



Article ID 1007-1202(2023)03-0271-06

DOI <https://doi.org/10.1051/wujns/2023283271>

High-Precision Dead-Time Intellectual Property Core and Its Compensation for Inverters

□ CHEN Hao¹, LIU Sanjun^{1†}, LAI Guohong^{1,2}

1. College of Intelligent Systems Science and Engineering, Hubei Minzu University, Enshi 445000, Hubei, China;

2. College of Physical Science and Technology, Central China Normal University, Wuhan 430070, Hubei, China

© Wuhan University 2023

Abstract: In the inverter circuit, there exists a specific on-off time in each power transistor. As such, to prevent a short circuit of the two switch devices on the upper and lower bridge arms, a specific dead time must be set in the pulse width modulation (PWM) and the sinusoidal pulse width modulation (SPWM) signals. In this paper, an intellectual property (IP) core that can introduce a high-precision dead time of arbitrary length into PWM or SPWM signals of the inverter is designed to increase the precision, convenience and generalization of dead time control, resulting in a boosted control accuracy of up to 10 ns. Moreover, the added Avalon bus enables IP cores to be accessed by the field programmable gate array (FPGA) processor in a standard manner and multiple IP cores of the same class can be easily incorporated. In addition, an application for setting and compensating for dead time in a three-phase inverter based on system on programmable chip (SOPC) technology is presented. With the Nios II CPU as its core, the system adopts the mean voltage compensation method to calculate the compensation voltage, and performs dead-time compensation in a feed-forward manner. The three dead-time IP cores are controlled by Avalon bus. These allow the dead time of three groups of power transistors to be accurately controlled and flexibly adjusted. The system also features the master computer communication function while boasting the advantages of flexible control, high precision and low cost.

Key words: field programmable gate array (FPGA); dead-time; sinusoidal pulse width modulation (SPWM)

CLC number: TM 464

0 Introduction

The inverter is an essential device in photovoltaic power generation, wind power generation, and other technical applications and therefore possesses significant research value^[1,2]. An inverter can convert direct current (DC) to alternating current (AC). In practical applications, the power transistor in the main circuit of the in-

verter is not an ideal device. It is therefore necessary to set a period of dead time in its sinusoidal pulse width modulation (SPWM) or pulse width modulation (PWM) signal to ensure that the upper and lower power transistors of the same bridge arm are not turned on at the same time. The length of the dead time can have a significant impact: the upper and lower bridge arms' power devices are still subject to the risk of shorting out if the dead

Received date: 2023-10-26

Foundation item: Supported by the National Natural Science Foundation of China (61961016), the Natural Science Foundation of Hubei Province (2019CFB593) and PhD Research Start-Up Foundation of Hubei Minzu University (MY2018B08)

Biography: CHEN Hao, male, Master candidate, research direction: power electronics, control technology of photovoltaic inverter. E-mail: chenhaobox1@163.com

† To whom correspondence should be addressed. E-mail: liusanjunbox1@126.com

This is an Open Access article distributed under the terms of the Creative Commons Attribution License (<https://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

time is too short; the system will manifest obvious dead zone effects such as an increase in high-order harmonic amplitude and significant distortion if it is too long, thereby compromising the system's stability and effectiveness^[3,4].

In light of the above considerations, the key to resolve the issue is to figure out how to perform dead-time compensation and generate appropriate dead time for the inverter's PWM and SPWM signals. Numerous studies have been conducted in this direction. Wan *et al*^[5] presented a generation method and a software compensation method of high-frequency dead time in inverters based on TMS320F2812. Ni *et al*^[6] proposed a pulsed dead-time compensation method, which can accurately compensate for the actual pulse in each switching cycle. Ding^[7] took the three-phase half-bridge inverter circuit as the example and analyzed the shortcomings of the two traditional dead-time compensation strategies, namely current feed-back and voltage feedback, along with designing a new dead-time compensation strategy based on digital signal processing (DSP). Zhang *et al*^[8] utilized the zero-voltage-switching (ZVT) technique to eliminate the blanking delay error, effectively reducing the total harmonic distortions (THDs) of the outputs.

The aforementioned literatures all assume a fixed inverter's dead time and finite dead time accuracy. By virtue of the high-precision adjustable dead time plan proposed in this article, the fixed dead time limit has been broken, and the user is allowed to set the dead time through the master computer in real-time. Besides, the system of this paper can achieve a dead time accuracy in the order of 10^{-8} . To accomplish highly high-precision voltage compensation in real-world applications, the post-inverter stage can fully incorporate the mean voltage feedforward compensation method.

