Document details

Proceedings of the International Conference on Microelectronics, ICM
2014, Article number 6842192, Pages 467-470
2014 29th International Conference on Microelectronics, MIEL 2014; Belgrade; Serbia; 12 May 2014 through 14 May 2014; Category numberCFP14432-ART; Code 106344

Behavioral and physical modeling of an incremental rotary encoder (Conference Paper)

Kovacheva, M., Stoimenov, E., Yakimov, P.
Department of Electronics, Faculty of Electronic Engineering and Technologies, Technical University of Sofia, 8 Kliment Ohridski blvd., 1000 Sofia, Bulgaria

Abstract

View references (6)

This paper explains the development of a behavioral model of a rotary incremental encoder and its physical implementation using VHDL. The basic characteristics and the behavior of the encoders are studied. A digital electronic circuit generating the output signals of the incremental encoder and simulating its operation is designed. Results of the behavior model simulations are given. The physical model output signals waveforms are measured and compared with the simulated ones. © 2014 IEEE.

SciVal Topic Prominence

Topic: Synchronous motors | Permanent magnets | Incremental encoder

Prominence percentile: 66.303

Indexed keywords

Engineering controlled terms: Computer simulation Microelectronics

2014 29th INTERNATIONAL CONFERENCE ON MICROELECTRONICS

PROCEEDINGS

MIEL 2014

Belgrade, Serbia
12-15 May 2014

organized by

IEEE Serbia and Montenegro Section - ED/SSC Chapter

under the co-sponsorship of

IEEE Electron Devices Society

with the cooperation of

IEEE Solid-State Circuits Society

IEEE Catalog No. CFP14432-PRT
ISBN 978-1-4799-5295-3

Published by the

ELECTRON DEVICES SOCIETY

of the

INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS, INC.

IEEE Catalog Number:	CFP14432-PRT
ISBN:	978-1-4799-5295-3

STEERING COMMITTEE

Asia & Pacific

S.S. Chung, National Chiao Tung University, Taiwan
S. Dimitrijev, Griffith University, Australia
K. Itoh, Hitachi, Japan
H. Iwai, Tokyo Institute of Technology, Japan
C. Jagadish, Australian National University, Australia
L. Ming, Institute of Microelectronics, China
M. K. Radhakrishnan, NanoRei, India
H. Wong, City University of Hong Kong, Hong Kong
X. Zhou, Nanyang Technological University, Singapore

Europe

S. Deleonibus, CEA/LETI, France
G. Golan, Holon Institute of Technology, Israel
Z. Jakšić, IHTM-CMTM, Serbia
V. Liberali, University of Milan, Italy
M. Ostling, Royal Institute of Technology, Sweden
V. Pershenkov, Moscow Physics Engineering Institute, Russia
R. Popović, EPFL, Switzerland
S. Selberherr, Technical University of Vienna, Austria
N. Stojadinović, University of Niš, Serbia

America

V. Arora, Wilkes University, PA, USA
J. Deen, McMaster University, Canada
M. Estrada, CINVESTAV-IPN, Mexico
J. Liou, University of Central Florida, FL, USA
M. Meyyappan, NASA, CA, USA
V. Oklobdžija, New Mexico State University, NM, USA
A. Ortiz-Conde, University Simon Bolivar, Venezuela
R. Singh, Clemson University, SC, USA
J. Stathis, IBM Research, NY, USA

PROGRAMME COMMITTEE

Chairman:
N. Stojadinović, University of Niš, Serbia

***Vice-Chairmen**:*
S. Dimitrijev, Griffith University, Australia
J. Liou, University of Central Florida, FL, USA
S. Selberherr, Technical University of Vienna, Austria
H. Wong, City University of Hong Kong, Hong Kong

Conference Secretary:
T. Jovanović, University of Niš, Serbia

CONTENTS

Mini-Colloquium
Nanoelectronics

Invited Keynote Papers

Plenary Sessions

Invited Keynote Papers

Session
Power Devices and ICs

Session
Nanoelectronic Devices

Session
MEMS and Sensors

Poster Session
MEMS and Sensors

Session
Modeling and Simulation

Session
Advanced Technologies and Devices

Poster Session
Device Physics and Technologies

Session
Reliability Physics

Session
Device Characterization

Poster Session
Reliability and Characterization

Session
Circuit Design and Testing

Session
System Design

Poster Session
Circuits and Systems

Behavioral and Physical Modeling of an Incremental Rotary Encoder

M. Kovacheva, E. Stoimenov and P. Yakimov

Abstract - This paper explains the development of a behavioral model of a rotary incremental encoder and its physical implementation using VHDL. The basic characteristics and the behavior of the encoders are studied. A digital electronic circuit generating the output signals of the incremental encoder and simulating its operation is designed. Results of the behavior model simulations are given. The physical model output signals waveforms are measured and compared with the simulated ones.

