PLD architecture for realization of control automat

D.I. Kardash
Department of Computer Science and Information Protection
Ufa State Aviation Technical University
Ufa, Russia
e-mail: kardashdi@narod.ru

Abstract[1]

In article the way of hardware realization of control devices is offered. It is reached by a waste from a programming principle. The way of equipment realization is offered for such devices. Also the new PLD architecture for this purpose is offered.

1. Introduction

It is known, that hardware computing procedures performance has the maximal speed. But hardware performance of control devices (CD) is characterized by bad flexibility. Combination positive qualities of productivity and flexibility represent valuable. The modern component base allows to make such combination. It is possible at application of integrated circuits (IC) programmed logic devices (PLD). PLD FPGA structure is considered as base. In similar IC change of the equipment can be carried out at an operation phase. Such updating is possible due to simplicity of configuration information change, available in special memory.

Program-controlled calculators are realized on base PLD more often. The program control principle is restriction for increase of performance in some cases applications of similar devices. The essence of offer the used approach consists in use of the automat to creation of control electronic devices. This approach will allow to increase productivity due to a programmed control withdrawal. New architecture PLD can be applied for practical realization of it. Scope of present offers is considered technical process control.

2. Control system base elements

The similar approach and examples of control devices creation is shown in [1]. The offers in this work are development of the approach to creation of control devices automats on the basis of special architecture PLD.

Work occurs in framework of Boolean mathematics. Similarly paradigm of automaton programming control process is represented a set "statuses" of control system. About a paradigm automat programming more in detail it is possible to read in [2]. Formally such control structure can be submitted modified Petri net. In the elementary case a status can be realized by D-trigger (or other storage device in size one bit). The status can be active (true is written down) or inactive (false is written down). The list of actions is put in conformity to each status. This list can be defined in the elementary case by such commands as: set and reset of various memory bits or the control device quits. Control programs by programming languages "Graph" and "HiGraph" are similarly written [3]. However control devices creation in a considered way is hardware realization of control functions but not by software of the specified paradigm. Physically active status is represented by true in trigger of status. The actions connected to this status become active when is active this signal. Some quantity other net nodes may be connected to an output of each status.

Nodes "transitions" are used in a net in addition to statuses nodes. The given node type serves what to pass or not pass a pulse of control transfer. The transfer of a pulse is determined by a special control input. The transition node is shown on fig. 1a. Return of a pulse to a status node is necessary for the organization of functioning of this net. Therefore it is possible to add to transition a line of signal return (fig. 1b). Transition is logic element AND. At creation of a net can be a need of several transitions signals connect to one point. These lines can be united on wired OR.

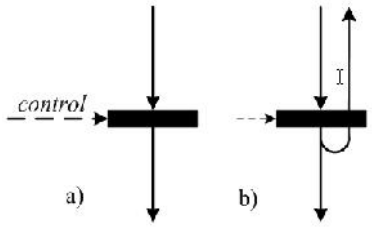

Fig. 1. Net transitions

The status can be represented by circle with input and output information lines. The example of such node is shown on fig. 2 (with the transitions connected).

Proceedings of the 14th international workshop on computer science and information technologies CSIT'2012, Ufa – Hamburg – Norwegian Fjords, 2012

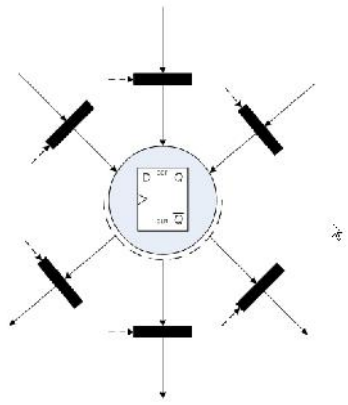

Fig. 2. Node "status" of a control net

Outputs of a status node contain also a returnable signal. Outputs are separated by dashed line for differentiation their from inputs. The essence of output connections is shown on fig. 3.