In addition, this scheme makes use of a shift register to precisely delay the SPWM or PWM pulses using field programmable gate array (FPGA)-based custom intellectual property (IP) core technology and transforms the PWM or SPWM pulses driving the inverter. The precise amount of the delay period is set by the Nios II CPU, which also controls the position of the extraction point using the C programming language for precise correction. Afterwards, a carefully constructed logic circuit generates pairs of SPWM and negated SPWM (or PWM and negated PWM) signals. The packed adjustable dead-time IP core offers the benefits of being simple to operate, highly precise, and simple to integrate into an FPGA

system.

1 The Analysis of Dead-Time Effect and Dead-Time Precision

The one-phase bridge arm of the inverter has two pairs of complementary power transistors which take a while to be turned on or off. As shown in Fig. 1, if T1 and T2 of the same bridge arm are turned off simultaneously, damage will be caused to the transistors.

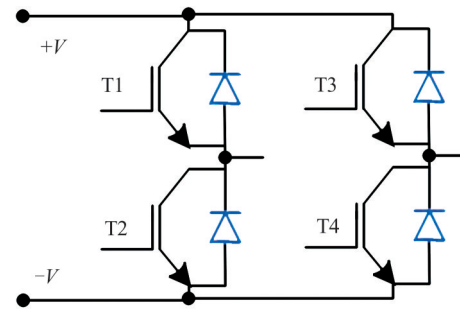


Fig. 1 The influence of dead-time accuracy analysis

Therefore, it is necessary to introduce a dead time when the driving signal is switched to ensure the safe operation of the power device. Despite being very short (usually a few microseconds), the dead time is incapable of affecting the performance of the system. Nevertheless, the building up of the dead-time effect of multiple consecutive cycles will cause the output voltage to contain a significant harmonic component, and the current waveform will be distorted^[9,10].

The dead-time effect analysis is depicted in Fig. 2. Here, a three-phase inverter is taken as an example. Setting the direction of the current flowing out of the switch tube of each phase as the positive direction, the error voltage caused by the dead time is given by

$$U_{dn} = -\text{sign}(i_n) U_{bus} \frac{t_{dn}}{T_s} \quad (1)$$

where n ($n=a, b, c$) represents the three phases of A, B and C, respectively; U_{bus} is the DC bus voltage; i_n is the phase current; t_{dn} is the dead time; T_s is the switching period of power tubes.

The phase current satisfies the following equation:

$$\begin{cases} \text{sign}(i_n) = 1, & i_n > 0 \\ \text{sign}(i_n) = -1, & i_n < 0 \end{cases} \quad (2)$$

The vector calculation equation for the three-phase voltage is as follows^[11]:

$$U_d = U_{da} + U_{db} e^{j\frac{2}{3}\pi} + U_{dc} e^{j\frac{4}{3}\pi} \quad (3)$$

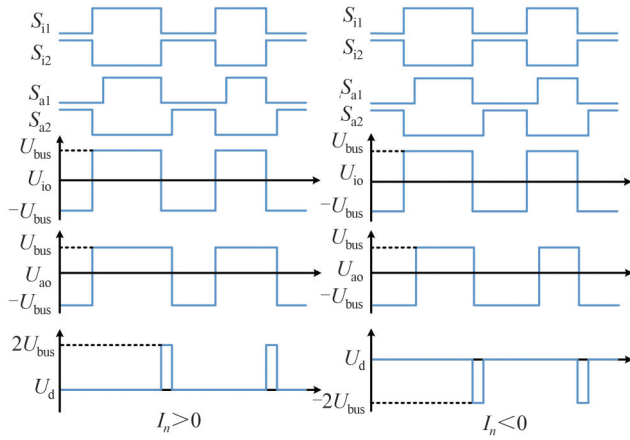


Fig. 2 Waveform analysis of dead-time effect

I_n is the phase current, S_{11} and S_{12} are the ideal driving signals of the same bridge arm, U_{io} is the ideal output voltage waveform, S_{a1} and S_{a2} are the actual driving signals, U_{ao} is the actual output voltage waveform, U_{bus} is the DC bus voltage and U_d is the dead-time voltage

Substituting Eq. (1) into Eq. (3), the resultant error voltage vector caused by the three-phase dead time can be derived to be

$$U_d = -\frac{U_{bus}}{T_s} \left(t_{da} \text{sign}(i_a) + t_{db} \text{sign}(i_b) e^{j\frac{2}{3}\pi} + t_{dc} \text{sign}(i_c) e^{j\frac{4}{3}\pi} \right) \quad (4)$$

Hence, in the bipolar modulation mode, the dead-time voltage that needs to be compensated for in each switching period is given by

$$U_{cn} = -\text{sign}(i_n) U_{bus} \frac{2t_{dn}}{T_s} \quad (5)$$

This compensating voltage is superimposed on the modulated wave voltage and is then compared with the carrier to obtain the SPWM signal.

