International Journal of Engineering & Technology, 7 (3.19) (2018) 81-84

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Device for Current Test Pulse Development Through a Diode in a Direct Direction

Nikolay Bespalov [1], Yury Goryachkin[2]

Department of electronics and nanoelectronics, National Research Mordovia State University, Saransk, Mordovia, Russian Federation
E-Mail: ka-mgu@mail.ru

Abstract

The article is devoted to the development of a device that allows to generate control current pulses to determine the current-voltage characteristic of diodes in the forward direction. To implement the device, we use NI Digital Electronics FPGA Board, which includes FPGA XC3S500E Xilinx Spartan-3E FPGA and the Linear Technology LTC2624 chip, containing four 12-bit DACs. We consider the creation of a software module via VHDL language that generates 12-bit digital code to create rectangular voltage control pulses with a successively increasing amplitude and transmitted via SPI interface as the part of 32-bit data transfer protocol, using Xilinx WebPACK ISE software.

Keywords: *debug board, FPGA, digital-to-analog converter, SPI interface, VHDL language, diode.*

1. Introduction

When they operate power semiconductor devices, which are widely represented by power diodes and thyristors, their reliability in devices of power electronics is determined by the initial quality and operating modes. At that diodes are often used in power electronics devices, in a group parallel or series connection, which requires a special selection of diodes according to the parameters of voltage-ampere characteristics (VAC), which is necessary to ensure the same thermal regime of the diodes during operation. In order to increase the reliability of diodes and power electronics devices in general, it is required to provide electric and thermal modes of their operation close to the nominal ones.

To solve this problem, it is necessary to have information about the parameter values of each individual device, which can be obtained only under special conditions of VAC measurement. In particular, some of the main parameters of a semiconductor diode are VAC parameters in the forward direction.

2. Methods of VAC characteristic measurement

The methods of VAC characteristic measurement are divided into two groups:

- static (or continuous) methods;
- dynamic (or impulse) methods.

Static methods involve the supply of a direct current through a test diode. Dynamic methods involve the feeding of short current pulses through a test diode that are repeated in time and vary in amplitude.

Among the static methods of VAC characteristic measurement, the method is most often used, during which a series of half-wave sinusoidal pulses with the increasing amplitude are fed to a test diode. The main disadvantage of continuous measurement methods is the heating of a studied device structure. This causes distortion of VAC characteristics and the occurrence of a methodical error, and also leads to a measuring range limitation.

When they measure the parameters of VAC characteristics of diodes by pulse method, a sequence of measuring pulses, increasing in amplitude, is fed to a test device. Since a pulse occurs within a short period of time, the heating of the tested device is significantly reduced in comparison with static methods.

In this regard, a pulse method is applied for measuring current development in the device ampere-voltage characteristics determination.

The concept of diffusion capacity is introduced for a diode switched on in the forward direction. The diffusion capacitance is defined as the ratio of a charge increment of excess minority carriers accumulated in p and n regions to the corresponding increment of the forward voltage at the junction. The circuit of a semiconductor diode replacement is shown on Fig. 1,a [1].

The current pulse passing through a diode for a short period of time provides minimal additional overheating of diode structures during a test. Figure 1,b shows the transient process of u_F voltage change on a diode when a forward current i_F of a rectangular shape flows through an impulse diode. Measurements must be performed in a steady state starting from the time t_0 (Fig. 1, b). This will eliminate the influence of the diffusion capacitance on the voltage-ampere characteristic of a diode in the forward direction.

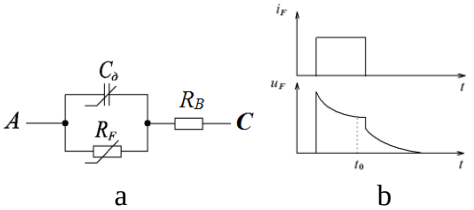

Figure 1: Diode replacement scheme at forward bias (a) and the transient process of voltage variation u_F on the diode (b) when the rectangular current pulse i_F is passed through it.

The expression for the diffusion capacity has the following form [1]:

$$c_{\partial} = \frac{q}{kT_j} i_F \tau_p,$$

consequently, the diffusion capacitance is proportional to the direct current.