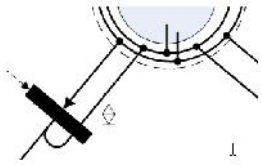

Fig. 3. Connection of the transition

It is necessary to note, that the status is realized by the circuit with memory, and transition - without it. On all status outputs is logic unit if this node is active. Otherwise - a logic zero. For transition typically that the high level of a signal appears on outputs of transition only at activity transfer moment from one status to another. Connection of status and transition can be realized by the schema shown on fig. 4. Signals lines in two used matrixes PLD (statuses matrix and transitions matrix) are shown on this schema.

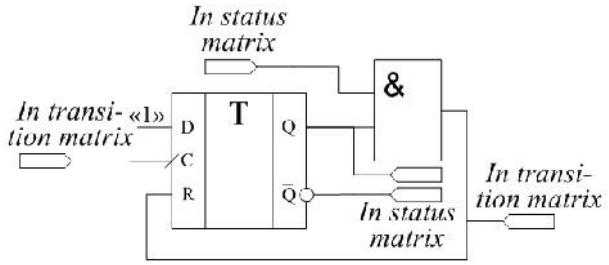

Fig. 4. Interconnection of a status and transition

Functioning of this schema is illustrated by time diagrams on fig. 5. Time diagrams are received for this schema on base FPGA PLD in program Quartus II. The signal "SostIn" comes from the previous status. The trigger of a status (a signal "SostOut") remembers logic unit in this moment. The signal of enable transition (a signal "Next") becomes active further. The narrow signal on an transition output after that is formed. Changing of this signal translates the following status node to an active level and resets previous node.

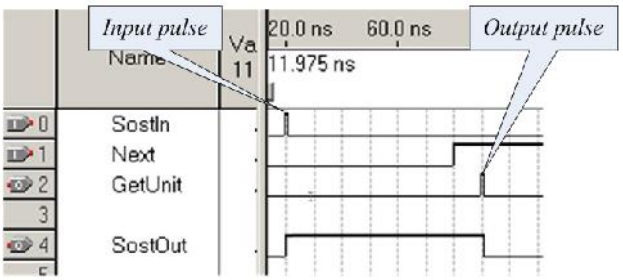

Fig. 5. Modeling of interconnection work of a status and transition

Transitions outputs connect to statuses inputs. The quantity of inputs and statuses outputs is limited only to their electric realization. Transition has a one input, and the statuses set can be connected to its output. Output signals of statuses or signals from special functional blocks (FB) connect to transitions control inputs. Also it is necessary to specify the following feature of offered schemas functioning: logic unit on status node create many copies by quantity of resolved transitions.

3. PLD architecture

It is possible to suggest many FB in addition to the described electronic cells of a switching matrix. These blocks carry out typical functions of electronic control devices. For example them can be: elements of comparison, analog-digital converters (ADC), timers, look up tables (LUT in PLD architecture), counters and other. The described elements can be united as new PLD architecture. It is submitted on fig. 6.

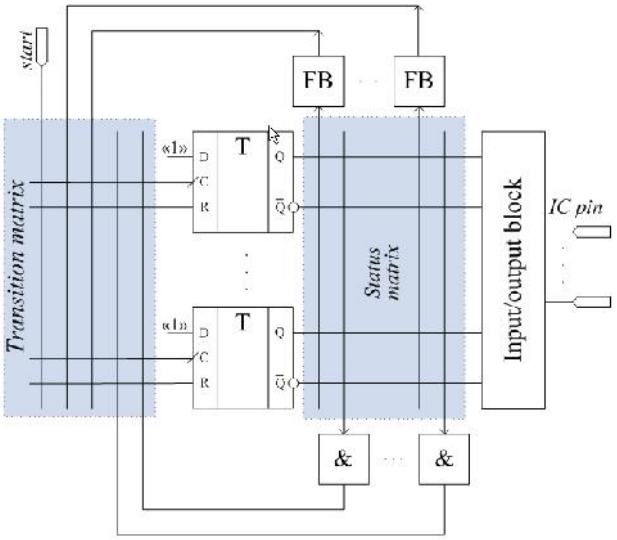

Fig. 6. Offered PLD architecture

The given architecture is created from sets: statuses, transitions and FB. Connections between them are carried out by two matrixes: transitions matrix and statuses matrix. The input / output block (IOB in PLD architecture) is connected to a status matrix. It serves for connection pin of IC to PLD signal lines. The line "Start" is included in a matrix of transitions. It is intended for starting initiation of statuses.

