SOFTWARE TRAINING AND DEMONSTRATION MODULE FOR OPERATIONS CONTROL BLOCK REALIZATION AND SIMULATION

Sergey Nedev, Diana Grigorova

Computer Systems Department, University of Sofia, 8 Kl.Ohridski blvd., Sofia, Bulgaria,

phone: +359 2 965 2044, e-mail: dgrigororova@tu-sofia.bg

Keywords: operations control block; processor; modelling

Abstract

The paper describes a software module, allowing the realization on circuit level of a simplified switching type operations control block (OCB) in a CPU. The module makes possible the simulation of the work of the created OCB. The described software module was created with the goal to leverage the training activity of the students majoring in "Computer Systems" at TU-Sofia.

INTRODUCTION

The operations control block (OCB) is a specialized and functionally separated part of the processor. It organizes the execution of all operations included in the instruction set of a certain processor. For each operation, the OCB generates and propagates towards the operational part of the processor (ALU, operation registers, etc.) corresponding series of control signals. These signals are distributed in space – using dedicated control lines, and in time – at different moments (elementary clock cycles). The functioning of the OCB is determined by:

- the type of operation in the executed instruction. For this reason the content of the operation code field (OPC) of the instruction register (IR) is transmitted to the OCB where it is decoded by a special decoder;
- the values of the service signals (P), representing conditions and tokens (flags), generated by the blocks and circuits in the operational part of the processor. Through them, the OCB keeps track of the processes in the operational part during the execution of the current instruction;
- the clocking signals (C), which are used to organize the work of the OCB in time.

Depending on the development of the control process in time, we can distinguish synchronous and asynchronous control. With synchronous control, the execution time of all operations is permanent and is called synchronous cycle. It consists of a certain number of clock cycles (called elementary), within each of which some elementary operation is being executed. With asynchronous control, the execution time of different operations is variable. Each operation is allocated only the time necessary for its execution. We will discuss only synchronous OCB further in the paper.

The functional behavior of an OCB can be described using timing diagrams, representing the distribution of control signals on its outputs. An example of such description is shown on Fig. 1.

