THE DAC RESOLUTION ENHANCEMENT FOR DC APPLICATIONS BY USING MULTIPLE PWMS ON ATMEGA328

NUT NEAMNARK

Department of Electrical Engineering, Dhurakij Pundit University, Bangkok, Thailand
Email: naaikido@hotmail.com

Abstract: To gain higher analog output levels from digital systems, a high resolution DAC is required, which is usually a cost tradeoff. This research demonstrates an affordable way to increase the DAC resolution with a low cost microcontroller. It works by merging 3 PWM signals which are generated by an ATMEGA328 microcontroller in order to be used as an input into an RC digital to analog voltage converter. The merging is completed by using digital circuit, which produces a new carrier frequency by reducing the original PWM carrier frequency by a half. The new carrier frequency is merged with 2 PWM signals by logical operation. The theories and experiment results reveal that converting the united PWM signal to analog voltages is possible. The conversion outcome is the improvement in analog voltage output resolution when compared to the conversion from a single PWM.

Keywords: ATMEGA328, Microcontroller, Digital to Analog Converter, DAC, PWM, Resolution

I. INTRODUCTION

PWM signals play an important role in control systems and commonly used in several control systems such as power electronics [1], speed controllers [2], and digital to analog voltage converters [3]. In digital to analog conversion, the resulting analog signal is mainly used in applications such as digital audio devices [4] and in data communication between controllers and measurement devices. One of a low complexity conventional conversions is, delivering a single PWM signal through a frequency filter which cut out a certain PWM carrier frequency. The outcome signal is analog voltage whose value depends on the PWM duty cycle value.

The ATMEGA328 microcontroller is capable of generating several PWM signals concurrently [5], which contain the same carrier frequency and independently specified duty cycle value. The value can be adjusted from 0-100 percent by specifying a particular value in a timer register, where the numbers are limited by the size of the built-in microcontroller's timer. Therefore, when the PWM signal is converted into an analog voltage signal, the output analog voltage level, which is known as resolution, is also limited.

A higher analog voltage control resolution means greater controller ability for voltage adjustment in a narrower gap. This adjustment ability clearly impacts the efficiency of a particular signal generation [6] and also affects the quantity of data represented by analog voltage levels. There are numerous ways to improve the digital-to-analog resolution which was introduced such as the combined Sigma-Delta and PWM interpolation technique [7] which gives a high performance result and low analog complexity. However, the overall implementation of this technique still requires many connections and elements and also some comprehensive mathematics. Another example is Dithered Output DAC which its theory is quick and periodic writing two or more different digital values to a DAC and filters the DAC output, and then the DAC output value becomes the average of the digital values written to the DAC [8]. The latter concept is simple but may make greater demands on a controller's computation time and resources.

To achieve higher resolution without running complex software computations and costly hardware units, this research offers a new approach by developing a few digital components to merge 3 generated PWM signals which come from a single microcontroller, ATMEGA328, so as to extend the duty cycle step numbers and diminish the intervals of adjacent analog voltage output values from the frequency filter.

II. BACKGROUND AND THEORIES

2.1. PWM

A PWM or Pulse Width Modulation signal is introduced as a constant frequency square wave. This frequency is called a carrier frequency. The PWM signal has 2 constant amplitude levels which is either V_{on} or V_{off} as described in Fig. 1.

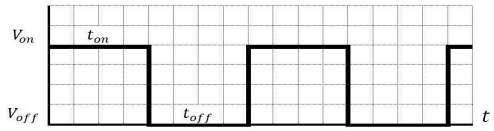


Fig. 1. The PWM signal with 50% duty cycle

The t_{on} and t_{off} are time durations while the amplitudes are V_{on} and V_{off} respectively. The percentage ratio of t_{on} and the period is called the Duty Cycle which is written as (1)

$$Dutycycle = \frac{t_{on}}{t_{on} + t_{off}} \times 100 = \frac{1}{period} \times 100 \quad (1)$$

The PWM signal construction can be achieved by using an analog circuit which makes a comparison between a saw tooth signal and a reference signal. The time during the saw tooth signal amplitude is greater than the reference signal amplitude is t_{on}, and is t_{off} when they are alternate. The PWM carrier frequency is the saw tooth frequency [9]. Another construction method is done by using a digital system circuit. It is the programming of a microcontroller to time a built-in timer to create t_{on} and t_{off} duration. Currently, microcontrollers are low cost and many are built in the form of application boards or learning kits which make them simple to understand and develop, for instance, an Arduino which makes things easier due to the simplified version of C++ programming and the availability of Arduino microcontroller kits [10] such as ATMEGA328 application boards. Therefore, using microcontrollers to construct PWM signals has become popular and has been widely implemented since it allows programmers to correct the PWM signal properties by modifying a microcontroller's source code. This is much simpler and takes less time than adjustment of the analog circuit PWM construction method as PWM properties are adjusted by rewiring the circuit and/or modifying the circuit devices.