This architecture is submitted without here excessive detailed elaboration. Not all connections it is shown on figure. Completion of this architecture is supposed the further. For example signals as global, and from a transitions matrix can be send on information triggers inputs. In this case they can be used as signals of the circuit work enabling. Also PLD realization sees useful by several blocks having this structure (as logic array blocks (LAB) in PLD architecture). However IC of offered architecture is not unique for realization of the described construction method of control devices. Can be applied also both FPGA and CPLD IC for this purpose. However efficiency of IC use will decrease in this case.

4. Example of control structure

Described method of control devices design is possible to illustrate the on an example. Only one node is active on each moment of time in this example. However it is not restriction of a method.

It is possible to result process oven control for this purpose. The law of temperature change is present on fig. 7. Heating to 100 degrees is carried out at the first stage of controlled process. Oven control is carried out in limits no more than 100 and not less 95 degrees at the second stage. The second stage is processed during established time. The stage of cooling down to 23 degrees after its ending begins.

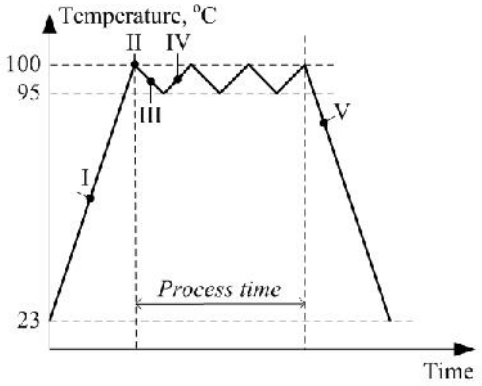

Fig. 7. Used control process

Following stage in considered process is possible to determinate: I – initial heating up to 100 °C; II – definition of a beginning of oven control time; III – cooling down to the bottom border of a range of oven control; IV – heating up to the top border of a range of oven control; V – cooling at the end of process.

The following FB also are required: "ADC" – the analog to digital converter; "D>100" – comparison of temperature with 100 °C; "D<95" – comparison of temperature with 95 °C; "D<23" – comparison of temperature with 23 °C; "Timer" – registration of oven control time.

Two carried out actions are accessible to realization of oven control: heating and cooling. The control network for realizing the control law is submitted on fig. 7. Numbers from ADC are designated by a symbol "D" in this figure. Lines of transitions control and actions start are shown by a dotted line.

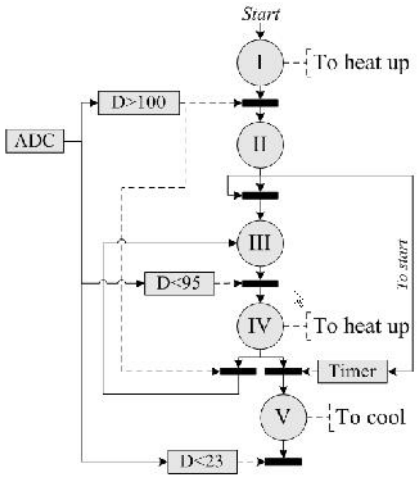

Fig. 8. Control net

The control net begins controlled process with arrival of a pulse "Start" for example a signal from the same line on fig. 6). Then statuses activity consistently is switched under the control of signals from FB. Process comes to an end, when V status will lose activity at enabling of the transition following it. The given net is realized by used electronic means above without additional structural changes.

Conclusion

The main results should be listed here:

- The hardware way of control devices building of is offered. Refusal of program control is considered as a way of productivity increase.
- The way of schema elements realization of a control net is described.
- The PLD architecture for use of control systems is developed.

References

1. Kardash D.I. "High-speed the controller on basis of programmed logic". In: The collection of Sciences academies of Bashkortostan republic proceedings for 2008. Machine conducting, control processes and critical technologies. Ufa, Russia, 2008, pp.198-204.
2. Polikarpov N.I., Shalyto A.A. "Automat programming", St. Petersburg, Russia, 2008.
3. SIMATIC S7-GRAPH V5.3 for S7-300/400. A5E00290656-01. Programming of consecutive control systems. A manual. SIEMENS.-Edition 02/2004.