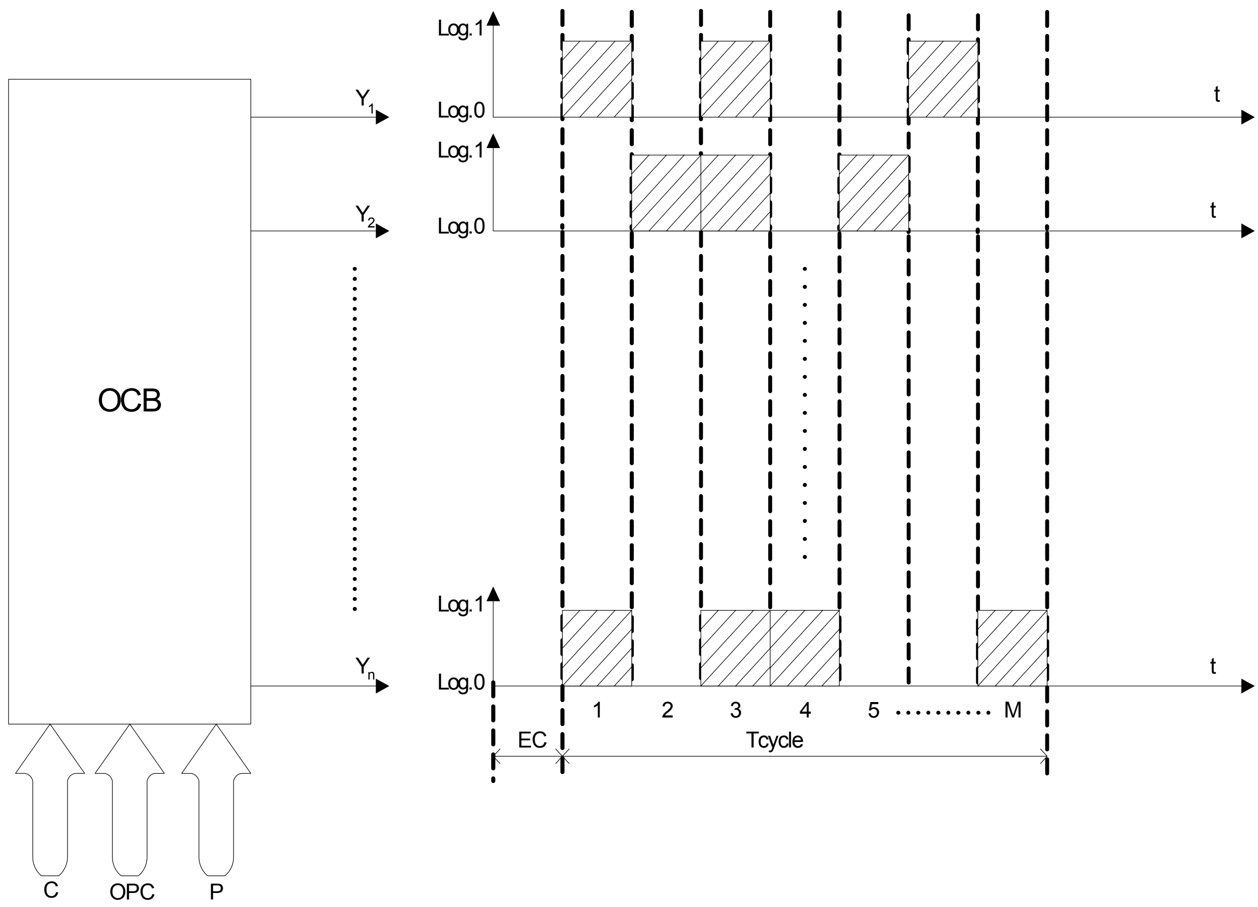

Fig.1

The software module was developed with the goal to help draft the circuitry of a sample OCB and to simulate its work.

DESCRIPTION OF THE SOFTWARE MODULE AND ITS FUNCTIONAL CAPABILITIES

The module has been realized using Microsoft Visual Basic 6.0. This is a software development environment allowing rapid and easy execution of Windows projects. Visual Basic is a complete high level programming language rendering powerful means for development of application user interfaces.

The program module makes possible the drafting of the OCB circuitry. The OCB circuitry is designed using a heuristic approach, applicable with limited size switching type synchronous control devices (small number of elementary clock cycles, operation code signals OPC and conditions P). The description and logical design of the OCB in this case is much simpler. Its functional behavior is described using timing diagrams, showing the distribution of the signals on each of the OCB outputs in relation to the input signals for Operation Code (OPC) and Condition (P), similar to the example from fig. 1. The program module allows the introduction of up to four such diagrams.

It is convenient that the OCB design is performed according to the block-diagram shown on Fig. 2. The input signals X are generalized as the equivalent influence of the OPC and the condition P. The module allows up to four input sets $X_0 \div X_3$, which are being encoded by the values of two input variables x_1 and x_2.

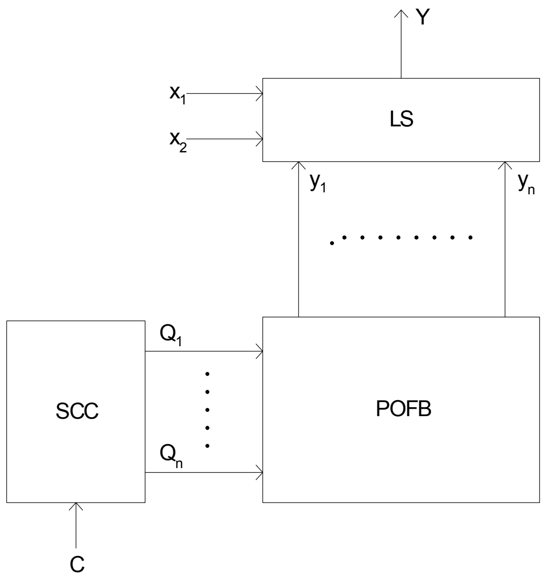

Fig. 2

The combinatory part of the OCB is divided in two parts – logical switch (LS) and partial output functions block (POFB). The POFB is intended to generate up to four time-pulse functions (similar to those on fig. 1), while the LS selects and gates to the output Y one of them, depending on the input signal X_i, active within the current cycle.

The quad JK triggers IC SN7493 is considered in order to design the synchronous cycle counter (SCC) in the program module. The modulus of this counter is equal to the number of elementary clock cycles within the synchronous cycle. The SCC counts the clock signals C in the same order with all possible operations – i.e. independent of the OPC signals and the service (token) signals P. In this case the maximum modulus of counting M_{max} is equal to 16. It is possible to reduce M_{max} by a 4-AND gate, which resets the counter immediately after setting it in the first idle state.