2.2. DAC

A DAC (Digital to Analog Converter) is a device which converts digital values to analog values. The conversion method can be in several ways. The simple one uses a RC frequency filter circuit which requires non-complex elements and connections. It is a resistor series with a capacitor. This circuit filters the input PWM signal to eliminate the carrier frequency which is above a frequency cut off (f_{cutoff}) and lets the lower frequency elements pass out, which are then recognized as the analog voltage output. The conversion process is shown in Fig. 2 [11].

The f_{cutoff} value in Hertz is determined by (2), where R is the resistance value in Ohm and C is the capacitive value in Farad. Appropriate value selection will result in direct variation between the output analog voltage value and the input PWM duty cycle value. Because a change of the PWM duty cycle value causes the level change in analog voltage, therefore both of their resolutions are on the same page.

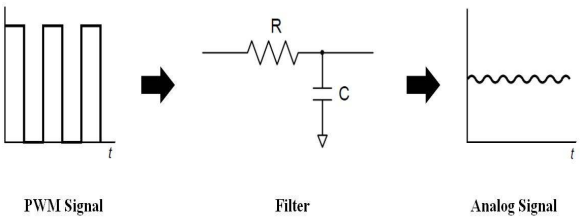


Fig. 2. The PWM to analog conversion by a RC frequency filter

$$f_{cutoff} = \frac{1}{2\pi RC} \qquad (2)$$

2.3. Microcontroller

This small electronic device functions as a computer. It has a digital control ability which is extensively used nowadays. A microcontroller structure is similar to a computer, and comprises processor units, memory units and input/output units. A processor executes a source code stored in the memory in order to process the input/output signals. This computability makes the microcontroller the heart of digital control and they are utilized in various project sizes. One of the microcontroller's roles is to produce a PWM signal to use as a control signal to control a device. This is achieved by coding a time delay function by using a built-in Interrupt Timer to be a time-based calculator. This method makes the microcontroller lose time in the major programming code execution as obviously seen in [12] that creating a function to generate a PWM signal consumes more resources and processing time.

Today, several microcontrollers are capable of producing a PWM signal with an independent adjustable duty cycle from the main executed programming code. This is achieved by a support of microcontroller internal hardware which results in processing time reduction.

The ATMEGA328 is one which produces 6 PWM signals simultaneously and independently. This makes it able to control various control devices at the same instant, which is evident from [13]. All 6 PWM signals have a carrier frequency controlled by an oscillator. Each one has distinct duty cycle values which ride on the 8-bit timer register values. Therefore, coping with the ATMEGA328 PWM duty cycles is just individually specifying integer numbers between 0-255 inclusive into those timer registers which is effortlessly done by using the *analogWrite( )* function in an Arduino IDE [14].

III. THE PWM MERGING CONCEPT

The analog voltage level from the frequency filter is up to the PWM duty cycle level generated from ATMEGA328. The original duty cycle levels can be different within 256 steps. Therefore, the analog voltage levels are within 256 steps either. These 256 steps may not be sufficient to provide good controllability for some devices. Expanding these steps with a simple way comes up by realizing that in some applications, they do not require many PWM channels to be operated so some are vacant as seen in Fig. 3.

