Digital control circuit of a pulse voltage stabilizer implemented by Field-Programmable Gate Array

S N Titovskii\textsuperscript{1,2,3}, T S Titovskaya\textsuperscript{2} and N V Titovskaya\textsuperscript{1,2}

\textsuperscript{1}Krasnoyarsk State Agrarian University, 90, Mira ave., 660049 Krasnoyarsk, Russia
\textsuperscript{2}Siberian Federal University, 79 Svobodny ave., 660041 Krasnoyarsk, Russia
\textsuperscript{3}E-mail: sntitovsky@rambler.ru

Abstract. The article describes the digital control circuit of a pulse voltage stabilizer using a kind of integral-differential control law. The article discusses the implementation of such a control circuit on the Cyclone II EP2C20 FPGA in the form of a Leon3 IP processor core, which contains, in addition to standard configuration nodes, a specially designed timer-counter. The test results of a stabilizer prototype with the described control circuit are presented.

Currently, pulse voltage stabilizers have almost completely replaced linear ones due to high efficiency and overall dimensions [1, 2]. The applying of the digital control contour, instead of the analog one, in the pulse stabilizers eliminates the temperature and time drift of parameters, which is typical for analog circuits and is being a modern subject of study [3-8].

The task of the voltage stabilizer is to maintain the output voltage (voltage at the load) at a certain level. In pulse voltage stabilizers, this problem is solved by the periodic charge of the inductance coil, which is subsequently discharged through the load [9,10]. At the same time, it is considered that the output voltage is directly proportional to the charge accumulated in the inductance coil, which, in turn, is proportional to the duration of the charge (the duration of the pulse that opens the stabilizer power switch), and then

\[ U_{\text{cur}} = K T_{\text{pul}}, \]  

(1)

where $U_{\text{cur}}$ – current output voltage, $T_{\text{pul}}$ – pulse duration that opens the stabilizer power switch, $K$ – some constant depending on the parameters of the stabilizer circuit.

This pulse is generated by the control device based on the deviation of the current value of the output voltage $U_{\text{cur}}$ from the required (reference) $U_{\text{et}}$.

In [9-12], various dependences of $T_{\text{pul}}$ on $U_{\text{cur}}$ and $U_{\text{et}}$ are considered, which in most cases are based on the proportional-integral-differential control law.

In [13,14] the following option is considered:

\[ T_{\text{pul}} = F(U_{\text{int}}, U_{\text{dif}}) \]  

(2)

where $T_{\text{pul}}$ – pulse duration that opens the stabilizer power switch, $U_{\text{int}}$ – integrator output voltage, $U_{\text{dif}}$ – differential component of the output voltage.
As a result of stabilizer modeling, it was revealed that for the inverting integrator the best results are obtained using the following function [13, 14]:

\[ T_i^{\text{pul}} = \frac{T_p}{U_{\text{max}}} (U_{i^{\text{int}}} - 7,5U_{i^{\text{dif}}} + 2,25U_{i-1^{\text{dif}}}) \]  

(3)

where \(i\) – stabilizer cycle number, \(T_p\) – stabilizer operation period, \(U_{\text{max}}\) – conditional maximum voltage (voltage at which the duration of the output pulse coincides with the period).

A digital device for calculation requires a preliminary analog-to-digital conversion; therefore, the generalized functional diagram of the control device is as follows (figure 1):

![Figure 1. General functional diagram of the control device. ADC – analog-to-digital converter, PWM – pulse-width modulator](image)

To implement the control device in integral form, Altera Cyclone II EP2C20 FPGA was used in combination with Maxim ADC MAX1308.

To ensure the possibility of changing the functioning algorithm of the stabilizer control device, it was decided to use a processor core that supports software control. In this case, a change in the calculation algorithm is implemented by making changes to the program executed by the processor.

Aeroflex Gaisler Leon3 based on the SPARC V8 architecture was chosen as such a processor core. The Leon3 template is freely available as part of the GRLIB library.

To organize the control device on the FPGA Cyclone II EP2C20, such a hardware configuration was implemented that includes a processor, a memory controller, a system bus, two I/O ports from the GRLIB-CPL-1.1.0-b4108 library, and an additionally developed timer counter (figure 2):

![Figure 2. FPGA Cyclone II EP2C20 Hardware Configuration.](image)

In the above figure, Leon3 is the processor core; AHBC - AMBA 2.0 AHB interface controller; MC - Leon2 memory controller; AHB / APB - a bridge between the AHB and APB interfaces (AMBA 2.0); PWM - specialized timer counter; GPIO0, GPIO1 - universal input / output ports.

The need for an additional timer is justified by the long data transfer time between the processor and the input / output ports. Therefore, software detection and signal generation is accompanied by significant delays, measured in hundreds of nanoseconds, which, in turn, leads to instability of the voltage stabilizer.

The developed timer-counter incorporates a frequency capture channel (for measuring the duration of the stabilizer start-up period) and two comparison channels operating in pulse-width modulation.
mode, generating pulses of the ADC start and opening the power switch of the voltage stabilizer. A block diagram of the designed timer is shown in figure 3.

![Figure 3. Timer-counter structure.](image)

In the above figure, start is the voltage stabilizer start signal; clock - clock frequency (50 MHz); APB - AMBA 2.0 APB interface; ST - binary counter (32 bits); RF - frequency capture channel register (32 bits); RC0, RC1 - registers of comparison channels 0 and 1 (32 bits); CMP - digital comparators (32 bits).

The logic of the timer is as follows: the stabilizer start signal with its rising edge rewrites the current contents of the ST counter into the RF register and resets the counter. Digital comparators CMP generate signals corresponding to the result of comparing the current contents of the counter with the code stored in the registers RC0 / RC1. The output signals CMP0 / CMP1 become equal to 1 when the code accumulated in the counter exceeds the contents of the corresponding register RC0 / RC1.

