Generation of Multiple Frequency Pulse Width Modulation Signals Using FPGA

توليد إشارات تعديل عرض النبض متعددة الترددات باستخدام مصفوفات البوابة القابلة للبرمجة

Samer Jaloudi
Assistant Professor/ Al Quds Open University/ Palestine
sgalodi@qou.edu

سامر جالودي
أستاذ مساعد / جامعة القدس المفتوحة/ فلسطين

Received: 23/02/2021, **Accepted**: 17/03/2021
DOI: 10.33977/2106-000-004-006
http://journals.qou.edu/index.php/PJTAS

تاريخ الاستلام: 2021/02/23، تاريخ القبول: 2021/03/17
E-ISSN: 2521-411X
P-ISSN: 2520-7431

ABSTRACT

Microcontrollers, such as the one used in Arduino, are low-cost, simple, and suitable for low frequency and speed applications. However, concurrent processing and concurrent generation of multiple signals are necessary in many applications such as automotive sector, broadcasting, consumer electronics, and industrial applications. Field Programmable Gate Arrays (FPGAs) can generate concurrent signals that are more accurate than signals generated by microcontrollers. Moreover, FPGAs have higher performance than microcontrollers do mainly in the high frequency applications. In this paper, multiple frequency pulse width modulation (PWM) signals are concurrently generated from an FPGA Cyclone IV core board. The frequency of the generated PWM signals, the duty cycle, and the bit resolution are reconfigurable via simple variables in the entity section. The solution consists of six PWM signals. The generated signals are employed in three practical scenarios. In the first scenario, three of the produced signals are used to control light emitting diodes (LEDs) and generate music tones, and alarms. In the second scenario, one of the produced PWM signals is used to generate a sine wave, and then smoothed via an add-on filter, which makes the develop system to work as a function generator. In the third scenario, one of the produced PWM signals is used to control the speed of a motor by changing the duty cycle percentage. In this context, the developed system works as an analogue to digital converter (DAC) that is able to control analogue systems from a digital environment. The proposed system is compact, low-cost, scalable, and generates accurate signals.

***Keywords**:* DAC, Motor Control, Duty Cycle, FPGA, LPF, PWM, Sine Wave Generation.

الملخص

تعتبر أجهزة التحكم الدقيقة، مثل تلك المستخدمة في متحكمات (Arduino)، منخفضة التكلفة وبسيطة، ومناسبة لتطبيقات السرعة والتردد المنخفض. ولكن، المعالجة المتزامنة والتوليد المتزامن لعدة إشارات ضروري في كثير من التطبيقات مثل قطاع السيارات، البثّ، إلكترونيات المستهلك، والتطبيقات الصناعية. يمكن لمصفوفات البوابة الميدانية القابلة للبرمجة (FPGAs) أن تنتج إشارات أكثر دقة من الإشارات التي يتم إنشاؤها بوساطة المتحكم الدقيق. علاوة على ذلك، تتمتع (FPGAs) بأداء أعلى من المتحكمات الدقيقة بشكل أساسي في التطبيقات عالية التردد. في هذا البحث، يتم إنشاء إشارات تعديل عرض النبضة متعددة الترددات (PWM) بشكل متزامن من اللوحة الأساسية (FPGA Cyclone IV). تردد إشارات (PWM) المتولدة، ودورة العمل، ودقة البت هي المتغيرات التي يمكن إعادة ضبطها من خلال متغيرات بسيطة في قسم الكيان (entity) يتكون الحل من ست إشارات (PWM). تم توظيف الإشارات التي أنتجت في ثلاثة سيناريوهات عملية. في السيناريو الأول، تم استخدام ثلاثة من إشارات (PWM) الناتجة للتحكم في الصمامات الثنائية الباعثة للضوء (LED)، وتوليد نغمات موسيقية وأصوات تحذيرية. في (السيناريو) الثاني، تم استخدام واحدة من الإشارات الناتجة لتوليد موجة جيبية، ثم رُشحت باستخدام مرشح مضاف إلى التصميم، مما يجعل النظام يعمل كمولّد إشارات. في السيناريو الثالث، تم استخدام إحدى إشارات (PWM) الناتجة للتحكم بسرعة محرك من خلال تغيير النسبة المئوية لدورة العمل. في هذا السياق، يعمل النظام المطور كمحول تناظري-رقمي (DAC) قادر على التحكم بالنظم التناظرية من بيئة رقمية. النظام المقترح صغير الحجم، ومنخفض التكلفة، وقابل للتوسعة، ويولد إشارات دقيقة.

الكلمات المفتاحية: محول رقمي-تناظري، تحكم بالمحرك، دورة العمل، مرشح تمرير ترددات منخفضة، توليد موجة جيبية.

INTRODUCTION

Pulse width modulation (PWM) is a technique used for control in a wide range of applications, such as control of power converters, control of motor speed, control of light emitting diodes (LEDs), and control of video displays. In addition, it can be used as counters, and for generating music tones. The PWM techniques are widely used for switching power converters, mainly in AC-to-DC rectifiers (Yoo et al., 2007), DC-to-AC inverters (Amorndechaphon, 2016), and power converters (Singh et al., 2018). Moreover, PWM signals are used for power control of mobile phones (Karthikeyan et al., 2011), sound applications (Estes et al., 2005), and audio applications (Liang et al., 2011). Furthermore, they are used for LED drivers (Abdelmessih et al., 2016), for LED video display application (Svilainis, 2012), for mobile LCDs (Park et al., 2014), and for wave generation (Sreekanth & Moni, 2013).

Figure 1 shows typical PWM signals with different duty cycles. The duty cycle determines the “ON” time of the PWM signal, which varies the width of the pulse according to a pre-defined value. If applied to LEDs, a duty cycle of 10% means 10% of the time-period, the light will be “ON”, and a duty cycle of 30% means 30% of the time-period, the light will be “ON”, etc.