I. Introduction

Simulation and modeling are investigations approaches which take place in different fields of scientific and applications developments [1]. They are essential for success because experiments are often too expensive, too slow, too dangerous or even impossible. The most popular simulation tool for electronic circuits design is PSpice A/D, which is available in multiple forms for various computer platforms [4], while Matlab/Simulink® is the most used software environment for simulation in developing models of electrical drives [2, 3].

One of the most frequently used position transducers is the incremental rotary encoder which transforms the mechanical rotary motion into electrical output. The number of the generated pulses is proportional to the angular position of the shaft and their frequency is proportional to the angular speed. Encoders enable design engineers to control motion by providing reliable feedback within the process loop.

The aim of the research is to develop a behavioral and a physical model of a rotary incremental encoder, producing the full set of really electric output signals. In order to take into account the real character of the information provided by the encoder already in extensive simulation works during research and early development stages of the drive systems, it is necessary to create the behavioral model and the corresponding simulation structure of the encoder. There must be analyzed the output signals of the device and to develop a simulation to reproduce that source signals in its entirety. The next step is to implement the model in an integrated development environment and to validate it measuring its output signals using standard measurement apparatus. This research will compare the simulated operation of the encoder with measured results to prove that the physical operation of electronic equipment can be predicted via simulation prior to testing. Testing then is just a validation of the design that is accomplished using the simulation tools.

Encoders are mechanical to electrical transducers which output is derived by "reading" a coded pattern on a rotating disk or a moving scale. The principle of the incremental encoder operation is generation of a symmetric, repeating waveform that can be used to monitor the input motion. It produces a series of square waves as it rotates. The number of square wave cycles produced per one turn of the shaft is called the encoder resolution. The pulse range of the encoder is dictated by the number of tracks of clear and opaque lines located on the disc. Thus, the resolution of the encoder is the same as the number of lines on the code disc.

Incremental encoders are usually supplied with two channels (A & B) that are offset from one another by 1/4 of a cycle (90 electrical degrees). This signal pattern is referred to as quadrature and allows the user to determine not only the speed of rotation but its direction as well. By examining the phase relationship between the A and B channels can be determined that A leads B for counterclockwise (CCW) rotation of the input shaft as it is shown on Fig. 1a. In contrary B leads A for turning clockwise (CW). This relationship of the encoder output signals is shown on Fig. 1b. The complete signals set includes also a "zero" pulse – Z. This signal is generated using another track on the disc that has only one opaque line. Signal Z rises high once per one turn of the shaft. Generally the complete set of output signals also contains the inverse forms of the signals A, B and Z. The square wave output is inherently easy for digital signal processing techniques to handle which explains the wide usage of encoders in modern motion controllers.

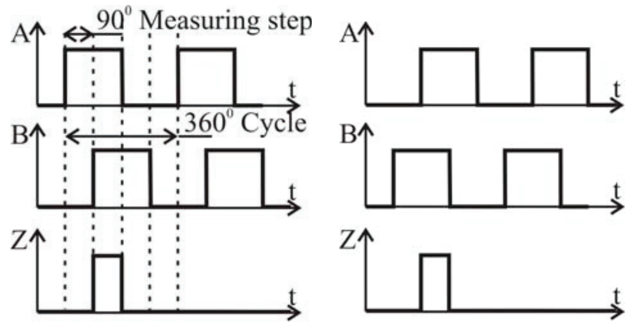

Fig. 1. Output signals relationship.

a) CCW rotation. b) CW rotation.

M. Kovacheva, E. Stoimenov and P. Yakimov are with the Department of Electronics, Faculty of Electronic Engineering and Technologies, Technical University of Sofia, 8 Kliment Ohridski blvd., 1000 Sofia, Bulgaria, E-mail: m_kovacheva@tu-sofia.bg, e_stoimenov@tu-sofia.bg, pij@tu-sofia.bg.