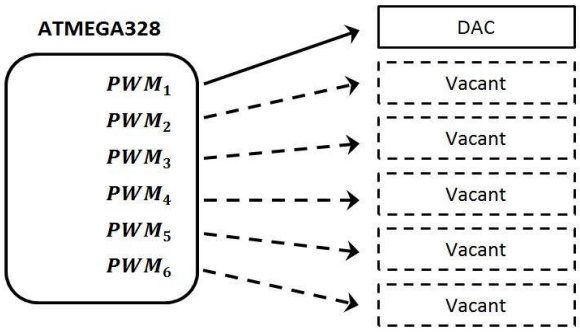


Fig. 3. Six PWM signals produced from a single ATMEGA328; one is occupied, the rest are available

An idea is making the most of the remaining PWM vacant channels by taking two more PWM channels, PWM_2 and PWM_3, joining them with PWM_1 to construct PWM_Z which increases the number of duty cycle setting steps. This new process is described in Fig. 4. It contributes the PWM_3 as a frequency divider input frequency, which is then divided by half to be a new carrier frequency Q and its inverse $\overline{Q}$. To form PWM_Z, the PWM_1, PWM_2, Q and $\overline{Q}$ are united by logical operation. The resulting PWM_Z is converted to an analog voltage signal by a RC frequency filter.

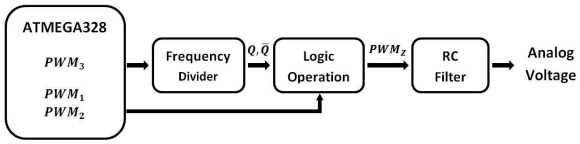


Fig. 4. Block Diagram of PWM to analog converting process

The frequency divider circuit is generally implemented by a D-Flip Flop [15]. The Boolean expression of the logical operation which forms PWM_Z is described in (3). It sums two comparison results together. The first one is between PWM_1 and Q, and the second one is between PWM_2 and $\overline{Q}$. The signal timing relation of (3) is shown in Fig. 5.

$$PWM_Z = PWM_1 Q + PWM_2 Q \qquad (3)$$

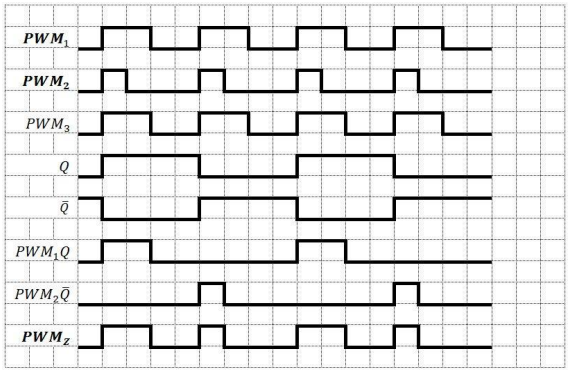


Fig. 5. Signal timing relationships according to (3)

Deriving (3) by applying the DeMorgan's Theorem [16] will obtain its multiplication form, that is (4). The frequency divider and the signal multiplication forms in (4) are schematically connected by the digital logic gates, shown in Fig. 6.

$$PWM_Z = \overline{\overline{PWM_1 Q} \cdot \overline{PWM_2 \overline{Q}}} \qquad (4)$$

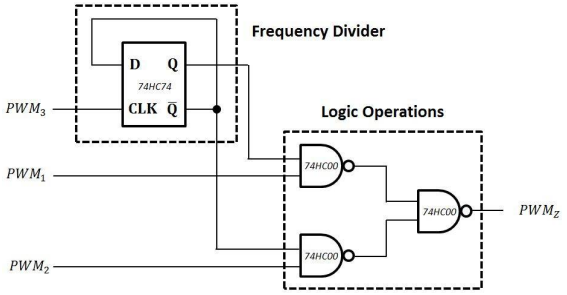


Fig. 6. The digital logic diagram of (4) to unite 3 PWM signals

IV. EXPERIMENTAL SETUP

All implemented circuit elements are supplied by the external voltage regulator. All VCC supplies are 4.968 V. The ATMEGA328 application board, shown in Fig. 7, reads the input signals from two delicate push buttons connected to the D12 and D13 channels to recognize the increasing or decreasing PWM_Z duty cycle specification. The push button BT_{up} is for an increasing signal and BT_{dn} is for a decreasing signal. To form the PWM shape of PWM_Z, the programming algorithm in Fig. 8 is developed and is running inside the ATMEGA328. The algorithm interprets the recognized PWM_Z duty cycle value to the particular PWM_1, PWM_2 and PWM_3 duty cycle value. These values are transferred to a PC for monitoring and are used inside the ATMEGA328 for PWM production. The three generated PWM signals are transferred to the next implemented circuits via the D9 D10 and D11 channels. The existing frequency divider circuit and the logic gate operation are built by a single 74HC74 – Dual D-Flip Flop and a single 74HC00 – Quad 2-Input NAND Gate. The wiring between the elements is shown in Fig. 9. Since the PWM frequency input to the divider circuit is 490 Hz, the new output from the divider circuit is 245 Hz. It is united with PWM_1 and PWM_2 to be PWM_Z by logic operation. Three PWM signals are individually transferred to each RC frequency filter circuit which is implemented as shown in Fig. 10. To achieve the f_{cutoff} value lower than 0.1 times of the carrier frequency value, the resistor value 82 kΩ and the capacitor value 100 nF are in consideration. Substituting them into (2), obtains the $f_{cutoff} = 1/\left(2\pi \times 82 \times 10^3 \times 100 \times 10^{-9}\right) = 19.4\text{Hz}$. At this point, the PWM_1, PWM_2 and PWM_Z are