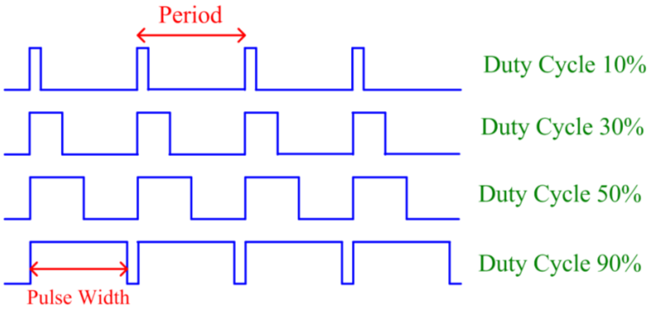


Figure 1. PWM signals with different duty cycles

Equation (1) demonstrates the relationship between the duty cycle, the pulse width, and the period.

$$Duty\_Cycle = \frac{Pulse\_Width}{Period} \times 100\% \quad \ldots\ldots (1)$$

Depending on the duty cycle, control signals generated from the PWM can be applied in different applications, for example, to control the speed and direction of motors, the light intensity of LEDs, etc. However, in some applications, the required frequency is high that the resolution obtained with classical techniques is not enough (de Castro et al., 2007). These days, FPGAs are small size and not expensive. The low-cost Cyclone® II FPGA board is ideal for PWM signal generation, with low-power consumption, and high performance. Moreover, the FPGA-based PWM has a high-switching speed and accurate signals, which give the PWM signals the property of precise control. Therefore, this paper proposes a generation of multiple-frequency PWM signals based on the resources available in the FPGA Cyclone IV core board. Thus, control PWM signals are produced and used in three practical scenarios, to generate audio tones and control LEDs, to generate a sine wave, and to control a motor speed. Concurrent processing and concurrent generation of multiple signals are necessary in many applications such as automotive sector, broadcasting, consumer electronics, and industrial applications. The proposed solution is low-cost, reliable, and compact. Moreover, it provides accurate PWM signals, good performance and control characteristics. Other benefits inherited from the use of FPGAs Cyclone IV include system integration, design reuse, low-power consumption, and scalability (Intel FPGAs, 2021).

The remainder of the paper is organized as follows. Section II introduces a literature review. Section III introduces the FPGA platform used in this paper, and the developed environment. While section IV introduces the results and three practical scenarios, section V concludes the paper.

LITERATURE REVIEW

The speed control of motors can be implemented using a driver circuit based on a microprocessor, a personal computer, a microcontroller, a programmable logic controller (PLC), or an FPGA. Researchers (Payak and Kumbhar, 2015) proposed a motor speed-control based on a simple PWM technique produced by an FPGA board. The FPGA signals have the control on the pulse width, which is used for firing an inverter. The inverter supplies the power to the driver circuit as per variation of the PWM. The control is accurate and fast, and the design is compact and low-cost.

Researchers (Kodama and Koutaki, 2019) described the fabrication and control of a robot that plays an acoustic guitar. The robot performs linear movement of solenoids and performs the fretting and picking operations in playing guitars. The strength and weakness of the sound is expressed by adjusting the operation of the solenoid using PWM control.

The research of (Mondal and Sharma, 2019) implemented an FPGA-based low power music system and digital data transmission environment. The output is taken from the headphone jack using PWM, and the music system provides an excellent music data to the audio output pin in real-time.

In order to solve the limitations and problems of a local music instrument, researchers (Saragih et al., 2020) developed a smart system that can automatically play the instruments notes. The instrument is driven by a DC motor, of which its speed is regulated using a PWM pin of an Arduino-based microcontroller. The Arduino program is remotely operated through the commands of an android application on a smart phone, via Bluetooth. The developed instrument has a frequency error of 0.49%, a vibration error of

3%, and a vibration error controlled by the PWM of 1.15%.

A monolithic controller for pulse width modulated (PWM) DC-to-DC converter is presented in the work of (Lu and Wu, 2009) for a LED driver circuit. For LED lighting, both the digital and analog dimming modules were integrated onto a chip, which were used to meet the demands of the two kinds of dimming applications respectively.

The paper of (Jung et al., 2010) presents a new dimmer using two active switches for AC LED lamps. The control method of the proposed dimmer is based on the pulse width. Compared to the conventional phase-controlled dimmer, the proposed PWM dimmer produces sine wave without harmonics problem. Furthermore, the proposed control method does not amplify the light flicker due to independence of the input voltage. The experimental result shows that the proposed PWM dimmer has good performance.

The work of (Martins et al., 2017) presents a real-time data monitoring to reduce automatically the brightness depending on the frames displayed on a large-scale LED panel. It employs a PWM algorithm implemented into a FPGA that takes into account the inferred power and if it exceeds a predetermined value, it dynamically reduces the power preventing overheating of drivers and connections.

The relationship between music rhythms and the output colors of red-green-blue-light-emitting-diode (RGB-LED) lamp is studied in the work of (Guo et al., 2019). Colors of the RGB-LED lamp are controlled by music rhythms. The study aims at solving the problem of matching the music rhythms and the light colors. The colors of the tricolor RGB are controlled by changing the ratio of pulse width modulation (PWM) generated from an STM32 single-chip microcomputer. The experimental results show an improvement in the control of the light colors by the music rhythms.

Multichannel PWM generator for control of LED brightness in automotive applications is presented in the work of (Wojtkowski, 2018). In some automotive applications, there is a need of multichannel control of many LED entities, which is necessary for independent control of each lighting channel. Unlike typical hardware solutions, the presented generator is based on microcontroller-based software. The solution is low-cost and easy to implement.

The research work of (Sreekanth and Moni, 2013) presents a reprogrammable architecture to implement the amplitude modulated triangular carrier pulse width modulation (AMTC-PWM) with perfect reproduction capability. The AMTC-PWM method is a natural sampled PWM method, which can extend the linearity of the sinusoidal PWM. Thus, it eliminates the need of over modulation in the pulse-dropping region to reach the square wave boundary. The architecture is implemented using the FPGA Spartan 6 family device LX45 from Xilinx.

SYSTEM ARCHITECTURE

The objective of this paper is to design and develop hardware environment for the generation of multiple-frequency PWM signals using an FPGA development board. FPGAs excel in concurrent processing, and hence, concurrent multiple PWM signals are generated using reconfigurable variables that represent frequencies and duty cycles. The first subsection describes the FPGA development board from WaveShare that hosts the Cyclone IV, and the second subsection describes the designed and developed environment.

The FPGA Development Board

This subsection describes the FPGA development board used in this work, which is the OpenEP4CE10-C from WaveShare. It is an FPGA development board that consists of the motherboard DVK600 and the FPGA core board EP4CE10F17C8N, which is the Cyclone IV. The board contains other accessories required for the operation of the core board and for basic testing, as shown in Figure 2. It has an FPGA core board connector for easily connecting core boards. Moreover, it has eight, sixteen, and thirty-two inputs and outputs (8I/Os_1, 8I/Os_2, 16I/Os_1, 16I/Os_2, 32I/Os_1, 32I/Os_2, 32I/Os_3) interfaces for connecting accessory boards and modules. All the I/O interfaces are capable of being operated as universal synchronous/asynchronous receiver/transmitter (USART) protocol, inter IC communications (I2C) protocol, and serial peripheral interface (SPI) protocol; and capable of driving devices such as USB and Ethernet. Furthermore, it has an SDRAM interface for connecting SDRAM accessory board, an LCD interface for connecting an LCD1602, a ONE-WIRE protocol interface that connects to the temperature sensor DS18B20, a 5V DC jack, a joystick, a buzzer for testing audio tones, a potentiometer for the LCD1602 contrast adjustment, and a power switch.