Besides, the setting of a reasonable dead time should follow the principle of "turning off first before turning on". It ought to include the action delay time of the transistors^[12]. To maximally avoid the influence of the turn-on and turn-off delay of the transistors on the waveform, the following considerations can be made. Taking phase of A as an example, the mean value of the switching function in one switching cycle can be expressed as

$$S_d = \frac{T_{on} - t_{d_{on}} - t_{on} + t_{off}}{T_s} \text{sign}(i_a) = (D - d) \text{sign}(i_a) \quad (6)$$

where T_{on} is turn-on time in each switching period, t_{on} is the turn-on delay of the transistor and t_{off} is the turn-off delay of the transistor. Since the switching frequency is much larger than the fundamental frequency when using

SPWM, the modulating wave can be approximated as a constant within one switching period. As such, duty cycle D of SPWM waves can be expressed in the amplitude of the triangular carrier V_{tri} and the amplitude of the modulated wave V_{mod} , and it is given by

$$D = \frac{1}{2} \left(\frac{V_{mod}}{V_{tri}} + 1 \right) \quad (7)$$

Combining Eqs. (6) and (7) yields the expression for the output voltage of the three-phase system inverter:

$$\begin{cases} U_{aN} = \frac{U_{bus} V_{mod}}{2V_{tri}} - \text{sign}(i_a) U_{bus} d \\ U_{bN} = \frac{U_{bus} V_{mod}}{2V_{tri}} - \text{sign}(i_b) U_{bus} d \\ U_{cN} = \frac{U_{bus} V_{mod}}{2V_{tri}} - \text{sign}(i_c) U_{bus} d \end{cases} \quad (8)$$

According to Eq. (4), for a fixed dead time, the error voltage vector can only be six vectors based on the direction of the electric current. Hence, the error voltage is only six vectors. In other words, the accuracy of the resultant error voltage vector is governed by the accuracy of both the current direction and the dead time. Combined with Eq. (6) and Eq. (8), it is further understood that the accuracy of the dead-time voltages compensated by the three phases is determined by the accuracy of d , and the duty cycle of the equivalent dead time and the switch delay time is given by

$$d = \frac{t_d + t_{on} - t_{off}}{T_s} \quad (9)$$

This paper employs a high-precision FPGA to synthesize the dead time. By selecting a 100 MHz clock, the dead time can be adjusted within a reasonable range at 10 ns precision based on $t_d = n \cdot T_c$. Let the percentage change of dead-time voltage due to other factors be E , the required dead time be t_{dr} , and the actual dead time set be t_{da} , then the relative percentage error of the actual dead-time voltage R_{eU} is given by

$$R_{eU} = \frac{|t_{dr} - t_{da}|}{t_{dr}} \times \frac{1}{E} \times 100\% \quad (10)$$

The dead-time accuracy of 10, 15, 30 and 50 ns is analyzed in MATLAB, under the assumption of a required dead time between 300 and 350 ns and the $1/E$ value in the range of 0.998 to 1.002. The results are presented in Fig. 3. It can be inferred that the lower the dead-time accuracy, the lower the relative error percentage of the dead-time voltage. Besides, for relatively high dead-time accuracy, the relative error percentage of dead-time voltage demonstrates apparent changes.