The lifetime of the minority charge carriers τ_p depends on the impurity concentration in the base region. The transient process takes about (3-5)τ_p. From these considerations, the time of the current pulse flow should be chosen about $10\,\tau_p$, this will ensure that, on the one hand, the transient process ends, and on the other hand, the semiconductor structure of the tested device will not be subject to substantial overheating when current pulses i_F pass through it.

3. Device Block Diagram

The block diagram of the device forming control pulses to determine the ampere-voltage characteristic of diodes in the forward direction is shown on Fig. 2.

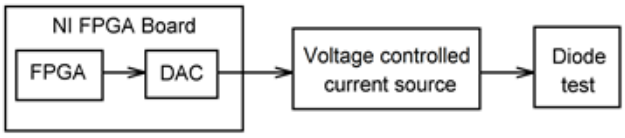

Figure 2: Block diagram of the device forming the current control pulses to determine the ampere-voltage characteristic of diodes in the forward direction

The part of the device is a debug card NI Digital Electronics FPGA Board [2-4], which is a joint development of National Instruments (NI) and Xilinx company. The basic component of the debug board is FPGA [5-6]. This debug board is widely used by developers, because it is a single convenient platform, which combines the possibilities for digital and analog circuit creation due to the fact that the board contains DAC and ADC circuits [7-8]. The conjugation of FPGA on the debug board with peripheral devices (DAC, ADC) occurs via SPI interface.

The device operates as follows. When a trigger signal arrives at the output of the debug board, a series of rectangular voltage pulses is developed with an increasing amplitude from 0 V to 3.3 V with a pitch of 3.222 mV. A generated control signal consists of 1024 rectangular pulses. Voltage pulses are fed to a controlled current source, which forms a series of measuring rectangular current pulses through a test diode respectively. The pulse duration $t_{имп}$ is chosen so that by the instant of time $t_{изм}$ of the beginning of the forward current I_{FM} amplitude measurement through a test device and the forward voltage U_{FM}, the transient process of voltage variation ends on the diode structure caused by the charge of the diffusion capacitance. The measurements of I_{FM} and U_{FM} are performed prior to the end of the current pulse. The duration of the pause t_n between pulses should be chosen such that at the beginning of a next current pulse action the temperature of the diode semiconductor structure has reached an initial value.

3.1 Components of the NI Digital Electronics FPGA Adjustment Board

The base component of the NI Digital Electronics FPGA Board is the FPGA XC3S500E Xilinx Spartan-3E. Unlike conventional digital circuits, the logic of FPGA operation is not determined during manufacture, but is set by programming. Programmers and debugging environments are used for programming, that allow you to set the desired structure of a digital device in the form of a circuit diagram or a program in special languages of equipment description. FPGA XC3S500E Xilinx Spartan-3E can be programmed with NI LabVIEW graphical programming tools or in the specialized WebPACK ISE Xilinx development environment [9].

Also an integrated circuit LTC2624 from Linear Technology is included in the debugger board. The chip contains four 12-bit DACs. The switching of digital signals formed in the FPGA to the DAC inputs is carried out using SPI interface. At that 32-bit data transfer protocol is used. The generated analog signals, whose level corresponds to the values of the input 12-bit binary code, are fed to A-D DAC outputs. Figure 3 shows the block diagram of the DAC node and its interface with the FPGA of the debug board [10].

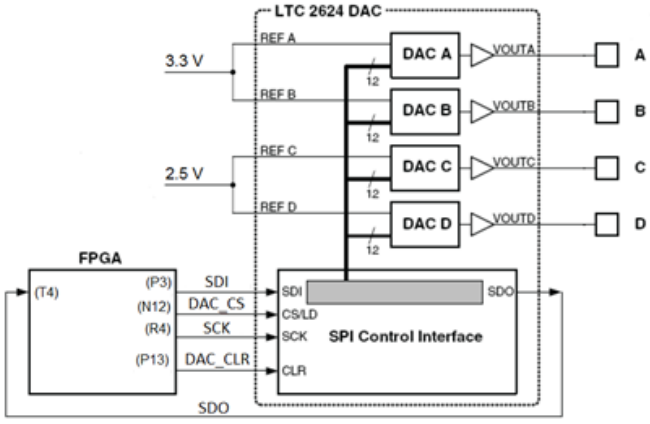

Figure 3: Block diagram of DAC node and its interface with FPGA XC3S500E

3.2 SPI interface

Figure 4 shows the timing diagram of SPI interface signals. Each bit is transmitted (received) relative to the internal SCK signal. After the assigning of a low logic level to DAC_CS signal, the FPGA transmits the data via SDI signal, starting with a senior bit. After the transfer of all 32 bits of data, FPGA completes the data transfer via SPI interface by returning DAC_CS signal to the logical 1 state.

