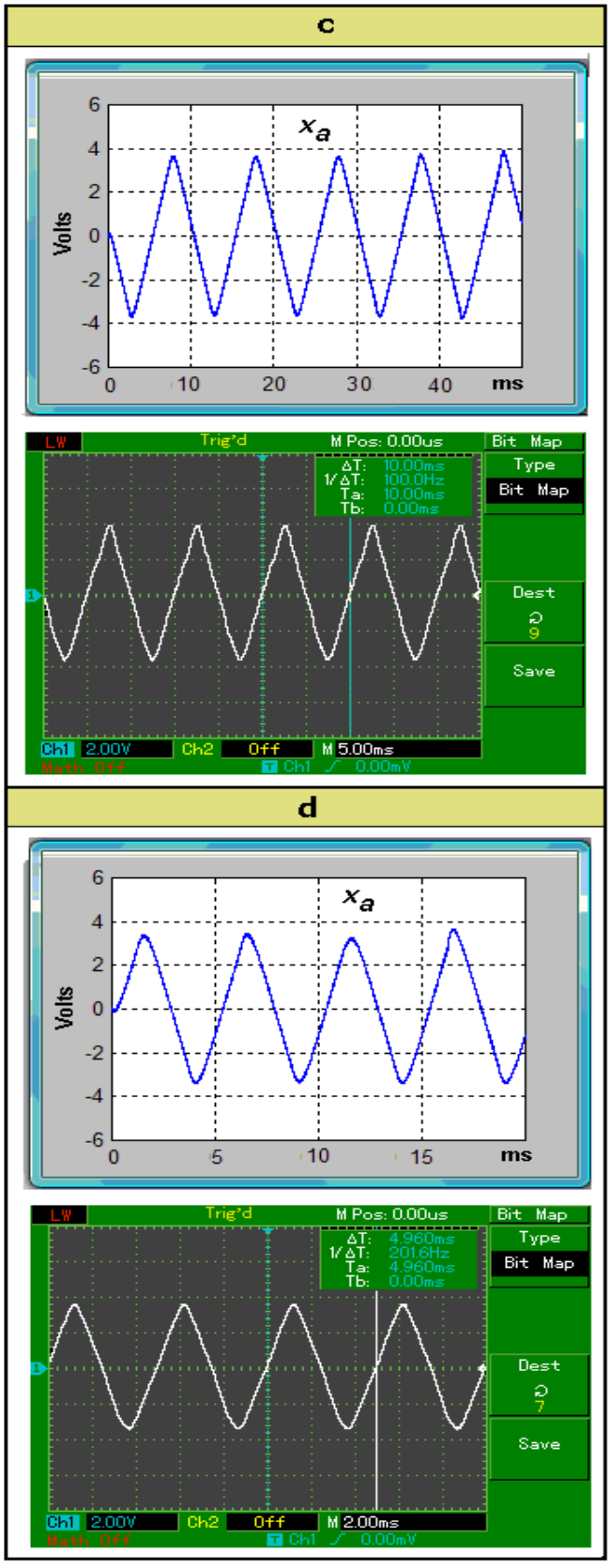
The predicted results shown previously in Section III (see Fig. 4), and the related experimental results obtained when processing identical digital modulating signals, are compared in Figs. 7, 8 and 9. In each case, the graph of the predicted output x_a is displayed at the bottom of the experimental response observed on the digital oscilloscope screen. Both predicted and experimental graphs have been set up to identical scales for that sake of visual comparison clarity.