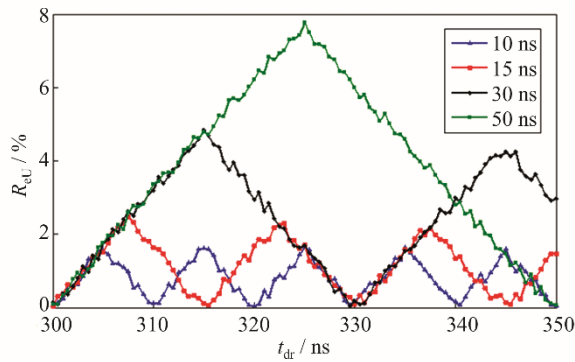


Fig. 3 The influence of dead-time accuracy analysis

2 The Design of Dead-Time IP Core

In order to realize the dynamic adjustment of dead time of a three-phase inverter, this paper takes full advantage of the system on programmable chip (SOPC) technology. One of the most prominent features of SOPC technology is the possibility of hardware design in software, in which common functional modules that can be directly controlled by Nios II CPU correspond to specific IP cores. Users are able to customize the IP core to achieve the intended function^[13]. In this paper, an adjustable high-precision dead-time IP core is customized whose structure is outlined in Fig. 4.

The IP core is mainly composed of an Avalon read/write control logic module, a delay shift register group module, a delay time register, a start control register and a dead time generation and control logic unit. It can introduce a dead time within the precision range to the input PWM or SPWM signal. Taking the input SPWM signal as an example, after being injected at SPWM_in port, the signal first undergoes a delay operation through

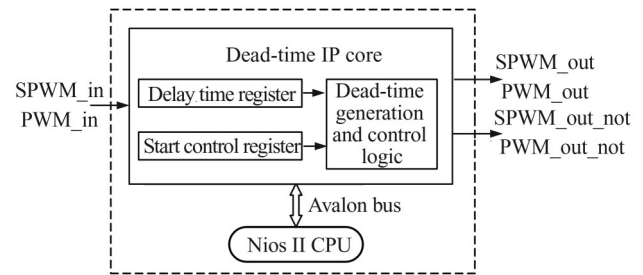


Fig. 4 Overall structure of the dead-time IP core

the delay shift register group to obtain the SPWM_delay signal.

Suppose there are N shift registers in the delay shift register group, each with the same system clock cycle T_c , and assume that the SPWM_delay signal is set by the Avalon read/write control logic to fetch from the output of the n th shift register, then the delay time of SPWM_delay signal relative to the SPWM_in signal is $n \cdot T_c$. Apparently, $N \cdot T_c$ is the largest delay time difference and T_c reflects the accuracy of the delay time. Specifically, in the hardware description language of the IP core, a macro definition of N can be made to achieve a sufficiently long delay time.

The data runs in the form of continuous flow in the delayed shift register. Concretely, on the rising edge (or falling edge) of each clock, the k -th (k is an integer, $1 < k < N$) register reads the data in the $(k-1)$ -th register and transmits its own stored data to the $(k+1)$ -th register. The details are illustrated in Fig. 5. In addition, the system defines the delay time register as a 32-bit unsigned integer. There is also logic in the delay shift register group module to make the PWM_delay signal fetch the data from the $d_data[n]$ register. The Nios II processor writes the value n to set the specific delay time.

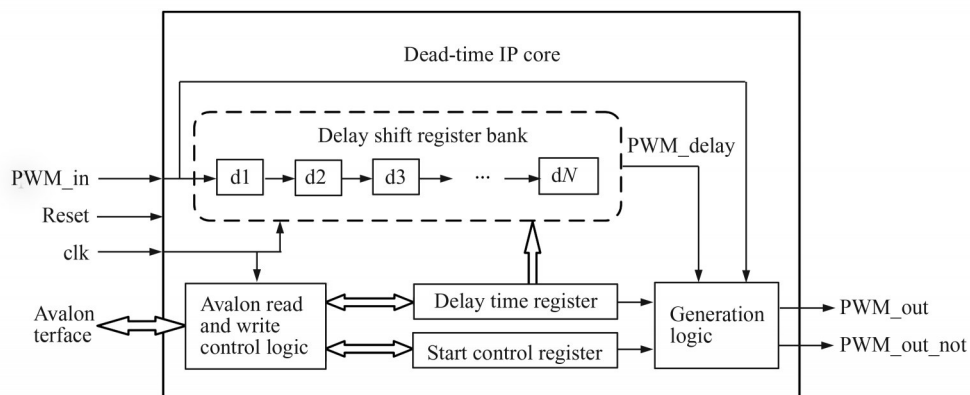


Fig. 5 Logic details of the dead-time IP core

The names of the delay time register and the start control register in the IP core are `delay_num_reg` and `start_reg` respectively. Any integer value n can be written into the delay time register through the Avalon bus to control the length of the dead time and write control commands into the start control register to dictate the start and stop of the IP core by the Nios II CPU. The process of the read/write control logic is as follows:

```
reg [31:0] delay_num_reg;
reg [31:0] start_reg;
always @(posedge clk)
if((CPU_CS==1)&&(CPU_WR==1)&&(CPU_
Addr==0))
delay_num_reg<=CPU_WR_DATA;
always@(posedge clk)
if((CPU_CS==1)&&(CPU_WR==1)&&(CPU_
Addr==1))
start_reg<=CPU_WR_DATA;
```