978-1-4799-5295-3/14/$31.00 © 2014 IEEE

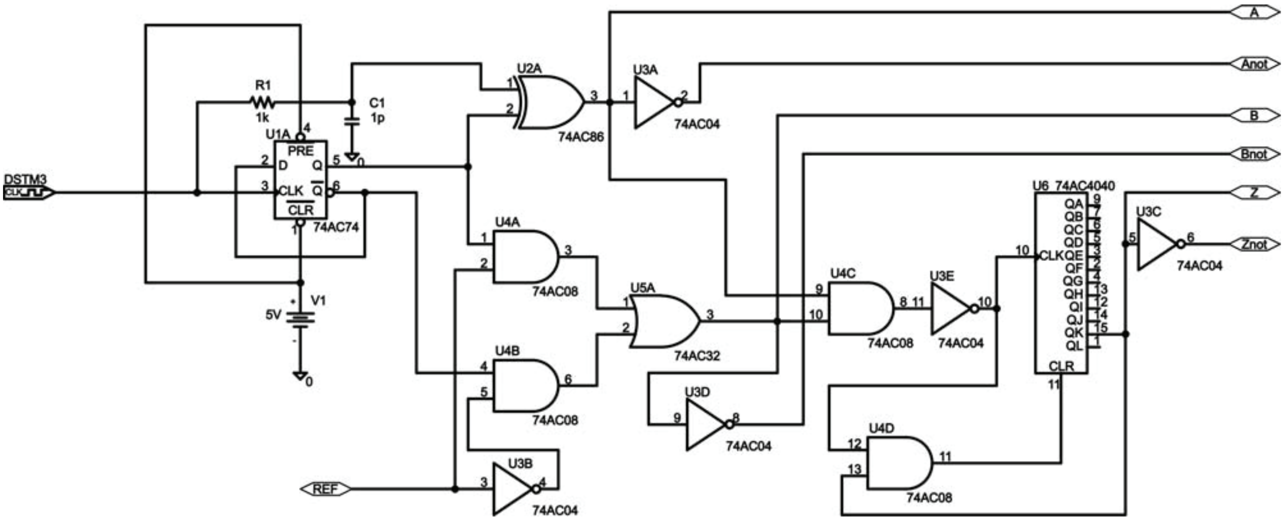

Fig. 2. Behavioral model electric circuit.

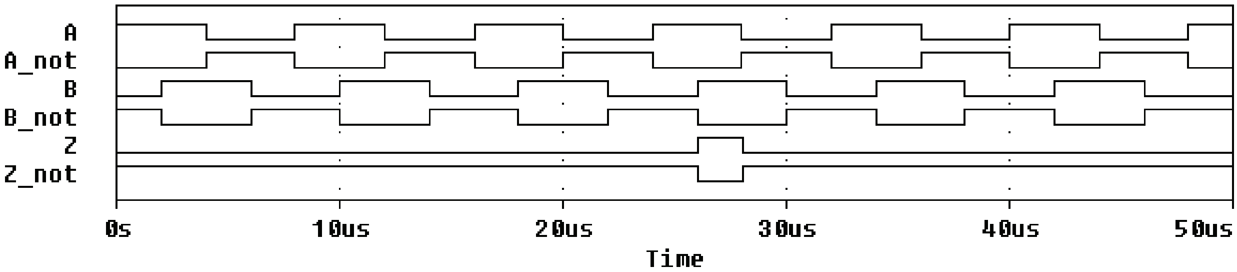

Fig. 3. Simulated output signals in CCW rotation.

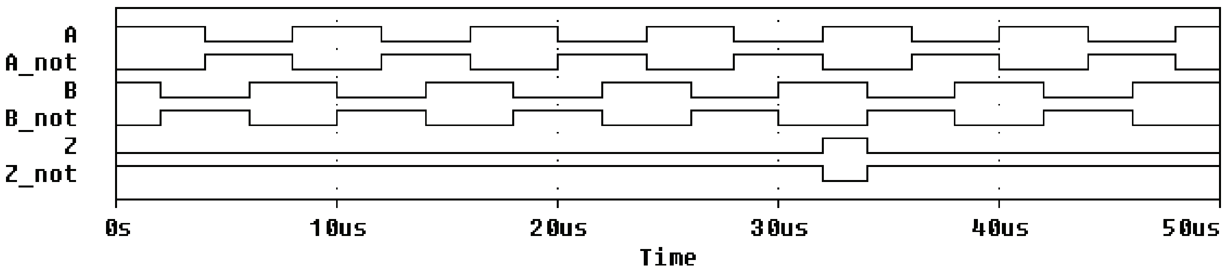

Fig. 4. Simulated output signals in CW rotation.