(a) 100 Hz **(b)** 200 Hz

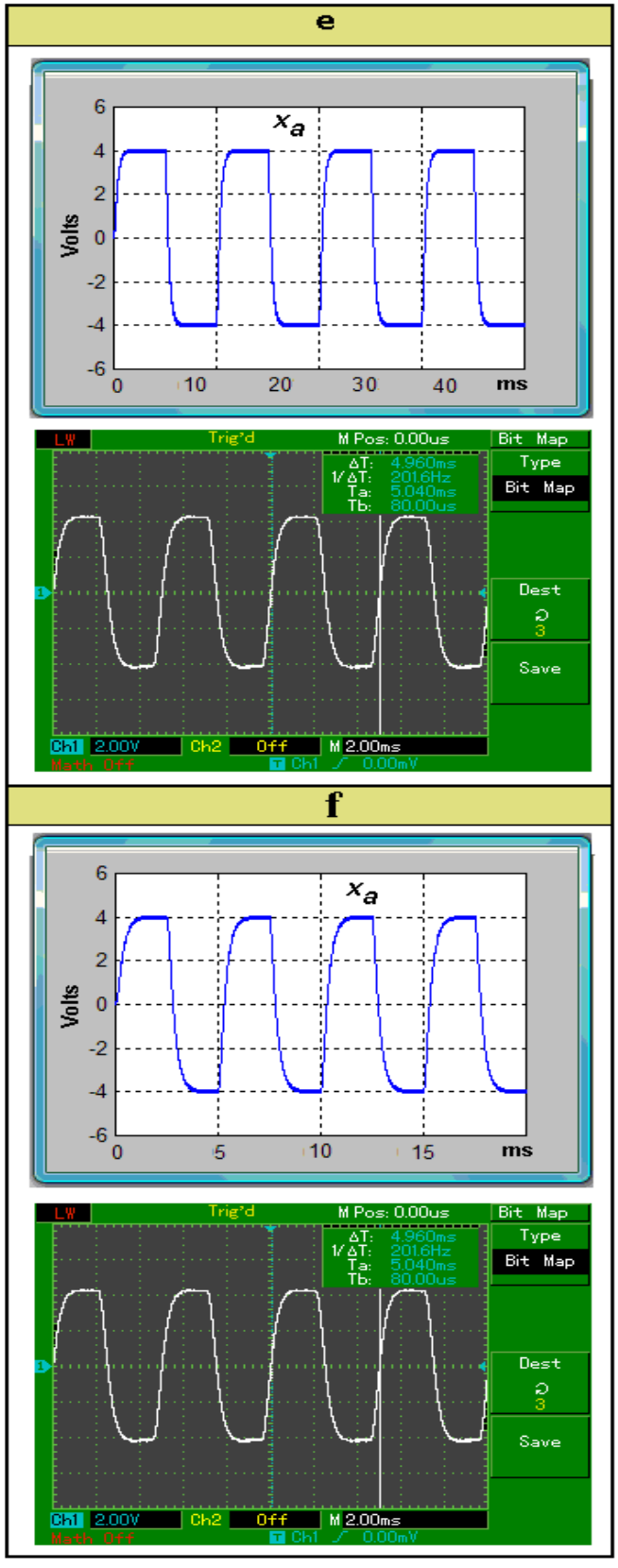
Fig. 7: Predicted and experimental results of the DDCM-based DAC of sine signals.

Fig. 7 presents the results related to the DDCM-based DAC of 100 Hz (see Fig. 7(a)) and 200 Hz (see Fig. 7(b)) sine signals, whereas the results obtained for 200 Hz (see Fig. 7(b)) triangle signals are presented in Fig. 8. Furthermore, even 100 Hz and 200 Hz square signals involving wider frequency spectra that those of a triangle wave, have been converted almost successfully as shown in Fig. 9.



(a) 100 Hz **(b)** 200 Hz

Fig. 8: Predicted and experimental results of the DDCM-based DAC of triangle signals.



(a) 100 Hz **(b)** 200 Hz

Fig. 9: Predicted and experimental results of the DDCM-based DAC of square signals.

The predicted and experimental results obtained and presented in Figs. 7, 8 and 9, show the realistic nature as well as the high quality of the novel DDCM-based DAC technique initiated in this paper.

V. CONCLUSION

The feasibility and quality of the novel DDCM-based DAC technique presented in this paper, have been shown using a mix of analytical reasoning, numerical analysis and experimentation. It would be interesting to find optimal parameters related to both the embedded DDCM code and the analog filter. In addition, the 1-bit nature of intermediary digital processing steps, might facilitate its extension to the n-channel D/A conversion via a single digital word consisting of of *n* DDCM bits. It would be also a challenge to implement multichannel DDCM DAC, from parallel computing technology, in order to extend their use to wide bandwidth application areas. These problems remains unsolved and might be explored in future research opportunities.