converted to analog voltage by those filter circuits with the calculated frequency cut off. The completed circuit for the experiment is implemented on the breadboard as shown in Fig. 11.

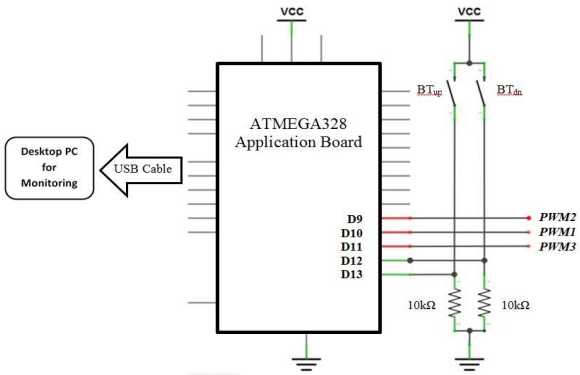


Fig. 7. The ATMEGA328 application board is connected to 2 push buttons with 10kΩ pull-down resistors. The data communication to PC takes place via USB cable

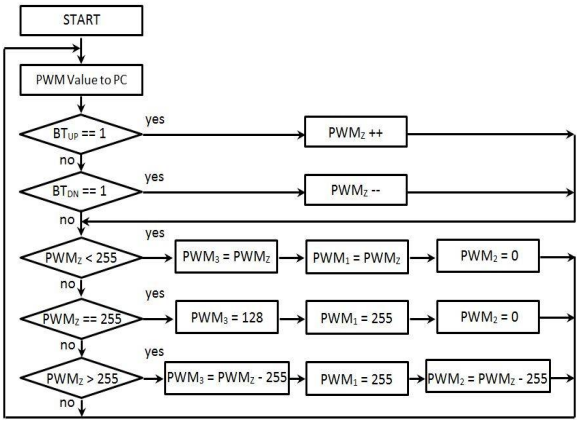


Fig. 8. The ATMEGA328 running algorithm for interpreting the PWM_z duty cycle value to the PWM_1, PWM_2 and PWM_3 duty cycle values

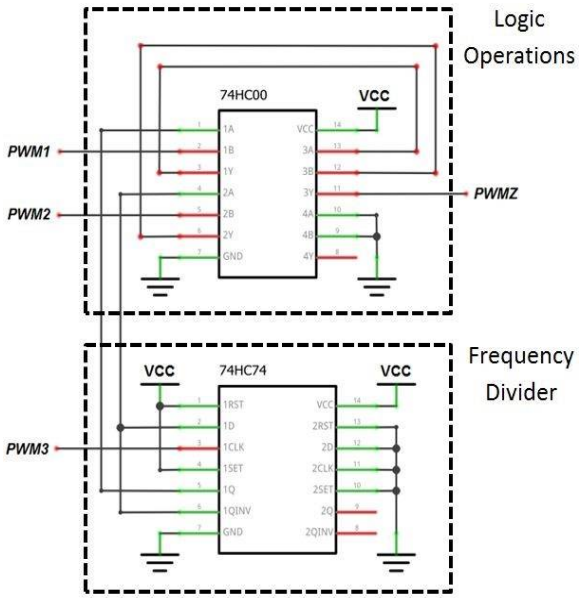


Fig. 9. Implemented circuit of Fig. 6. One 74HC74 is used in the frequency divider and one 74HC00 is used in the logic operation

