Lux junior 2011
23. bis 25.9.11 Dörnfeld

Intelligent Lighting Control System for Education Buildings

Iva C. Petrinska, Filip F. Filipov, Tsvetomir Z. Dimov, Krasimira W. Filipova
Technical University of Sofia, Bulgaria

Abstract

The current paper presents investigations for the development of daylight dependent lighting control system for educational buildings. The system proposed consists of control modules for the lighting systems and shutters for every room in the building and a base station for data storage. Each lighting control module operates autonomously and consists of embedded system with programmable electronic elements of the type FPGA and FPAA. Aiming decrease of the investments for developing the presented system the power line is used as a communication medium.

Building sector is a main consumer of electrical energy for artificial lighting – it is responsible for around 40% of the electrical energy, consumed for lighting in the European Union. From these 40%, the share of the educational buildings is about 10 - 15%. The quality of lighting is a basic criterion in these buildings. It depends on several factors most of which are subjective like raise of the productivity, visual comfort and generation of different perceptions.

A lot of research projects about the role of the daylight exist, proving its positive influence on people's perceptions and productivity [1, 2]. Most of them show increase of the overall performance of the individuals when daylight is present, which corresponds to the natural circadian rhythms in the human body. Employing lighting control system, which operates according to the natural light levels, has positive influence both on pupils' focus and perceptions and on the efficiency of the lighting systems. The problem with the employment of daylight is glare. It is avoided through of an effective control algorithm which manages both lighting and shutters, thus leading to economy of electrical energy for lighting and visual comfort by means of cheap technical devices.

Achieving quality of the lighting system does not necessarily mean using more energy. On the contrary, close analysis of the visual task and the employment of appropriate control algorithm lead to decrease of the electrical energy consumption and simultaneously increase the quality of lighting.

Introduction

The current paper presents a lighting control system based on the presence of natural lighting and intended for use in educational buildings. The implementation of new data technologies for development or modernization of building automation systems [3, 6] and particularly of lighting systems, allows for centralized control, adaptive management and decrease of energy consumption in the buildings sector. At the same time these technologies contribute for better use of the positive effects of the natural daylight. The use of the power lines as communication medium provides flexibility in the lighting control of educational buildings, opportunity for control and compliance with the limits for energy consumption and fast detection of equipment malfunctions.

Lighting Control Algorithm

For the geographical coordinates of Bulgaria, a short period of transition between day and night is typical. Also the percentage of the overcast days is small. This is the reason why better use of the natural light and restriction of the use of artificial lighting, aiming energy efficiency, may be realized by employing simple control systems (on/off control of the separate rows of luminaires in the room), instead of complicated dimming systems [4, 5]. The illumination from the natural light inside changes dynamically during the day and its value depends on a lot of factors – position of the Sun, geographical exposure of the room, cloudiness, season and others. That is why the natural illuminance is estimated indirectly by daylight coefficients (DC) [4]. The daylight coefficients are calculated by (1), where E_{in} is the natural illuminance inside, E_{out} is the diffuse component of the natural illuminance outside (overcast sky) and both illuminances are measured simultaneously [3, 4].