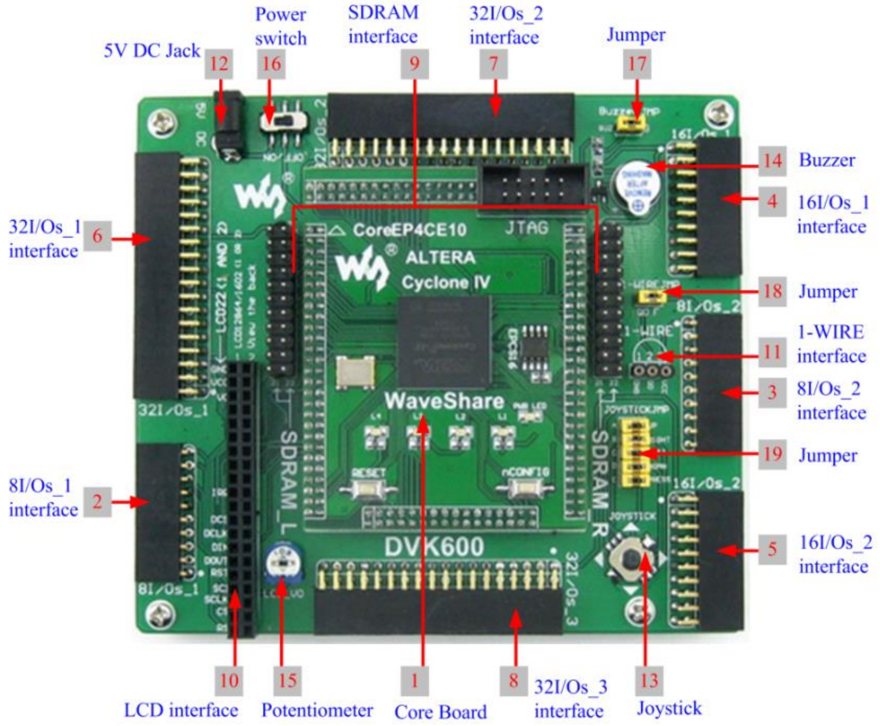


Figure 2. The FPGA development board from WaveShare for FPGA Cyclone IV

As shown in Figure 2, the EP4CE10F17C8 core board consists of voltage regulators, EPCS16 onboard serial FLASH memory for storing code, power indicator, four LEDs for testing, reset button, nCONFIG button for re-configuring the FPGA chip, 50MHz active crystal oscillator for clocking, JTAG interface for programming, and the Cyclone IV FPGA device with the features shown in Table 1.

Table 1 Features of Cyclone IV core EP4CE10F17C8

Feature	Value
System Frequency	50MHz
Core Voltage	1.2V
Inputs/Outputs	180
Logical Elements (LEs)	10320
RAM	414kB
Phase-Locked Loops (PLLs)	2
Programming Interface	JTAG

The next subsection describes the environment design in more details using the resources and features of Cyclone IV, mentioned above.

Environment Design and Development

This subsection describes the design of a hardware module, which is written in VHDL for the FPGA Cyclone IV, and generates multiple-frequency (PWM) signals. The generated PWM signals, come out of the module, are based on the duty cycle. The value of the duty cycle is set by the user in the definitions section of the entity as a generic value. Other configurable values that are set as generics by the user in the definitions section include frequencies of the generated PWM signals, bit resolution, and the system frequency, which is 50MHz in the case of the EP4CE10 core. In more details, the duty cycle and the frequencies are reconfigurable by changing the variables values in the entity section of the VHDL program.

When the module is set to generate multiple frequencies concurrently, one PWM signal is generated per frequency. In this paper, the number of frequencies is set to three, and hence the module generates three PWM signals, in addition to another three that are generated from the inverse states of the original signals. In total, six signals are produced concurrently, and the processing is running concurrently as well via three separate processes.

A schematic diagram of the developed module, which generates multiple-frequency PWM signals, is shown in Figure 3, based on the

FPGA Cyclone IV. The input is the System_Clock, which is 50MHz. The outputs, which are Cyclone IV output pins, are PWMout[1..3] and PWMinverse[1..3] pins. These signals control external LEDs and a buzzer. The buzzer produces alarm sounds or music tones, depending on the frequency defined by the user as a generic value in the entity section. The input frequencies and duty cycles are 440 Hz with 12.5% duty cycle, 3Hz with 25% duty cycle, and 16 kHz with 50% duty cycle, respectively. These output signals are employed in three practical scenarios. The 440 Hz tone is routed to the buzzer pin and a LED, the 3Hz signal controls two LEDs, and the 16 kHz sine wave signal is routed to an add-on RC low pass filter (LPF) circuit for smoothing. The LPF resistor and capacitor values are 25Ω and 0.47μF respectively. The third scenario employs one of the signals to control a motor speed.

The signals in the register transfer level (RTL) are wires that connect the process variables to the output pins of the FPGA board. Each process produces two PWM signals based on counters that are triggered via the system clock. Each process calculates a counter value that is compatible with its duty cycle specified in the entity section of the VHDL program. Each clock cycle the counter increases by one and an "if" statement checks whether the intended value is reached according to the duty cycle, otherwise the counter continues counting to produce the logic one, which is interpreted as high at the FPGA pin. If the counter reaches the intended value according to the duty cycle, it produces logic zero, which is interpreted as low at the FPGA pin. However, the opposite values are assigned for the inverse signals (PWMinverse). Therefore, for LED2 and LED4 that flashes three times per second, when LED2 is "ON" LED4 is "OFF" and when LED2 is "OFF" LED4 is "ON".