Selecting one of the four partial output functions y_i in relation to the input signal X_i is described analytically by the generalized logical function Y of the OCB: $Y=X_1y_1 \vee X_2y_2 \vee X_3y_3 \vee X_4y_4$. This logical function represents multiplexing type of commutation. Subsequently, the LS can be implemented directly by a regular integrated multiplexer with the necessary dimension – in this particular case 4 -> 1 (SN74153). The partial output functions y_i generated at the outputs of POFB are gated to the information inputs of the multiplexer e_i in the same order. The address inputs of the multiplexer receive the input variables x_1 and x_2 by order.

The POFB is a multi-output combinatory switching circuit, which design depends on the selected element base. The selected output time-diagrams of the OCB, represented as equivalent logical functions, are the input data for the logical design. The outputs of the SCC triggers are the independent variables of those functions. In the offered software module, the partial output functions are realized by using four multiplexers with dimension 8 -> 1 (SN74151), one for each function y_i. If the corresponding logical function y_i is partially defined, the undefined values are being set to 0 or 1 in a way that would lead to the most simple and easy realization.

In order to obtain the functions y_i, the values of the corresponding functions of the given OCB time-diagrams are gated to the information inputs of the multiplexers. The address inputs of the multiplexers receive the independent variables (by order of binary weight) of the partial output functions – the outputs of the SCC triggers. If the counting modulus of the SCC is bigger than 8, then the procedure for synthesis of logical functions of n input variables with multiplexer with n-1 address inputs applies.

All necessary circuit components (triggers, multiplexers and logic elements) for building the described simplified OCB are statically deployed on the monitor screen after starting the program module. Each of the graphic objects on the screen has been assigned an auxiliary index (a number in brackets), facilitating the access to it during the connecting phase. In order to build the circuit, the user can select between the keyboard and the mouse. If certain connection is being built using the keyboard, then the "KeyInput" button from the program menu is used. Using the auxiliary index, the user can select the element from which the connection begins and the element where it ends. A dialog window pops up, inviting the user to enter the number of the output of the element from where the connection begins and the number of the input of the element where the connection ends. After pressing the "Enter" button on the keyboard or the "Connect" button from the program menu, the connection is drawn. If we have made a wrong connection, it can be removed in a similar manner. A connection can be removed either by pressing the "Esc" button on the keyboard or the "Disconnect" button from the program menu.

If we use the mouse to draw a connection, the actions are similar. The difference is that the elements, which we'd like to connect, are being referenced not by their auxiliary indices but by clicking on them with the left mouse button.

All element inputs, which remain unconnected after completing the circuit, are treated during the modeling phase as hardwired to logical "1". This is so, because the electronic components used are of TTL type.

The user has the option to store in a file his current configuration, obtained after drawing of certain connections. The file is of binary type and has an extension .sme. This file can be loaded afterwards for further simulations.

To make things easier for the user, a "Clear All Connections" button has been implemented in the program menu, removing from the screen all connection lines drawn so far, as well as a "Reset" button, which resets the SCC.

There exist three possibilities to simulate the functioning of the created circuit of a simplified OCB.

With the first one, the "Impulse" button from the program menu is used. With each pressing of the button, a single pulse of the clock signal C is generated on the SCC input. At the same time, all electrical connections in the circuit, having a value of logical "1", are highlighted in red and all connections having a value of logical "0" - in green. This way, pulse by pulse, we can trace the functioning of the circuit.

With the second possibility, the "Auto Clock" button from the program menu is used. Here the simulation is similar to the first mode. The difference is that single pulses of the clock signal C on the SCC input are being generated every 3 seconds until exhausting the number of pulses in the synchronous cycle of the OCB.

The third simulation possibility is linked to the "Fast Cycle" button from the program menu. Here, the functioning if the circuit is simulated fast enough (in the range of milliseconds), after which the four partial output functions, generated on the multiplexer outputs, are displayed on the monitor screen as time-diagrams. Simultaneously, depending on the selected values of the input variables of the logical switch, its output function is being displayed.

CONCLUSIONS

The software training and demonstration module described in the paper makes possible the tracing and visualizing of the main phases of the design of a simplified OCB. The provided simulation capabilities allow the user to determine if the electric circuit designed by himself functions according to the specification of the task.

REFERENCES

[1] Halverson M., Microsoft Visual Basic 6.0. Sofia, SoftPress, 2000.
[2] Romansky R., Basics of Computer Technique. Sofia, SUB, 2002.
[3] Borovsky B., A.Egorov. Digital Electronic Computing Machines. Sofia, Technica, 1989.