APPENDIX

C routines for DDCM-based DAC

```
void parameter (void)  // Constant parameters
{  Fe = 500000.0;       // Sampling frequency
   Fs = 200.0;          // Modulating frequency
   Te = 1 / Fe;         // Modulating period
   E = 9.0;             // Dcm amplitude
      a1 = 0.2495;     a2 = 0.7505;  // Dcm Constants
     C1 = (1 - a1) * E;    C2 = (1 + a1) * E;
      G = pow(C2, 2);      H = pow(C1,2);
}

float Xm(float u, unsigned long m)  // DDCM of
{                                   // u ← Vout, m ← i
  int n;  floats P;  float A;
  A = a2 * u;   A = pow(A, 2);  A = log((A - G) / (A - H));
  Rm = 0.04352379875*u + 0.5;  // Linear model.
  P = 10000 * 0.000000033 * A;
            s = 0;
  for (n = 1; n < 11; n++)     // Nh = 1
  {  1
     s = s + (4 * E / (n * 3.14))  * (Sin(3.14 * n * Rm)) *
        (Cos(n * 2 * 3.14 * m * Te / P));
  }
     s = s + 2 * E * Rm - E;     return( s );
}

void Modula(float u, unsigned long num)
{    If (Xm (u, num) > 0) // Call DDCM
   {              // for u ← Vout, num ← i
       bmTab[num] = 1;  // offline high state
    }
   else {bmTab[num] = 0; } // offline low state
}

void Sinus(void) //  e.g., case of Sine wave
{    float Ampl, offset, Vout;
     Ampl = 4.0; offset = 0.0;
     Nech = Fe / Fs; // Number of sample
 for(i = 0; i < Nech; i++)
 {
   Vout = offset + Ampl *  Sin(2 * 3.14 * Fs *  i * (1 / Fe));  //
   Modula(Vout, i); // Call for DDCM of each Vout, i
 }
}

void Generat(void) // DDCM generation
 { while(1) // Infinite outer loop
   {  for(i = 0; i < Nech; i++) // DDCM loop
      {
         PORTB = bmTab[i]; // 1-bit output
  _Delay_cycles(38); // Sampling interval 20MHz/38 = 500kHZ
      }
   }
}
```

REFERENCES

[1] A. Prodic, D. Maksimovic and R. W. Erickson, "Design and implementation of a digital PWM controller for a high frequency Switching DC-DC Power converter, "27th annual conference of the IEEE Industrial Electronics Society, 2001, pp. 893-898.

[2] *D. M. Alter*, "Using PWM Output as a Digital-to-analog converter on a MS320F280x Digital Signal controller", Application report SPRAA88A, September, Texas Instruments, 2008.

[3] M. Verle, "PIC Microcontrollers - Programming in C" , MikroElektronika, 1st edition (2009).

[4] Altera Corporation, "Controlling analog output from a digital CPLD using PWM", White paper WP-01085, November 2008.

[5] A.S. Sedra, K.C. Smith, "Microelectronic circuits", *Oxford University Press; Fourth Edition edition (Jun 1 1997)*.

[6] Y. Matsuya, K. Uchimura, A. Iwata, and T. Kaneko, "A 17-bit Oversampling D-to-A conversion technology using multistage noise shaping, "IEEE journal of solid-state circuits, pp 969-975, August 1989.

[7] Actel Corporation, "Simple digital to analog converter », Application note AC280, 2006.

[8] C. P. Hung, W. G. Liu, H. J., Jia-Wai Chen and B. Mi. Chiu, "PIC-based multi-channel PWM signal generation method and application to motion control of sx feet robot toy", International Journal of circuits, systems and signal processing, issue 2, vol 3, pp. 73-83, 2009.

[9] J. Mbihi, B. Ndjali, and M. Mbouenda, "Modelling and Simulations of a Class of Duty Cycle Modulators for Industrial Instrumentation", *Iranian Journal of Electrical and Computer Engineering, vol.4, N°2, 2005*.

[10] J. Mbihi, F. Ndjali Beng, M. Kom, and L. Nneme Nneme "A Novel Analog-to-Digital Conversion technique using Nonlinear Duty-Cycle Modulation", International Journal of Electronics and Computer Science Engineering, Volume 1, Number 3, pp. 818-825, 2012.

[11] J. Mbihi, L. Nneme Nneme, "*A* multi-channel analog-to-Digital conversion technique using parallel duty-cycle modulation", International Journal of Electronics and Computer Science Engineering, Volume 1, Number 3, pp. 826-833, 2012.

[12] J. Mbihi and A. Motto, "Informatique industrielle Instrumentation virtuelle assistée par ordinateur : Principes et technique" 240 p., © Ellipses Edition, 2012, Paris, France.

[13] B. Nisarga, "PWM DAC Using MSP430 High Resolution Timer", Application report SLAA497, July 2011, www.ti.com, © Texas Instrument.