II. BEHAVIORAL MODEL DEVELOPMENT

To simulate the behavior of any device its operation must be analyzed in depth. In order to produce electric signals a corresponding circuit has to be designed. The behavior of the incremental rotary encoder is simulated using a proper electric circuit that produces the full set of output signals. This circuit is designed using standard advanced CMOS combinational and sequential logic circuits and is shown on Fig. 2.

The frequency of the output signals A and B is set by the digital stimulus DSTM3 and can be adjusted editing its model parameters. Thus the speed of the rotation is determined. The main part of the circuit is the generation of the signals with quadrature relation. This is obtained using the D flip-flop (U1A) acting as a frequency divider and the XOR logic gate (U2A). In this way the signal produced by the XOR gate and the signal produced by the normal output of the flip-flop are with 90 electrical degrees phase shift. If the initial state of the flip-flop is high then the signal from its normal output will lead the signal produced by the XOR gate. In the same time the signal obtained from the inverted output of the flip-flop will lag the output signal of the XOR gate. If it is accepted that the output of the XOR gate produces the signal A, then there must be a part of the circuit that will simulate the signal B and the change of the direction of the rotation. Thus there is a two-input multiplexer and the signal B is obtained from its output. An external logic signal applied to the input REF determines the relation between the signals A and B. The complete set of signals of the rotary encoder includes the “index” signal Z. For this purpose is intended another part of the circuit consisting the 12-stage binary ripple counter 74AC4040 (U6) and the logic gates U4C, U4D, U3E and U3C. This circuit must produce a pulse on every previously defined number of pulses on outputs A and B. The number of cycles of A and B signals per one turn of the encoder shaft is simulated by choosing the output of the counter which the Z signal will be derived from. In the shown example the Z signal is derived from the output QK which means that there will be 1024 cycles per one turn. The inverted forms of the signals are generated using inverters. So all output signals of a real incremental rotary encoder are available.

The operation of the circuit has been verified using the universal circuit simulator PSpice. The timing diagrams of the simulated output signals in both directions of rotation are shown on Fig. 3 and Fig. 4.

III. FPGA-BASED PHYSICAL MODEL IMPLEMENTATION

The physical model of the incremental rotary encoder is realized on Nexys3 development board. It is a complete, ready-to-use digital circuit development platform based on the Xilinx Spartan-6 LX16 FPGA. The behavioral code of the incremental encoder is created in the Xilinx program – ISE Webpack 14.5 and loaded on the FPGA board Nexys3 using the program Adept.

The proposed physical model is an improved version of that described in [5]. The new feature of the model is the possibility to work with an embedded clock source in addition to the external one as it is depicted in Fig. 5.

There are defined input signals from type std_logic: *d0* – internal clock, *d1* - external clock, supplied by a generator, *sel* – changing the clock source either from the external or from the embedded generator, *reset* – enabling the FPGA board and *ref* - changing the phase of output signals A and B. Output signals from the same type are also declared: *A* - first output signal, with 2 times lower frequency than the main clock, *A_INV* – inverted A signal, *B* – second output signal shifted with 90 degrees from A, *B_INV* – inverted B signal, *Z* - output signal, that is logical AND of A and B, and is repeated every 1024 clock periods and *Z_INV* – inverted Z signal. There are two processes declared – for forming the output signals A and B and for producing the output signal Z when both A and B are logical “1” and this pulse is generated every 1024 periods of A and B.