In addition, the input signal of the dead time generation logic contains `PWM_in`, `PWM_delay`, and `start_reg` [0], whereas the output signal contains `PWM_out` and `PWM_out_not`. Specifically, the logical relationship between the output and the input is given by:

```
assign PWM_out= (start_reg[0] ==1)? (PWM
&PWM_
delay):0;
assign PWM_out_not=(start_reg[0]==1)? (~PWM|
PWM_delay):0;
```

As detailed in Fig. 6, this paper carries out the dead-time IP core function test in Modelsim. The reference clock runs at 100 MHz. The last two signals are the finally generated SPWM signals within the dead time which is set arbitrarily with 10 ns steps under the maximum value. It can be seen from the simulation results that the function of the custom dead-time IP core meets the requirements, with nanosecond accuracy of the dead time.

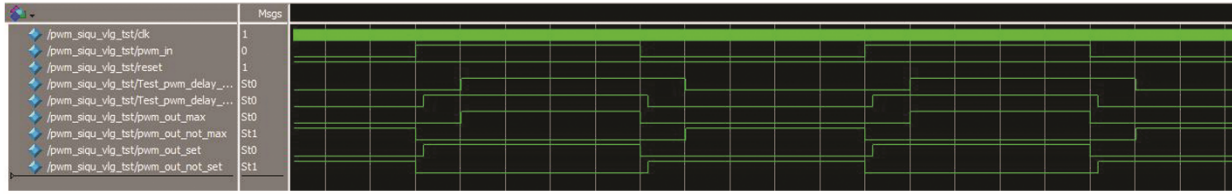


Fig. 6 Simulation of the dead-time IP core in Modelsim

3 Compensation of Dead-Time IP Core in Three-Phase Inverter

The dead time of each of the three bridge arms of a three-phase inverter can be independently altered and the appropriate dead-time compensation voltage can be obtained using the approach presented in this paper. A three-phase inverter circuit with six Insulated Gate Bipolar Transistors (IGBTs) is shown in the following example. In order to successfully control the six IGBTs of the three bridge arms on the inverter, this paper leverages three custom dead time generation IP cores to generate six control signals, and the Nios II CPU communicates with the three IP cores through the Avalon bus. The specific control structure is diagrammed in the Fig. 7.

The entire system entails a three-phase inverter circuit, a sampling circuit, a modulator circuit and a FPGA part. It is known that a reasonable dead time of each IGBT is between 0.5 and 1.2 μ s. The dead time of the

SPWM signal can be set within the accuracy of 10 ns by the master computer. Moreover, after the three-phase current is sampled by the sampling circuit, the current data is transmitted to the current direction calculation module built in the FPGA system. According to Eq. (5), the dead-time voltage will be calculated in real time, before the modified SPWM signal is generated by the modulator module.

Furthermore, the names of the three dead-time IP cores added to the SOPC system are `dzt1`, `dzt2` and `dzt3` respectively. Taking `dzt1` as an example, its base address in the FPGA development environment (i. e., Nios II Software Build Tools for Eclipse) is `DZT1_BASE`. With the assumption that the dead time of `dzt1` is $x \mu$ s, the C program language to be applied is as follows:

```
n1=(unsigned int)(x*1000/10);
IOWR_32DIRECT(DZT1_BASE, 0, n1);
IOWR_32DIRECT(DZT1_BASE, 4, 1);
```