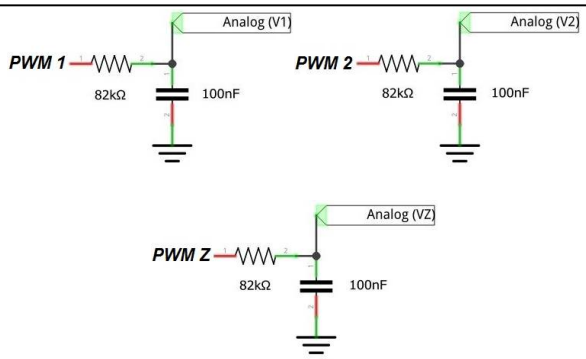


Fig. 10. The 3 identical DAC circuits or RC frequency filters, where each is implemented by one resistor and one capacitor. They convert PWM_1, PWM_2 and PWM_z signals to V_1, V_2 and V_z analog voltage signals respectively.

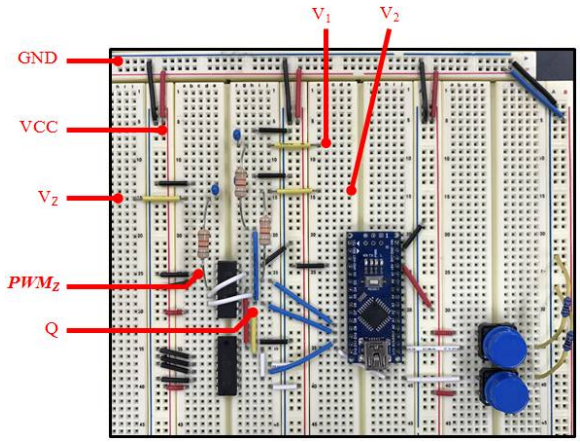
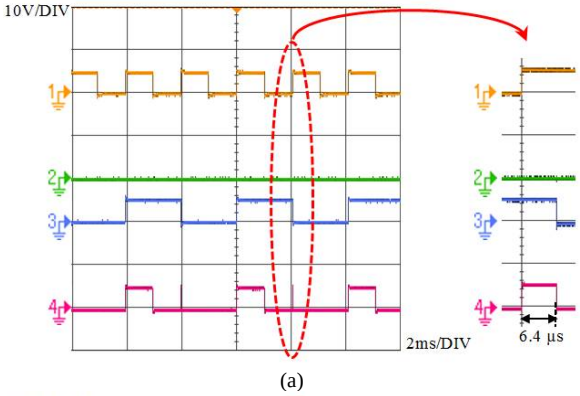
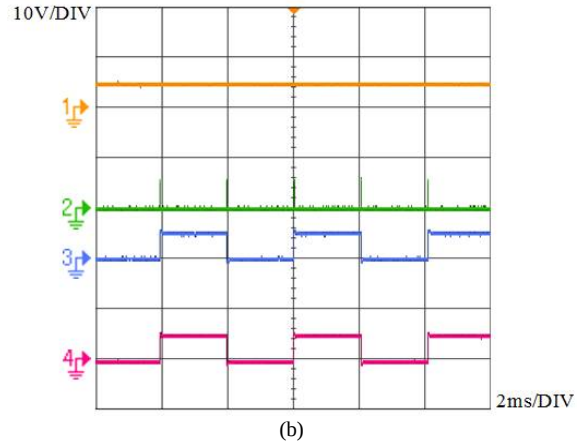


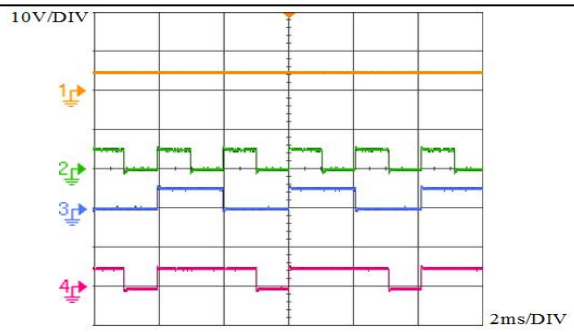
Fig. 11. The hardware implementation on a breadboard



(a)



(b)



(c)

Fig. 12. The oscilloscope channels 1 to 4 are connected to PWM_1, PWM_2, Q and PWM_Z in order respectively. The scales are set to 10V/DIV vertical and 2ms/DIV horizontal. In (a) The PWM_Z duty cycle value is 128, the algorithm creates the PWM_1 and the PWM_2 duty cycle values are 128 and 0 respectively In (b) the PWM_Z duty cycle value is 256, the algorithm creates the PWM_1 and the PWM_2 duty cycle values are 255 and 1 respectively. In (c) the PWM_Z duty cycle value is 383, the algorithm creates PWM_1 and the PWM_2 duty cycle values are 255 and 128 respectively.