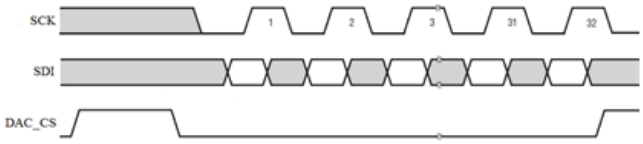

Figure 4: SPI interface signals

Figure 5 shows the communication protocol required for the interaction of FPGA with an integrated DAC circuit. Each 32-bit command consists of 4-bit command and 4-bit address, as well as of 12-bit data value. When a new command arrives to the DAC, the previous 32-bit command is returned back to a master (in this case, FPGA) [6].

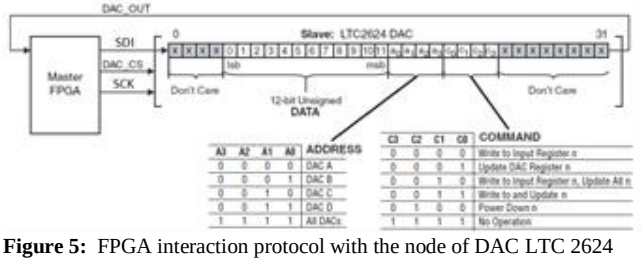

Figure 5: FPGA interaction protocol with the node of DAC LTC 2624

First of all, FPGA sends eight dummy bits, followed by a 4-bit command. FPGA then selects one or all of the output DAC chan-

nels through an address field consisting of four bits. After the address field, FPGA transmits a 12-bit unsigned data value that the DAC converts to an analog value for a selected output. Finally, four fictitious (non-significant) bits finally form a 32-bit command.

3.3 Software Implementation of SPI interface

To implement SPI interface, they use the input port RES and three output ones (SDI, DAC_CS, DAC_CLR). The implemented SPI interface allows to form a 12-bit data sending for DAC node. Thus, the signals whose level corresponds to the values of the input 12-bit binary code are fed to the terminals A-D DAC. To select channel A, they use the address field A3A2A1A0 = "0000". The program module developed in WebPACK ISE via VHDL language has the following form [11]:

```
procedure com_dac
begin
        if RES = '1' then
                memory_dac(23 downto 20) <= "0011";
                memory_dac(19 downto 16) <= "0000";
                count_DAC <= 0;
                DAC_CS <= '1';
        elsif RES = '0' then
                count_DAC <= count_DAC + 1;
                case count_DAC is
                        when 1 => DAC_CS <= '0';
                        memory_dac(15 downto 4) <=
D_reg;
                memory_dac(19 downto 16) <= "0000";
                        memory_dac(23 downto 20) <=
"0011";
                        SDI <= memory_dac(31);
                        when 33 => DAC_CS <= '1';
                        when 64 => count_DAC <= 0;
                        when others =>
                        SDI <= memory_dac(31-
((count_DAC-1) mod 32));
                end case;
        end if;
DAC_CLR <= not(RES);
end com_dac;
```

DAC interface module is a finite state machine based on the count_DAC signal. The module is implemented as a com_dac procedure, which is called by the main program as needed. If count_DAC makes one, the data is loaded into the interface memory (memory_dac) and prepared for data transfer in the direction of DAC selected channels. After all 32 bits have been transmitted, the DAC_CS signal is set to the logical unit state, in order to instruct the DAC to initiate the entire process again. This process is repeated until it is interrupted by the reset signal RES. The DAC_CLR signal is obtained by the inversion of RES signal.