The multiplexer in the block diagram allows the input clock of the encoder to be selected between two sources – an external clock generator or the Nexys3 onboard 100 MHz generator. The VHDL source code of the multiplexer is shown below.

```
library ieee;
use ieee.std_logic_1164.all;              -- The multiplexer selects the clock source
                                          -- of the encoder. There are two clock
                                          -- sources - external and the internal board
entity mux is                             -- clock source.

port ( sel  : in  std_logic;              -- Select input
       en   : in  std_logic;              -- Enable pin. Tied to '1'
       d0, d1  : in  std_logic;           -- Inputs
       output : out std_logic );          -- Output
end mux;

architecture Behavioral of mux is
begin

 output <= d0 when sel = '0' and en = '1' else    -- set output to appropriate
 d1 when sel = '1' and en = '1' else              -- input based on select
 'Z';                                             -- output is undefined

end Behavioral;
```

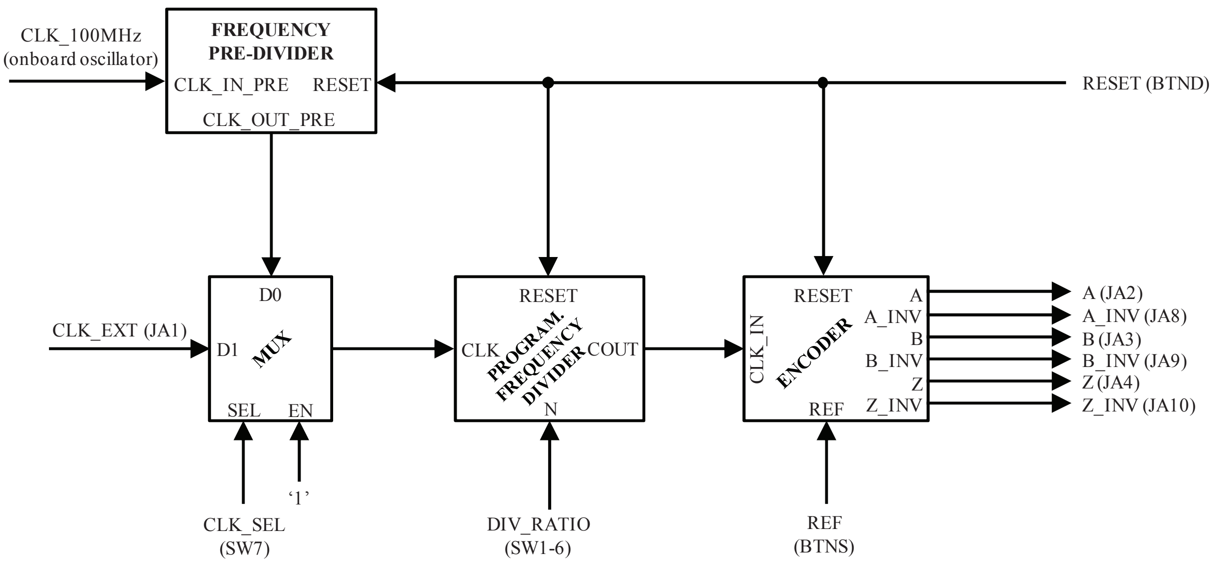

Fig. 5. Physical model block diagram.

The "SEL" input selects the clock source. It should be noted that the frequency of the Nexys3 onboard oscillator is too high for the realistic modeling of an encoder operation. In this order a frequency pre-divider is used. This block decreases the 100 MHz frequency to approximately 300 kHz which is close to a real encoder output frequency.

Moreover a programmable frequency divider is placed before the encoder block. This divider allows flexible control over the encoder output signals frequency. As shown in the source code below, the frequency divider is realized with a classical algorithm using a counter variable. The counter increments on every rising edge of the input clock signal till it reach the value of *N*. The value of the *N* constant is defined by the user. The coefficient of the frequency divider block should be multiple of 2.

```
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_unsigned.all;

entity freq_div is          -- The frequency divider divides the input frequency
                            -- of the encoder N. N should be multiple of 2.

port (
        N     :in std_logic_vector (6 downto 0);   -- Divider coefficient - N
        cout  :out std_logic := '0';               -- Output clock
        clk   :in  std_logic;                      -- Input clock
        reset :in  std_logic := '1'                -- Input reset
        );
end freq_div;

architecture Behavioral of freq_div is
        signal temporal: STD_LOGIC :='1';
        signal counter : integer := 0;
        begin
        process (reset, clk)    -- The divider is realized with counter - count.
        begin                   -- The divider coefficient N is set by the user.
        if (reset = '1') then
           temporal <= '0';
           counter <= 0;
        elsif rising_edge(clk) then
        if (counter >= (N-1)) then
     temporal <= NOT(temporal);
     counter <= 0;
        else
     counter <= counter + 1;
        end if;
        end if;
        end process;

   cout <= temporal;
end Behavioral;
```