V. RESULTS AND DISCUSSION

To examine the invented merging circuit operation and algorithm, the example of PWM_Z duty cycle values are set. They are within 3 intervals; less than 256, equal to 256 and greater than 256. Each setting duty cycle value creates its own individual PWM_1, PWM_2, Q and PWM_Z which are captured by an oscilloscope as shown in Fig. 12(a), (b) and (c). All signals act according to the merging concept. The PWM_Z is clearly the result of the combination of the 4 signals explained in (3). The small time delay between the output and input of 74HC74 makes produced Q shifts from both PWM_1 and PWM_2 for 6.4 µs. The effect of this signal shift is clearly seen when the PWM_Z duty cycle value is less than 256, where there is a spike that develops in every period as shown in the dash ellipse in Fig. 12(a). This spike makes the PWM_Z frame not look perfectly alike to a typical PWM but it eventually takes no effect to the analog voltage output value since it is eliminated by the serried RC frequency filter in Fig. 10.

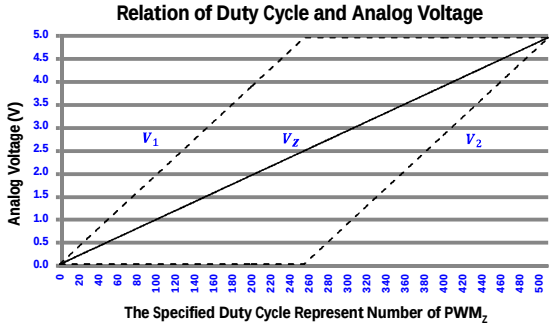


Fig. 13. The analog voltage V_1, V_2 and V_Z varies according to the PWM_Z duty cycle adjustment from integer number 0 to 510

In order to observe the variation of analog voltage values V_1, V_2 and V_Z, the PWM_Z duty cycle value is re-specified. Let it begin from integer number 0 (0% duty cycle) and increasing one step at a time to the final value at 510 (100% duty cycle). The resulting analog voltage from the RC frequency filters are measured by multi-meters as shown in Fig. 13. The solid line V_Z is gradually increasing along with the growth of the PWM_Z duty cycle value. It begins from 0 V minimum value to 4.936 V maximum value. The result reveals that the analog voltage values relate to the PWM_Z specified duty cycle value, they are direct variation and one to one function. The interval of analog voltage V_Z when the duty cycle value does one step change has an average at 0.010 V.

Both of PWM_1 and PWM_2 is also being observed during the experiment, their duty cycle varies according to the algorithm in Fig. 8. As they are increasing, the analog voltage V_1 and V_2, shown in the dash lines in Fig. 13, is gradually increasing from 0.003 V and reach 4.929 V maximum value. The relation between PWM_1 and V_1 is direct variation and one to one function as similar as the relation of PWM_Z and V_Z. This also applies to the PWM_2 and V_2 as well. The average interval of both analog voltage V_1 and analog voltage V_2, when the duty cycle has a one step change is 0.019 V which is nearly twice as much when compared to the interval of analog voltage V_Z.

The experiment measurement data demonstrates the minimum of V_Z is less than the minimum of both V_1 and V_2 for 0.003 V; the maximum of V_Z is greater than the maximum of both V_1 and V_2 for 0.007 V. These are because a small output voltage offset occurs from 74HC00. In the application, the minimum and the maximum output analog voltage values can be adjusted to desired values by applying an operation amplifier circuit to make correction.

The graphic comparison in Fig. 13 between the solid line of V_Z and the dash line of either V_1 or V_2 demonstrates the less steep V_Z is presented as the result from the decision algorithm in Fig. 8 and the implemented digital circuit in Fig. 9. This smaller slope of V_Z designates the output analog resolution is enhanced.

CONCLUSIONS

The analog voltage signal which comes from the conversion of united PWM signal (PWM_Z) from the implemented circuit in Fig. 9 is increasing in resolution when compared with the single one which comes from the conversion of the original PWM

signal from the ATMEGA328 microcontroller. The analog voltage resolution expands from the original 256 levels to 511 levels. The relation between the PWM_Z duty cycle and the analog voltage output is a direct variation and is a one-to-one function. When the PWM_Z duty circle rises, the analog voltage grows, and they behave the same when declining.