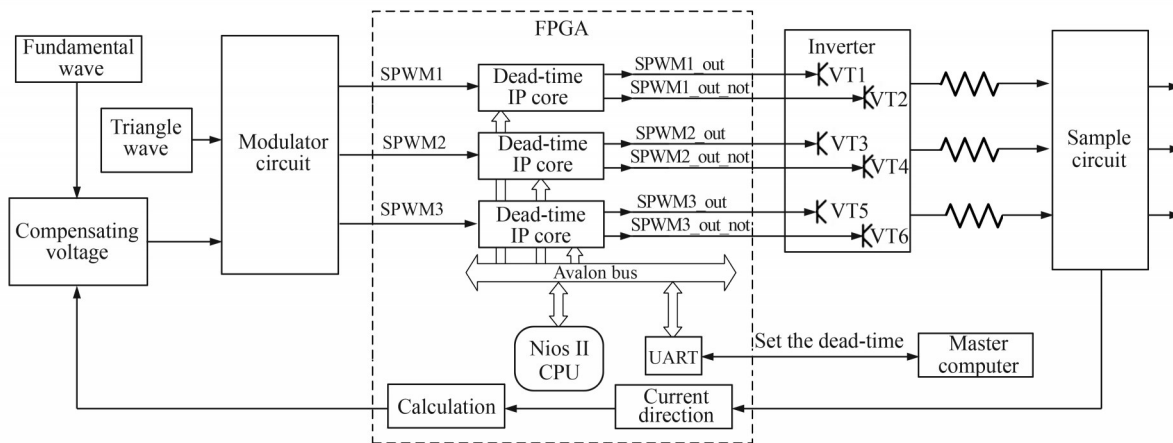



Fig. 7 System structure of a three-phase inverter within the dead-time IP core

4 Conclusion

In this paper, the design of an IP core capable of generating high-precision adjustable dead time for SPWM and PWM pulses based on FPGA is presented, along with a practical scheme for three-phase inverter circuit control and dead-time compensation using the dead time IP core. By encapsulating IP cores and customizing peripherals in SOPC technology, not only can the dead time be set to nanosecond accuracy thereby resolving the dead time effect, but the overall running speed and stability of the FPGA-based inverter system are also boosted. These improvements will allow users to build SOPC systems with dead time settings for multiple IGBTs or MOSFETs.

References

- [1] Li Y, Li J Y, Zhao W Y, *et al.* Analysis on the effect of dead-time of SPWM inverter [J]. *Power Electronics*, 2021, **55** (10): 69-73+79(Ch).
- [2] Suroso S, Nugroho D T, Noguchi T. New dead-time compensation method of power inverter using carrier based sinusoidal pulse-width modulation [J]. *International Journal of Electrical and Computer Engineering (IJECE)*, 2018, **8**(6): 4880-4891.
- [3] Jiao N, Wang S N, Liu T Q, *et al.* Harmonic quantitative analysis for dead-time effects in SPWM inverters [J]. *IEEE Access*, 2019, **7**(1): 43143-43152.
- [4] Pan W G, Zhang L, Zou Y P, *et al.* A method to compensate the dead time for three-phase PWM inverters [J]. *Advanced Materials Research*, 2014, **3530**(1044-1045): 681-687.
- [5] Wan Y Y, Li H. Research on compensation method of high-frequency dead-time based on TMS320F2812 [J]. *Electrical Application*, 2019, **38**(2): 29-32(Ch).
- [6] Ni R Z, Li T, Chen J, *et al.* Research on a pulse dead zone compensation method [J]. *Transactions of China Electrotechnical Society*, 2019, **34**(S2): 553-559(Ch).
- [7] Ding X L. Dead time characterization and compensation strategy for three-phase SPWM inverter [J]. *Power Electronics*, 2014, **48**(01): 33-35(Ch).
- [8] Zhang H L, Kou B Q, Zhang L, *et al.* Analysis and compensation of dead-time effect of a ZVT PWM inverter considering the rise- and fall-times [J]. *Applied Sciences*, 2016, **6** (11): 344.
- [9] Furutani S, Satake A, Hozuki T. Inverter dead time compensation method using on-line tuning [J]. *IEEJ Journal of Industry Applications*, 2021, **10**(2): 264-272.
- [10] Yuan J X, Zhao Z, Chen B C, *et al.* An immune-algorithm-based dead-time elimination PWM control strategy in a single-phase inverter [J]. *IEEE Transactions on Power Electronics*, 2015, **30**(7): 3964-3975.
- [11] Wang J B. *Research on Control Algorithms of Three-phase Voltage Source Inverter* [D]. Shaanxi: Shaanxi University of Technology, 2020(Ch).
- [12] Seong U S, Lee J S, Hwang S H, *et al.* Dead time compensation of single-phase grid-connected inverter using SOGI [J]. *The Transactions of the Korean Institute of Power Electronics*, 2017, **22**(2): 166-174.
- [13] Liu S J, Yang J H, Chen H. High precision absolute gravity meter based on time-to-digital conversion in FPGA [J]. *Wuhan University Journal of Natural Sciences*, 2021, **26**(4): 358-364.