IV. EXPERIMENTAL RESULTS

The physical model is validated measuring its output signals using a standard digital four channel oscilloscope. The code is tested on a FPGA board. The test conditions are - input clock signal with frequency of 10 kHz. The amplitude of all signals is 3,3V. On Fig. 6 are shown outputs A (CH3), B (CH2), Z (CH4) and Z_INV (CH1) when the signal *ref* is set to "1". Thus the signal A leads signal B which is shifted with 90 degrees and CCW rotation is simulated. The same signals are presented on Fig. 7 when the signal *ref* is cleared to '0' and CW rotation is simulated.

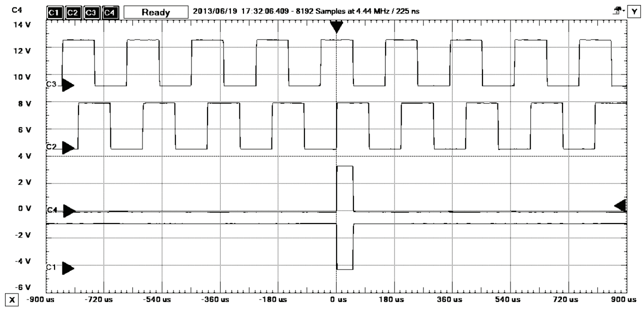

Fig. 6. Output signals simulating CCW rotation.

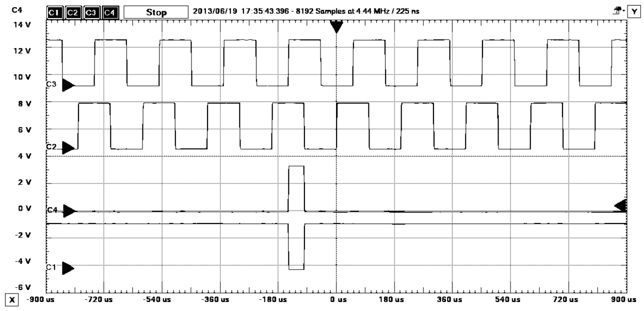

Fig. 7. Output signals simulating CW rotation.

V. CONCLUSION

The operation of a real incremental rotary encoder is fully simulated. All output signals with their normal and inverted forms are generated. The behavior of the encoder is verified using the general-purpose circuit analysis program Cadence PSpice. The physical model is validated measuring its output signals using a standard digital four channel oscilloscope. The results from the model operation correspond to the expected and are very close to the real ones.

REFERENCES

[1] S. Robinson, "Conceptual Modeling for Simulation: Issues and Research Requirements", in *Proc. of the 2006 Winter Simulation Conference*, 2006, pp. 792-799.

[2] J. J. Incze, Cs. Szabó, and M. Imecs, Modeling and Simulation of an Incremental Encoder Used in Electrical Drives, in *Proc. of 10th International Symposium of Hungarian Researchers on Computational Intelligence and Informatics CINTI 2009*, Budapest, Hungary, 2009, pp. 97-109.

[3] I. Incze, A. Negrea, M. Imecs, and Cs. Szabó, "Incremental Encoder Based Position and Speed Identification: Modeling and Simulation", *Acta Universitatis Sapientiae, Electrical and Mechanical Engineering*, 2 (2010), pp. 27-39.

[4] M. Kovacheva, and P. Yakimov, "Incremental Encoder Macromodel for Educational Purpose", in *Proc. of XLVIII International Scientific Conference on Information, Communication and Energy Systems and Technologies*, Ohrid, Macedonia, June 26-29, 2013, Vol. 2, pp. 667-670.

[5] M. Kovacheva, E. Stoimenov, and P. Yakimov, "FPGA-based Model of Incremental Rotary Encoder", *Annual journal of electronics*, 2013, vol. 7, pp. 17-20.

[6] http://www.beiied.com/BEI-Industrial-Encoders.pdf.

Източник: https://www.scimagojr.com/journalsearch.php?q=144854&tip=sid&clean=0