The PWM_Z is able to be constructed without running complex software computation. Only a decision algorithm in ATMEGA328 microcontroller is required as shown in Fig. 8. It operates in company with the low cost digital circuit implemented from a few components according to Fig. 11. However, the constructed PWM_Z shape is not identical to the theoretical PWM shape due to the occurrence of a spike. Therefore, if this signal is used to drive any electronic element, the spike may impact the element's efficiency.

REFERENCES

[1] Donny Radianto, Gamal M. Dousoky, Masahito Shoyama, "Design and Implementation of Fast PWM Boost Converter based on Low Cost Microcontroller for Photovoltaic Systems," in *IECON2015*, Yokohama, 9-12 November 2015.

[2] I G. A. P. Raka Agung, S. Huda, I W. Arta Wijaya, "Speed Control for DC Motor with Pulse Width Modulation (PWM) Method Using Infrared Remote Control Based on ATmega16 Microcontroller," in *International Conference on Smart Green Technology in Electrical and Information Systems (ICSGTEIS)*, Bali, 5 - 7 November 2014, pp. 108-112.

[3] Feng Zhou, Wei Xiong, "Using PWM Output as a Digital-to-Analog Converter on DSP," in *International Conference on System Science, Engineering Design and Manufacturing Informatization*, 2010, pp. 279-281.

[4] M.B. Sandler, Digital-to-Analogue conversion using pulse width modulation," in *Electronics & Communication Engineering Journal*, DEC 1993, pp. 339-348.

[5] Atmel 8-Bit Microcontroller with 4/8/32 Kbytes In-System Programmable Flash, Atmel Corporation, 2015 Atmel Corporation. / Rev.: Atmel-8271J-AVR-ATmega48A/48PA/88A/88PA/168A/168PA/328/328P-Datasheet_11/2015.

[6] Marian Kampik, Analysis of the effect of DAC resolution on AC voltage generated by digitally synthesized source," in *IEEE Transactions on Instrumentation and Measurement, Vol. 63*, No. 5, May 2014, pp. 1235-1243.

[7] S. -D. Grigorescu, C. Iliescu, B. Pantelimon, "Sigma-Delta modulation and PWM performs high-resolution DAC for DC applications," in *Proceedings of 20th Biennial Conference on Precision Electromagnetic Measurements*, 1996, pp.45-46.

[8] M. Hastings, "PSoC 3 and PSoC 5LP getting more resolution from 8-bit DACs," Cypress Corporation, San Jose, CA, USA, Tech. Rep. AN64275, 2012.

[9] Report, accessed (2016, November 1.), What is PWM? Massachusetts Institute of Technology, Massachusetts, U.S.A., [Online]. Available: fab.cba.mit.edu/classes/961.04/topics/pwm.pdf

[10] Yusuf Abdullahi Badamasi, "The working principle of an Arduino," *11th International Conference on Electronics, Computer and Computation (ICECCO)*, 2014, pp. 1-4.

[11] David M. Alter, 2008 September, Using PWM Output as a Digital-to-Analog Converter on a TMS320F280x Digital Signal Controller. Texas Instruments, Dallas, Texas, U.S.A. [Online]. Available: www.ti.com/lit/an/spraa88a/spraa88a.pdf

[12] E. Ramalho and J. A. Siqueira Dias, "A differential digital to analog converter using a CMOS digital PWM," in *PROC. 22nd International Conference on Microelectronics (MIEL 2000) Vol 2*, NIS, Serbia, 14-17 May 2000, pp. 699-702.

[13] Rosidah Sam, Mohd Nor Md Tan, Mohamad Safari Ismail, "Quad-copter using ATmega328 microcontroller," in *International Conference on Electrical Machines and Systems (ICEMS)*, 2013, pp. 556-570.

[14] Brain Evans, *Beginning Arduino Programming*, Apress, 2011, Chapter 6, pp. 86.

[15] John F. Wakerly, *Digital Design Principles & Practices, 3rd ed*, Prentice Hall, 1999, Chapter 7, pp. 451.

[16] John F. Wakerly, *Digital Design Principles & Practices, 3rd ed*, Prentice Hall, 1999, Chapter 4, pp. 199.

★★★