The implementation of the stabilizer control device consists of an operational amplifier, ADC and FPGA Cyclone II EP2C20. The GPIO1 port is used to transmit control signals, the GPIO0 port is used to transfer data between the ADC and the processor. The most significant bit of the code received from the ADC is connected to the five bits of the GPIO0 port.

Data from the ADC is read after all conversions have been completed, since the results of the conversions from individual channels appear with an interval of 200 ns, while the time of program signal detection is approximately 400 ÷ 500 ns.

The block diagram of such a control device is as follows (figure 4):

![Figure 4. Control device block diagram.](image)
For clocking the FPGA Cyclone II EP2C20, a rectangular pulse generator with a frequency of 50 MHz was used, for the ADC MAX1308 - an internal clock generator with a frequency of 15 MHz.

The stabilizer control device is implemented in the form of a software-hardware complex, shown in figure 5.

![Software and hardware organization of the control device](image)

**Figure 5.** Software and hardware organization of the control device.

The rising edge from pin GPIO1 [0] restarts the PWM timer-counter and, at the same time, saves the counter value in the RF register - the time elapsed since the previous start (the duration of the previous start period $T_p$).

To start the analog-to-digital converter, the CC0 timer-counter comparison channel is used, which delays the moment of the ADC start-up relative to the moment of switching the key of the stabilizer power section, accompanied by significant impulse noise.

After the time (the number of ticks) determined by the contents of the CC0 register expires, the signal from the zero comparison channel through the GPIO1 pin [1] goes to the CONVST (Conversion Start) input and starts the ADC with its rising edge.

After the conversion of the data, the ADC generates an EOLC (End of Last Conversion) signal, which is input to GPIO1 [4]. The falling edge of the EOLC signal is detected by the program and, as a result, initiates the $T_{\text{pul}}$ calculation procedure. The calculation procedure transmits read signals (RD) to the ADC via GPIO1 [3], reads two-byte codes of the input voltages $U_{\text{int}}$, $U_{\text{dif}}$ through the GPIO0 port. Next, $T_{\text{pul}}$ is calculated from the $T_p$, $U_{\text{int}}$, $U_{\text{dif}}$ values obtained and in the register RC1 is written a code that ensures the formation of an output pulse of the required duration at the output of GPIO1 [7].

The described stabilizer uses modulation of the rising edge of the switch control pulse, therefore, the difference $T_p - T_{\text{pul}}$ is placed in the RC1 register as the pulse duration.

In the same procedure, the value 2,25 $U_{\text{dif}}$ is calculated and placed in the processor register R31, which will be used in the calculations of the next launch period.

The operation of the control device at 120 kHz is illustrated by the oscillograms shown in figure 6.

Oscillograms 6a and 6b show the stabilizer start pulses (yellow line in the lower part) and the output pulses of the stabilizer switch control (green line in the middle part) in the case when $U_{\text{int}} = U_{\text{max}}$, $U_{\text{dif}} = 0$. It can be seen that the maximum possible duration of the output pulses is approximately 5.4 µs.

Thus, at a frequency of 120 KHz, the maximum duration of the output pulse does not exceed 65% of the period. With an increase in the frequency at which the stabilizer operates, it will decrease, which, in turn, reduces the range of variation of the output current at which a constant output voltage is observed.
Figure 6. Oscillograms obtained from the stabilizer control device.

The study showed that the FPGA hardware implementation of a high-speed processor core in combination with an external high-speed high-precision ADC provides a stabilizer operating frequency of approximately 120 KHz with high quality stabilization of the output voltage. According to preliminary estimates, when implementing a control device in the form of a digital state machine, the frequency of the stabilizer operation can reach (0.6 ÷ 1) MHz or more.

References
[1] Shtrapenin G 2007 Components and technologies 8 114-8
[2] 2014 Analog world of Maxim. New power chips (Moscow, Maxim Integrated Products) 3
[3] Krasnobaev Y V, Nepomnyashchyi O V, Ivanchura V I, Pozharkova I N and Yablonskiy A P 2018 *Bulletin of the Tomsk Polytechnic University. Geo Assets Engineering* 329(11) 61-73

[4] Krasnobaev Y V, Kapulin D V and Goncharuk D V 2012 *Journal of Siberian Federal University. Engineering & Technologies* 3(2012 5) 327-39

[5] Titovskii S N et al 2019 IOP Conf. Ser.: Mater. Sci. Eng.(2019) 537 062033

[6] Hamidi A, Ahmadi A and Karimi S 2018 *Turkish Journal of Electrical Engineering and Computer Sciences.* 26(5) 2347-60

[7] Tarchanidis K N, Lygouras J N and Botsaris P 2013 *Journal of Engineering Science and Technology Review* 6(1) 38-43

[8] Titovskii S N et al 2019 J. Phys.: Conf. Ser. 1399 022051

[9] Meleshin V I 2005 *Transistor converter technology* (Moscow: Technosphere)

[10] Tsypkin Y Z 1963 *Theory of linear impulse systems* (Moscow: Fizmatlit)

[11] Ivanchura V I and Krasnobaev Y V 2006 *Modular high-speed voltage stabilizers with PWM* (Krasnoyarsk:KSTU)

[12] Lucas V A 1990 Theory of automatic control: Textbook for universities (M.: Nedra)

[13] Titovskaiia N V and Titovskii S N 2015 *Bulletin of KrasGAU* 7(106) 58-63

[14] Titovskaiia N V and Titovskii S N 2015 *Bulletin of KrasGAU* 8(107) 87-92