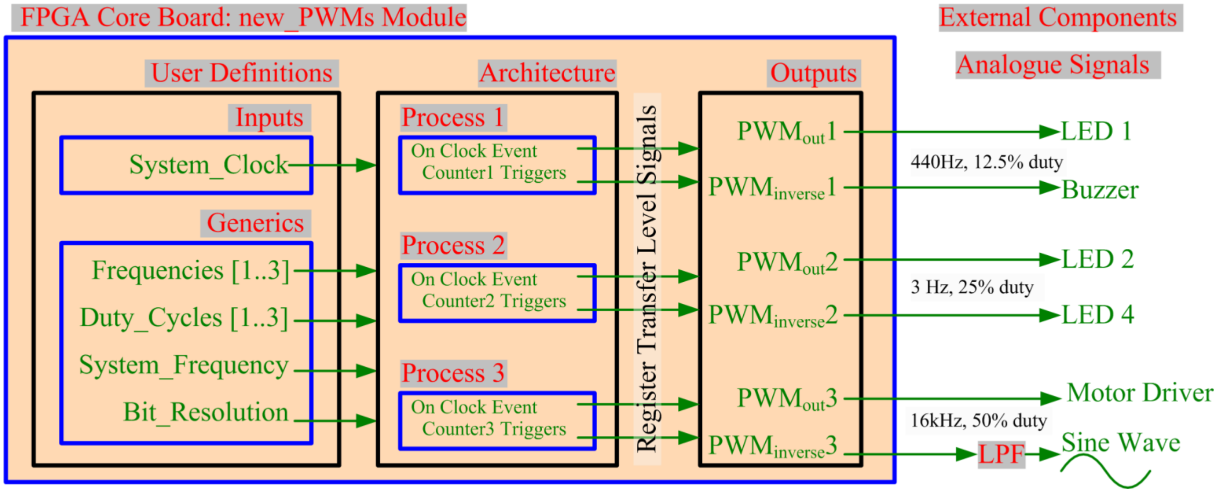


Figure 3. The developed environment

As appear in Figure 3, the environment consists of two parts, a digital part represented via FPGA, and an analogue part represented via external components. The digital part is developed using VHDL, and resides inside the FPGA core. The analogue part represents the external components, and resides on the motherboard DVK600. However, the motherboard does not have a built-in LPF. Hence, an external filter is designed and built to obtain a sine wave from the sixth PWM signal. The details of the filter design and analysis are demonstrated in the next section.

RESULTS AND PRACTICAL SCENARIOS

The integrated development environment of the FPGA Cyclone IV is the Quartus II, version 13.0, web edition. Table 2 shows the result that is taken from the Quartus II after compilation, which demonstrates a successful implementation and synthesis of the developed module on the EP4CE10F17C8 core.

For the results illustrated in table 2, the input frequencies and duty cycles are 440Hz with 12.5% duty cycle, 3Hz with 25% duty cycle and 16 kHz with 50% duty cycle, respectively. The total logic elements (LEs) are 135, and the total registers are 61. This is an indication about the developed environment, which is compact, reliable, and low-cost. Since the digital circuit produced by the RTL viewer is very huge, it is not included in the paper.

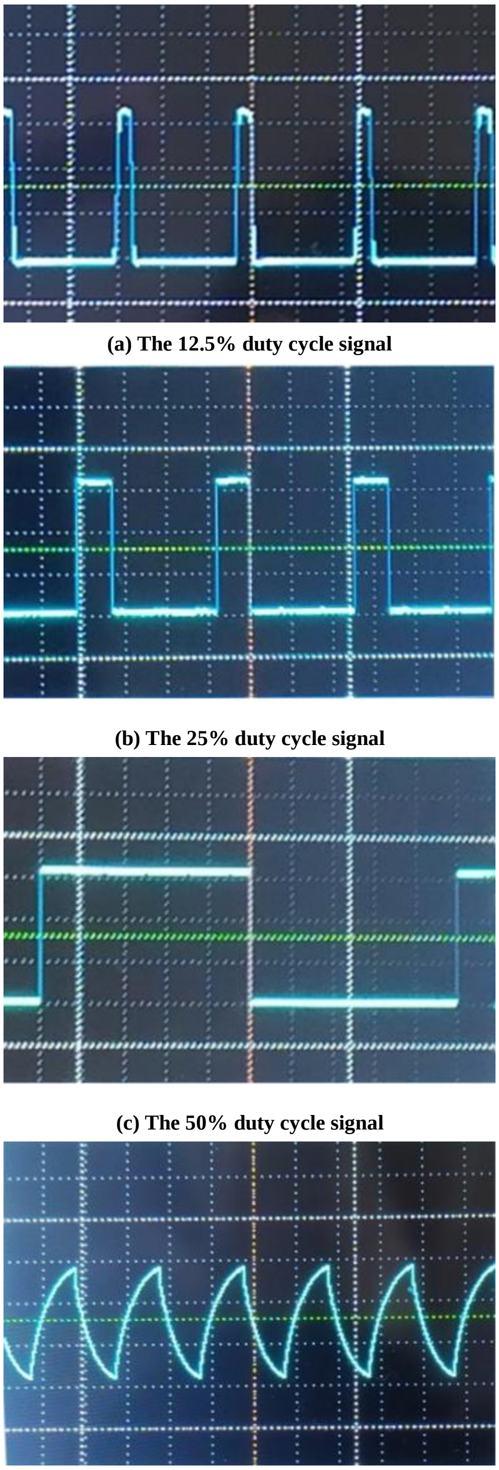
Table 2 Synthesis of the Developed Module on the FPGA

Feature	Value
Flow Status	Successful - Fri Mar 12 16:37:30 2021
Quartus II 32-bit Version	13.0.0 Build 156 04/24/2013 SJ Web Edition
Top-level Entity Name	new_PWMs
Family	Cyclone IV E
Device	EP4CE10F17C8
Total logic elements	135 / 10,320 (1 %)
Total combinational functions	134 / 10,320 (1 %)
Dedicated logic registers	61 / 10,320 (< 1 %)
Total registers	61
Total pins	7 / 180 (4 %)
Total virtual pins	0
Total memory bits	0 / 423,936 (0 %)
Embedded Multiplier 9-bit elements	0 / 46 (0 %)
Total PLLs	0 / 2 (0 %)

Using the DSO201 nano oscilloscope, snapshots of the obtained signals are taken from the screen of the pocketsize oscilloscope. These snapshots are illustrated in Figure 4 with the generated pulses depending on the specified duty cycles. However, the sketches produced by the nano oscilloscope are estimated based on the oscilloscope circuits, and does not reflect the true values produced by the FPGA. The pocketsize oscilloscope is basic oscilloscope, which gives an indication about the measured signals, and therefore these results are indicative.

Concurrent processing and concurrent generation of multiple signals are necessary in many applications such as automotive, broadcasting, consumer electronics, and industrial applications. The generated PWM signals are tested in three practical scenarios.

The first scenario is concerned with the concurrent generation of alarms and flashers by the buzzer and three LEDS. The LED number 1, LED number 2, and LED number 4 are working concurrently, producing alarms in a specific sequence described above. The buzzer produces a continuous alarm using the "A" tone, which has a frequency of 440 Hz. Such a practical scenario is widely used not only for alarms in industrial applications but also in automotive.



(a) The 12.5% duty cycle signal

(b) The 25% duty cycle signal

(c) The 50% duty cycle signal

(d) Filtered PWM signal

Figure 4. Snapshots taken from the screen of a pocket oscilloscope for each generated PWM signal.