3.4 Software Signal Development

A 12-bit digital code is transmitted as the part of the 32-bit data transfer protocol from FPGA to the DAC via SPI interface. The program in VHDL language, which generates a 12-bit digital code for the generation of voltage control pulses of rectangular shape with a successively increasing amplitude, has four input ports (CLK, RES, A and B), and three output ports (SCK, DATA and SYNC). CLK is a clock signal. Various combinations of signals on the ports A and B allow you to set the duration of the pause t_n between the pulses. The SCK is a clock signal. DATA is a 12-bit digital code. It is the input one to the program that generates the 32-bit protocol mentioned above. 1024 pulses are generated with an increasing amplitude from 0 V to 3.3 V with a pitch of 3.222 mV. SYNC determines the time instant t_0 of the measurement start. The program module implemented in WebPACK ISE has the following form [12]:

```
begin
if RES = '1' then
count := 0;
elsif RES = '0' then
    if   B = '0' and A = '0' then pause_width := 50000;
    elsif B = '0' and A = '1' then pause_width := 500000;
    elsif B = '1' and A = '0' then pause_width := 5000000;
    elsif B = '1' and A = '1' then pause_width := 50000000;
    end if;
if clk'event and CLK = '1' then
count := count + 1;
if count > 0 and count < pause_width then
        D_reg <= "000000000000"; CS <= '1'; SYNC <=
'0';com_dac;
elsif count >= pause_width and count < pause_width+8000 then
        D_reg <= conv_std_logic_vector(4*count_of_pulse-1,
        12);
        SYNC <= '0'; com_dac;
elsif count >= pause_width+8000 and count <pause_width+8500
then
        D_reg <= conv_std_logic_vector(4*count_of_pulse-1,
        12);
        SYNC <= '1'; com_dac;
elsif count >= pause_width+8500 and count <
pause_width+10000 then
        D_reg <= conv_std_logic_vector(4*count_of_pulse-1,
        12);
        SYNC <= '0'; com_dac;
else count := 0; count_of_pulse := count_of_pulse+1;
end if;
end if;
end if;
SCK <= not(CLK);
DATA <= D_reg;
end process;
end Behavioral;
```

The program module is a finite state machine based on the variable count. Data generation is permitted when the variable RES takes the value '0'. Also, when data is generated, an operator sets a pause duration between pulses by the combination of signals on the ports A and B. The pause can be 1, 10, 100 and 1000 ms. Each

time on the front of the clock signal CLK with the frequency of 50 MHz, the count variable is incremented by 1 and depending on the value of this variable one of four conditions is fulfilled specified by if and elsif operators. At that, a 12-digit digital code is formed to develop the control voltage pulses of rectangular shape with a successively increasing amplitude of 200 μs duration and a pause specified by an operator. Simultaneously with the voltage pulses, a pulse is formed that determines the beginning of measurements.

An example of a pulse development for measurement beginning and a voltage pulse (1024-th with the amplitude of 3.3 V) is shown on Fig. 6.

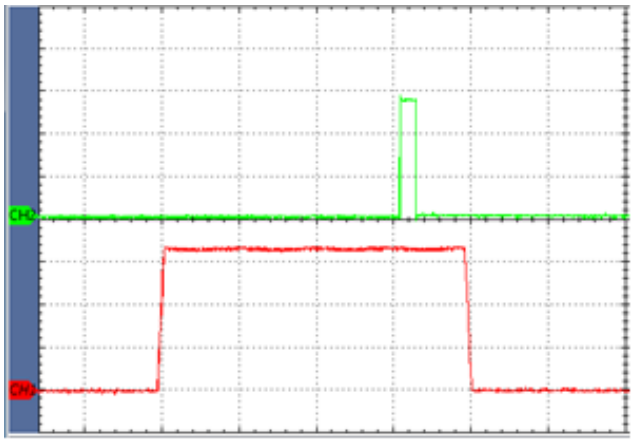

Figure 6: Oscillograms of the pulses for the beginning of measurements and the voltage pulse, respectively

3.5 Controlled Current Source

A current source controlled by voltage based on an operational amplifier and a field effect transistor is selected as a controlled current source. The block diagram of the controlled current source is shown on Fig. 7.