The sixth PWM signal has a 50% duty cycle, and a frequency of 16 kHz. It is injected in an external, add-on, simple RC LPF, which smooths the sine wave. However, the output was not very smooth, as shown in Figure 4 (d). Therefore, in the second scenario, another filter is designed using PSpice software, student version. The filter is fourth order, Butterworth LPF, with 16 kHz cutoff frequency. As shown in Figure 5, the filter consists of two stages of the Sallen-Key configuration that employs an operational amplifier with a specific arrangement of resistances and capacitances; each stage is a second order Butterworth LPF, and the cutoff frequency for each stage is calculated using eq. (2) and eq. (3) respectively:

$$f_{C1} = \frac{1}{2\pi\sqrt{R_5C_3R_4C_4}} \qquad \dots (2)$$

$$f_{C2} = \frac{1}{2\pi\sqrt{R_3C_1R_2C_2}} \qquad \dots (3)$$

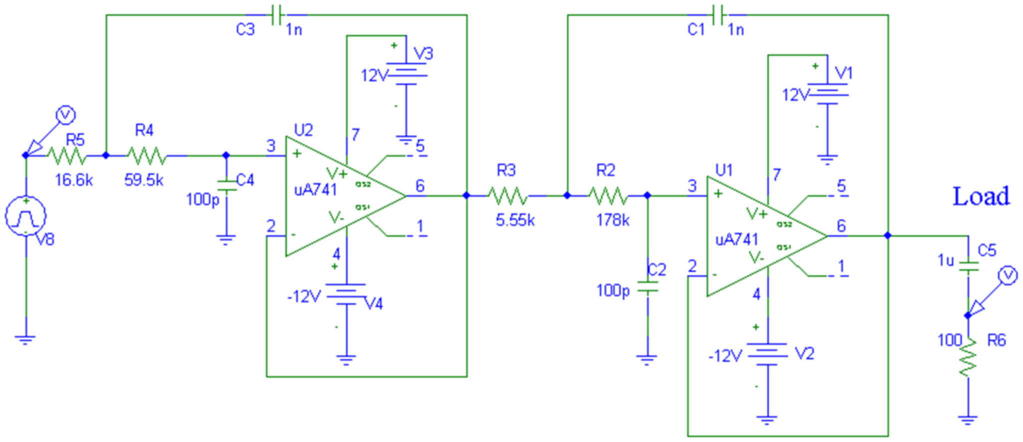


Figure 5. Fourth order, Butterworth LPF.

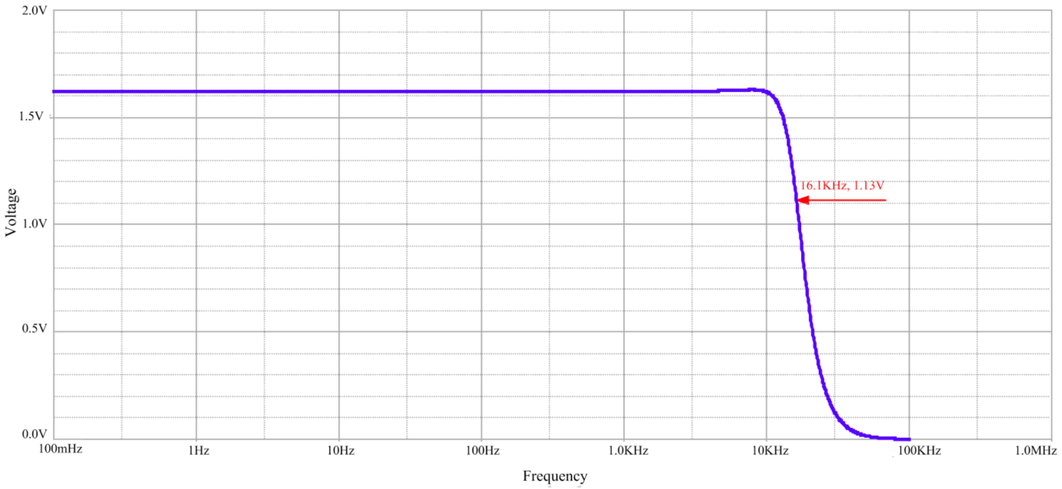


Figure 6. AC analysis of the fourth order, active, Butterworth LPF.

According to eq. (2), the cutoff frequency of the first stage is 16.01 kHz. According to eq. (3), the cutoff frequency of the second stage is 16.01 kHz as well. As shown in Figure 5, a capacitor of 1µF is used in series with a load of 100Ω in order to remove the DC component that may leak in the load resistor. The input frequency of the filter is 16 kHz, the duty cycle is 50%, and the maximum input voltage is 1.62V, which are the specifications of the FPGA pin that produces the sixth PWM signal. At the cutoff frequency, which is 16 kHz, the output voltage is 0.707 of the input voltage, or 1.14V. However, the result of the AC analysis shown in Figure 6 and made up to 100 kHz, estimates the cutoff frequency as 16.1 kHz and the output voltage as 1.13V. The AC-analysis is made using an AC sine-wave voltage source, and the transient analysis is made using a pulse-wave voltage source.

The sketch shown in Figure 7 demonstrates the transient analysis of the designed filter in the first 500 microseconds. The input signal is square wave (PWM of 50% duty cycle) with 62.5 µs period (16 kHz-frequency). The filter output is sine wave. The transient analysis shows a stable response of the filter in the third cycle. The input frequency is 16 kHz, and hence the period is 62.5 microseconds, which means that the stability of the filter starts to appear at 200 µs. The output is a sine wave that swings between -0.71V and +0.71V after stability, and without the DC component, which is removed by the 1µF capacitor. The developed system produces a sine wave, generated from the FPGA-based PWM signal, which makes the developed system to act as a function generator that produces PWM and sinusoidal waves.

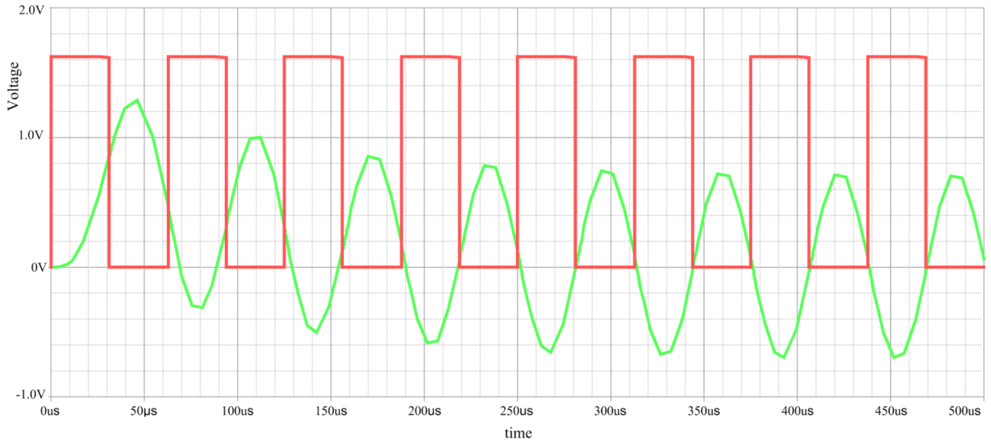


Figure 7. Transient analysis of the fourth order, active, Butterworth LPF.