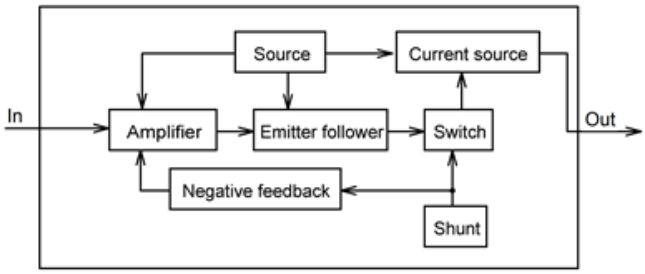

Figure 7: Block diagram of controlled current source

The diagram of this current source is widely used because of its simplicity, accuracy and current control depth. A reference voltage is supplied to the non-inverting input of the operational amplifier (Amplifier), and the inverting input has a voltage drop on the current-measuring resistor (Shunt). The signal from the operational amplifier is fed to a two-stroke emitter follower, which acts as a direct current amplifier. With zero input voltage, there is no collector current in the emitter follower and the power is not dissipated. The output of the emitter follower is fed to the gate of a field transistor (Switch), which acts as a key. Thus, the operational amplifier controls the degree of a key opening and, therefore, by the current through it. The more current on the field transistor, the greater the voltage drop on the shunt. Negative feedback is obtained. If after heating, the characteristics of the power switch change so that the current through it increases, this will cause the voltage increase on the shunt. A negative voltage difference will appear at the inputs of an operational amplifier and the output voltage of the amplifier will begin to decrease (the degree of key opening and the current through it will begin to decrease) until the voltage difference becomes zero. If the current through the key decreases, this will cause the decrease of the voltage drop on the shunt. A positive voltage difference will appear at the inputs of an operational amplifier and the output voltage of the amplifier will begin to increase (at that the degree of key opening and the current through it will begin to increase) until the voltage difference at the inputs of the operational amplifier becomes zero.

Thus, by changing the reference voltage, it is possible to control the current through the key arbitrarily, and the given current is obtained as stable, since it depends only on the magnitude of the reference voltage and shunt resistance, and does not depend on the parameters of the power switch, which can be changed significantly after heating.

4. Summary

In the process of this work performance, the following results were obtained:

- The hardware part of the control device was developed (controlled current source) to provide voltage pulses with specified parameters from the output of the debug board to a controlled current source designed to create a series of current test pulses through a diode during the determination of its VAC in the forward direction;

- they developed a program module in VHDL within the Xilinx WebPACK ISE software environment, which allows to control DAC in the debugging board;

- the created module was modeled in ISE Simulator (ISim) environment, where the correctness of this module operation was checked, was compiled and programmed in FPGA.

References

[1] Evseev Yu.A. Power semiconductor devices / Yu. A. Evseev, P.G. Dermenzhi. - M.: Energoizdat, 1981. - 472 p.
[2] National Instruments Corporation NI Digital Electronics FPGA Board. User Manual. Circuit Development Platform. – USA: National Instruments Corporation, 2009. – 52 p.
[3] NI Digital Electronics FPGA Board. http://www.ni.com/pdf/manuals/372809b.pdf
[4] FPGA Board Programmable with NI LabVIEW and Xilinx ISE Tools and Integrated with NI ELVIS II/II+. http://www.ni.com/pdf/products/us/cat_defpga.pdf.
[5] Spartan-3E FPGA Starter Kit Board.
[6] https://www.xilinx.com/support/documentation/boards_and_kits/ug230.pdf.
[7] Spartan-3E FPGA Familyt. https://www.xilinx.com/support/documentation/data_sheets/ds312.pdf.
[8] LTC2624 - Quad 12-Bit Rail-to-Rail DACs in 16-Lead SSOP.
[9] http://www.linear.com/docs/en/datasheet/2604fd.pdf.
[10] LTC1407/LTC1407A - Serial 12-Bit/14-Bit, 3Msps Simultaneous Sampling ADCs with Shutdown. http://cds.linear.com/docs/en/datasheet/1407fb.pdf.
[11] ISE Design Suite (All Editions).
[12] https://www.xilinx.com/support/download/index.html/content/xilinx/en/downloadNav/design-tools.html.
[13] Design of a BPSK modulator with VHDL.
[14] https://oghdez.webs.ull.es/scd/eng/pdf/Design%20of%20a%20BPSK%20modulator%20with%20VHDL.pdf.
[15] N.N. Bespalov, Y.V. Goryachkin, D.V. Tundykov. Software Implementation Control Devices on FPGA for Data Interchange on SPI Interface // Scientific and Technical Volga region Bulletin. – 2016. № 3. – P. 75-78.
[16] N.N. Bespalov, Y.V. Goryachkin, D.V. Tundykov. Software Implementation Control Pulse Voltage of Square Wave FPGA // Scientific and Technical Volga region Bulletin. – 2016. № 5. – P. 166-169.