These signals are in use for communications, electric circuits, consumer electronics, broadcasting, etc. In this context, the PWM signal converted the digital values, produced by the FPGA pin, into an analogue signal, and worked as a simple digital to analogue converter (DAC). In fact, the PWM is efficient to control analogue components and electronic circuits from digital FPGA modules.

The third scenario is concerned with the speed control of a DC motor using one of the generated PWM signals. DC motors are in use in many applications, include but not limited to lathe machines, centrifugal pumps, fans, blowers, conveyors, lifts, weaving machine, spinning machines, etc. (Barua and Abedin, 2018). There are many types of DC motors, which are chosen according to the intended application. Industrial applications require an accurate speed control for precise movement, in addition to avoid safety problems. In fact, FPGAs can generate accurate PWM signals required for motor control. Hence, a third scenario is proposed here, as shown in Figure 8 to control the speed of a DC motor by changing the duty cycle of the PWM signal.

A mini DC motor is used in this scenario, which operates on a voltage ranges from 0.5V to 6.0V. The motor, shown in Figure 8, is connected in series with the collector of a general-purpose transistor of type 2N2222A. In addition, a protective fast-switching diode is connected in parallel to handle spikes. A ceramic capacitor is connected in parallel to compensate for the inductive reactance caused by the coils of the motor at the selected frequency of the PWM signal, which is 16 KHz.

The audible noise (humming sound) generated by the DC motor while operating, is an effect of driving it with switched current produced by the PWM signal. The magnetic field causes the enclosure and coils to act as a speaker, and hence, generates the noise at the PWM frequency. In general, a PWM frequency over 16 kHz, which is out of the audible range, reduces the humming noise. However, choosing a PWM signal that has higher frequency, the inductance of the motor starts to have higher effects, the motor dissipates more heat, and hence more power losses. On the other hand, a PWM signal that has a PWM frequency in the audible range causes noisy humming sound.

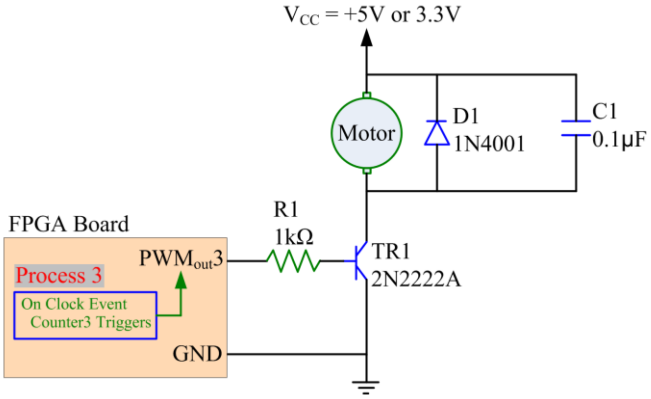


Figure 8. Circuit driver for the speed control of the DC motor

As mentioned above, the motor has coils, and hence it has an inductive reactance value that is influenced by the selected frequency (16 kHz) and the coils inductance (1 mH), as shown in eq (4):

$$X_L = \omega \cdot L = 2 \times \pi \times f \times L \qquad \ldots\ldots (4)$$

Accordingly, the inductive reactance is 100.5Ω. Together with the coils resistance, they form an RL filter, which will have a noticeable effect in high frequencies. Therefore, a capacitor of 0.1μF is added in parallel to the motor's terminals. The capacitor value is selected depending on eq. (5), which evaluates the capacitive reactance that must compensate for the inductive reactance:

$$X_C = \frac{1}{\omega \cdot C} = \frac{1}{2 \times \pi \times f \times C} \qquad \ldots\ldots (5)$$

The goal of driving a DC motor by a PWM signal with a certain duty cycle is to provide it with a portion of the full DC voltage, and hence speed control can be achieved by changing the duty cycle percentage.

Table 3 and table 4 illustrate this idea. The motor terminals voltages are measured according to the changes in the duty cycle that is adjusted by the user. Measurements are taken for transistor voltage (VCC) of 3.3V and 5.0V. The voltage at the FPGA pin is varied according to the duty cycle of the PWM signal.

Table 3 Measured motor terminals voltage and speed according to duty cycle changes at V_{CC}=3.3V

Duty Cycle (%)	PWM Pin Voltage	Terminals Voltage	Speed (rpm)
12.5	0.40	0.02	0
25.1	0.81	0.07	0
37.6	1.21	0.34	510
50.2	1.62	0.75	1125
56.5	1.83	0.98	1470
62.7	2.02	1.12	1680
75.0	2.42	1.50	2250
87.8	2.83	1.90	2850
94.1	3.04	2.22	3330
99.6	3.26	2.34	3510

Table 4 Measured motor terminals voltage and speed according to duty cycle changes at V_{CC}=5.0V

Duty Cycle (%)	PWM Pin Voltage	Terminals Voltage	Speed (rpm)
12.5	0.40	0.05	0
25.1	0.81	0.37	555
37.6	1.21	0.90	1350
50.2	1.62	1.50	2250
56.5	1.83	1.83	2745
62.7	2.02	2.08	3120
75.0	2.42	2.75	4125
87.8	2.83	3.51	5265
94.1	3.04	3.82	5730
99.6	3.26	4.10	6150

In ten experiments, ten different duty cycle values have been applied in the generic part of the entity section. Each time, the voltage at the FPGA pin (PWMout3 signal) is measured, and the voltage at the motor's terminals is measured as well. These results of the ten experiments are illustrated in table 3. The motor speed is calculated (in revolutions per minute, rpm) according to the measured voltage at the terminals of the motor. A typical speed of 9000 rpm is obtained when a voltage of 6.0V is applied across the motor's terminals. Thus, the speed is calculated according to eq. (6):

$$Speed = \frac{Terminal\_Voltage}{6.0} \times 9000 \qquad \ldots (6)$$

The maximum voltage at the FPGA pin is 3.3V when the duty cycle is 100% and 0.0V when the duty cycle is 0.0%. This voltage is applied to the base of the transistor, which controls the current flow between the collector and the emitter. The extra voltage that does not apply to the motor is consumed by the transistor as collector-to-emitter voltage (VCE) drop, which is necessary for the operation of the transistor itself. For example, when a duty cycle of 50% is applied to the PWM signal, a voltage of 1.62V is driven out of the FPGA pin to the transistor base, which switches the transistor "ON" and "OFF" 16000 times per second (16 kHz frequency). The measured voltage on the motor terminals is 1.5V, and hence the VCE is 3.5V since the VCC value is 5V. This is calculated via eq. (7):

$$V_{CC} = V_{Terminals} + V_{CE} \qquad \ldots (7)$$

These values are illustrated in table 3 and table 4.

The results of table 3 and table 4 are plotted in Figure 9. The relationship between the FPGA pin voltage and the specified duty cycle is plotted in the top-left of the sketch, which is linear relationship. The curve that represents the relationship between the measured voltage at the motor's terminals and the duty cycle is plotted in the top-right of the sketch. The duty cycle is taken in logarithmic scale to show percent changes. Moreover, the curve that represents the relationship between the motor speed and the specified duty cycle is plotted in the bottom-left of the sketch. The speed is in thousands, so the scale is logarithmic to respond to skewness towards large values. In addition, the relationship between the motor terminals voltage and the motor speed is plotted in the bottom-right of the sketch. The speed, which is located in the x-axis, is logarithmic scale as well for the same reason mentioned above. Four curves are plotted for two values of the transistor feed voltage (VCC), of 3.3V and 5.0V respectively. Both have almost the same response, and hence both lines are identical in the fourth plot.

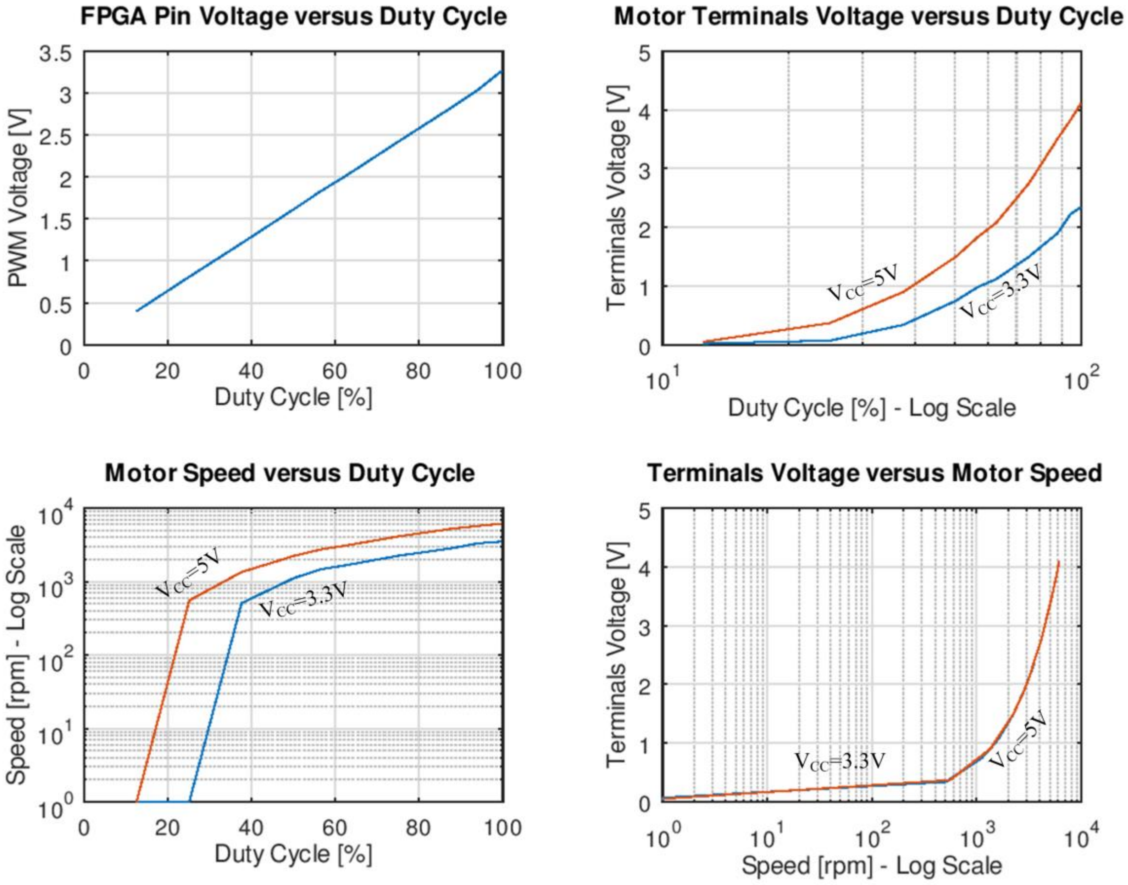


Figure 9: Measured motor terminals voltage, FPGA pin voltage, and speed according to duty cycle changes

CONCLUSION

The paper describes a successful implementation and synthesis of FPGA-based generation of concurrent multiple-frequency PWM signals to control LED flashers and motor speed, and to generate audio tones and sinusoidal wave. Many applications, such as automotive sector, consumer electronics, broadcasting, and industrial applications, require the concurrent generation of accurate multiple-frequency PWM signals. Although separate ICs exist in the market, synthesis of PWM signals in FPGAs can integrate these components into the embedded architecture. This results in reduced cost, power consumption, and circuit board space, which improves system reliability and performance. The developed environment is compact and low-cost, and the generated signals are accurate. As a proof of concept, this solution is used in three practical scenarios. The first scenario is concerned with the control of LED flashers and audio tones. This solution can be found in many industrial applications and the automotive applications as well. The second scenario is concerned with the generation of waveforms that are smoothed via an add-on Butterworth LPF. This makes the developed system to act as a function generator that produces PWM signals and sinusoidal waves. Hence, the PWM signal converted the digital values, produced by the FPGA pin, into an analogue signal, and the system worked as a simple DAC. Such signals are required in many fields of applications such as communications, electric circuits, consumer electronics, automation, etc. The third scenario is concerned with the generation of PWM signal to control the motor speed by changing the duty cycle percentage. Voltage measurements are taken from the FPGA pin that produces the PWM signal based on a variable duty cycle value. The voltage on the motor's terminals is measured as well, and then the speed of the motor is calculated. Ten experiments are executed based on various duty cycle values, and then results are plotted in four curves. The solution is applicable to other DC motors. In fact, the three scenarios may exist in one application such as cars or an industrial application, where digital control of a production line, for example, employs them all in one solution. Moreover, the developed environment can be used for education of PWM and related applications such as the digital control of analogue systems.

REFERENCES

- *Abdelmessih, G. Z., Alonso, J. M., & Perdigão, M. S. (2016, September). Hybrid series-parallel PWM dimming technique for integrated-converter-based HPF LED drivers. In 2016 51st International Universities Power Engineering Conference (UPEC) (pp. 1-6). IEEE. https://doi.org/10.1109/upec.2016.8113996.*
- *Amorndechaphon, D. (2016, September). High-efficiency PWM DC-AC inverter for small PV power generation system. In 2016 International Conference on Cogeneration, Small Power Plants and District Energy (ICUE) (pp. 1-6). IEEE. https://doi.org/10.1109/cogen.2016.7728941.*
- *Barua, J., & Abedin, Md. A. (2018, November). Design and Implementation of a FPGA Based Closed Loop Speed Controller for DC Motor using PWM Technique. In 2018 International Conference on Advancement in Electrical and Electronic Engineering (ICAEEE) (pp. 1-4). IEEE. https://doi.org/10.1109/icaeee.2018.8642965.*
- *de Castro, A., Sutter, G., Huerta, S. C., & Cobos, J. A. (2007, February). High Resolution Pulse Width Modulators in FPGA. In 2007 3rd Southern Conference on Programmable Logic (pp. 137-142). IEEE. https://doi.org/10.1109/spl.2007.371737.*
- *Estes, F., Lentijo, S., & Monti, A. (2005, June). A FPGA-based approach to the digital control of a class-D amplifier for sound applications. In 2005 IEEE 36th Power Electronics Specialists Conference (pp. 122-126). IEEE. https://doi.org/10.1109/pesc.2005.1581612.*
- *Guo, X. Y., Wan, G. C., & Tong, M. S. (2019, December). An Intelligent Control System of Music Rhythms by RGB-LED Lamp. In 2019 Photonics & Electromagnetics Research Symposium-Fall (PIERS-Fall) (pp. 1622-1627). IEEE. https://doi.org/10.1109/piers-fall48861.2019.9021892.*
- *Intel FPGAs, C. (2021). Cyclone® IV FPGAs Devices - Intel® FPGA. Retrieved 14 March 2021, from https://www.intel.com/content/www/us/en/products/programmable/fpga/cyclone-iv.html.*
- *Jung, H. M., Kim, J. H., Lee, B. K., & Yoo, D. W. (2010, June). A new PWM dimmer using two active switches for AC LED lamp. In The 2010 International Power Electronics Conference-ECCE ASIA- (pp. 1547-1551). IEEE. https://doi.org/10.1109/ipec.2010.5544536.*
- *Karthikeyan, K., Jaisheela, J., Kumar, M. D., & Kumar, K. S. (2011, July). Sinusoidal PWM signal generation using TMS320C6711 DSP for power control in mobile phones. In 2011 International Conference on Process Automation, Control and Computing (pp. 1-4). IEEE. https://doi.org/10.1109/pacc.2011.5979019.*
- *Kodama, K., & Koutaki, G. (2019, October). Development of guitar playing robot by PWM control of solenoid. In 2019 IEEE 8th Global Conference on Consumer Electronics (GCCE) (pp. 291-293). IEEE. https://doi.org/10.1109/gcce46687.2019.9015587.*
- *Liang, J., Gao, S., & Li, Y. (2011, December). Design of audio directional system based on FPGA. In Proceedings 2011 International Conference on Transportation, Mechanical, and Electrical Engineering (TMEE) (pp. 613-616). IEEE. https://doi.org/10.1109/tmee.2011.6199278.*
- *Lu, J., & Wu, X. (2009, May). A novel multiple modes PWM controller for LEDs. In 2009 IEEE International Symposium on Circuits and Systems (ISCAS) (pp. 1767-1770). IEEE. https://doi.org/10.1109/iscas.2009.5118118.*
- *Martins, D. S., Tozetto, E. H., Iano, Y., & Swart, J. W. (2017, November). PWM algorithm for real-time power monitoring and brightness control of large scale matrix leds. In 2017 Brazilian Power Electronics Conference (COBEP) (pp. 1-6). IEEE. https://doi.org/10.1109/cobep.2017.8257350.*
- *Mondal, S., & Sharma, R. K. (2019, July). Low Power Hardware Based Real Time Music System and Digital Data Transmission Using FPGA. In 2019 International Conference on Communication and Electronics Systems (ICCES) (pp. 21-23). IEEE. https://doi.org/10.1109/icces45898.2019.9002227.*
- *Park, C., Kong, T. H., & Cho, G. H. (2014, September). Low drain voltage S/H type PWM LED current driver for BLU in mobile LCD. In Proceedings of the IEEE 2014 Custom Integrated Circuits Conference (pp. 1-4). IEEE. https://doi.org/10.1109/cicc.2014.6946086.*
- *Payak, M., & Kumbhar, S. R. (2015, October). FPGA based PWM control of lnduction motor drive and its parameter estimation. In 2015 International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT) (pp. 631-635). IEEE. https://doi.org/10.1109/icatcct.2015.7456961.*
- *Saragih, Y., Dermawan, R. D., Latifa, U., & Ming, C. C. (2020, June). Smart Angklung 2 Octave. In 2020 3rd International Conference on Mechanical, Electronics, Computer, and Industrial Technology (MECnIT) (pp. 276-281). IEEE. https://doi.org/10.1109/mecnit48290.2020.9166645.*
- *Singh, P. D., Sharma, A., & Gao, S. (2018, June). PWM based AC-DC-AC converter for an Isolated hydro power generation with variable turbine input. In 2018 2nd International Conference on Power, Energy and Environment: Towards Smart Technology (ICEPE) (pp. 1-6). IEEE. https://doi.org/10.1109/epetsg.2018.8659179.*
- *Sreekanth, C., & Moni, R. S. (2013, April). Fpga implementation of simplified amtc-pwm algorithm for a single phase voltage source inverter. In 2013 IEEE Conference on Information & Communication Technologies (pp. 1282-1287). IEEE. https://doi.org/10.1109/cict.2013.6558299.*
- *Svilainis, L. (2012). Comparison of the EMI performance of LED PWM dimming techniques for LED video display application. Journal of display technology, 8(3), 162-165. https://doi.org/10.1109/jdt.2011.2175362.*
- *Wojtkowski, W. (2018, September). Automotive LED Lighting with Software PWM Generators. In 2018 VII Lighting Conference of the Visegrad Countries (Lumen V4) (pp. 1-4). IEEE. https://doi.org/10.1109/lumenv.2018.8521129.*
- *Yoo, H., Kim, J. H., & Sul, S. K. (2007). Sensorless operation of a PWM rectifier for a distributed generation. IEEE Transactions on Power Electronics, 22(3), 1014-1018. https://doi.org/10.1109/tpel.2007.897094.*