(1) $DC = (E_{in}/E_{out})*100$, %

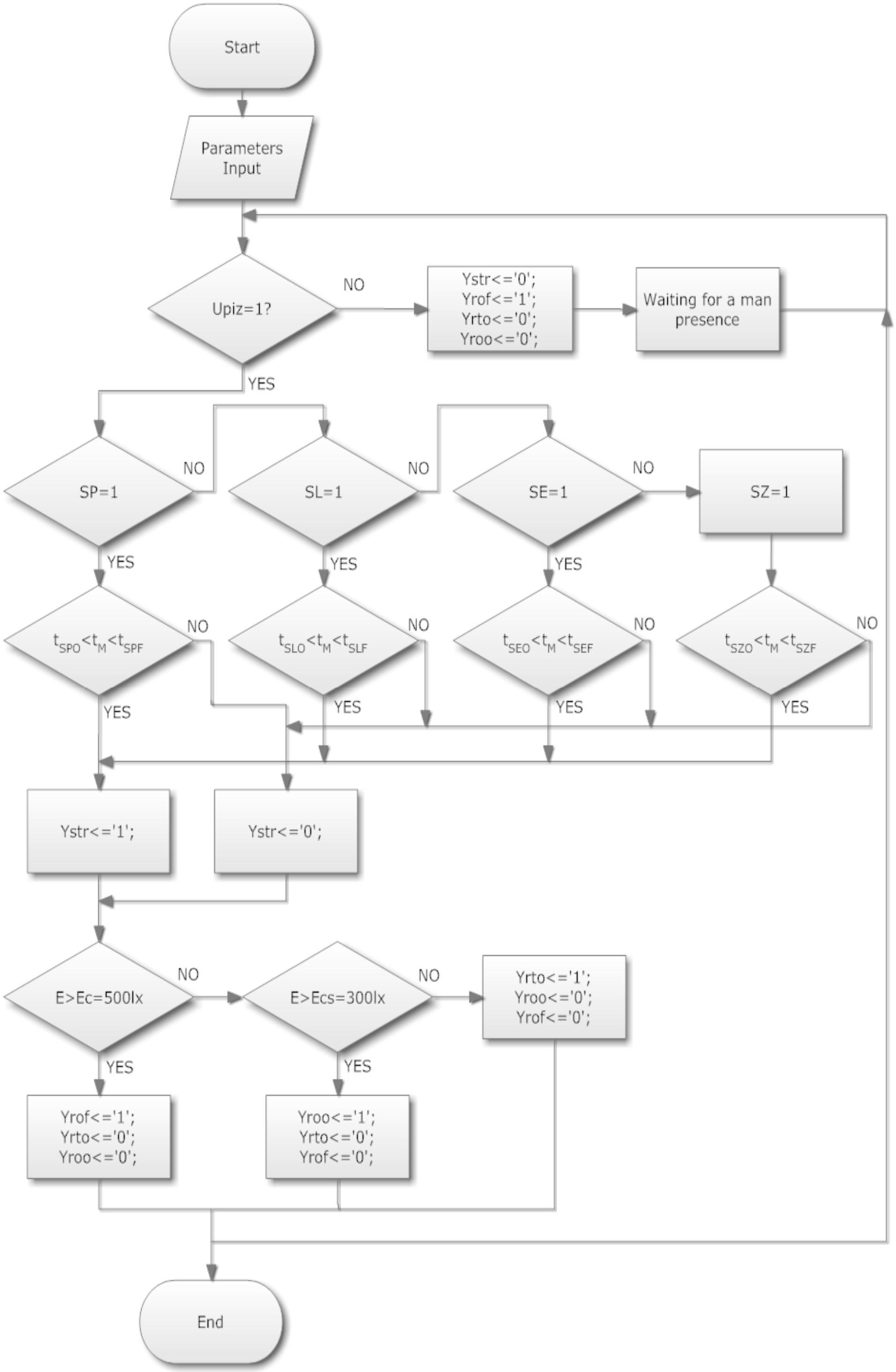
Critical natural lighting (E_{crit}) is a term used to define the level of the outside illuminance, when on the working place inside with known DC, the norm illuminance (E_{norm}) is reached:

(2) $E_{crit}=(E_{norm}/ DC)*100$, lx

Knowing this value of the natural illuminance allows daylight – dependent lighting control of the luminaires by rows (groups). The definition of the values of E_{crit} for each day of the year is only possible if the DCs for the different working places are defined for different geographical orientations and according to the depth of the premises. These values are experimentally defined by means of year-round measurements, carried out in the Technical University of Sofia. The results acquired are statistically verified and used for development of the lighting control algorithm, presented in the current paper and patented by the University [3].

When there is a lighting control system implemented, the effective time of use of the lighting system depends on the working schedule and the time when the value of the critical natural illuminance is reached for two conditional groups (rows) of luminaires. The values of E_{crit} for the two groups of luminaires will be different, because the DCs are different. In other words the artificial lighting system will work from the beginning of the working time until the norm illuminance is reached in the room and in the evening from the moment, when the illuminance inside decreases under its norm value until the end of the working time.

For estimation of the illuminance outside, a photo sensor is used in the proposed control algorithm. It will be mounted outside and the lighting system and the blinds will be automatically controlled - if in the morning, when the first person comes in (according to a signal from a presence detector) the natural illuminance outside is below the critical value, first the shutters are opened, and after that one or both groups of luminaires are turned on – fig. 1. Whenever the natural illminance reaches its critical value for the first group of luminaires, it is turned off and only the second group remains lit until the critical illuminance outside reaches the value defined for it to turn off. Besides the automatic control, manual override of the system is possible, whenever necessary.



Legend:
Upiz – Presence of a man signal
Ystr – Window blinds signal
Yrto – Both light lines are turned on
Yroo – The first light line is turned on and the second line is turned off
Yrof – Both light lines are turned off
SP – Spring season
SL – Summer season
SE – Autumn season
SZ – Winter season
Tm – Current time
Tspo – Start of daylight hours in the spring
Tspf – End of daylight hours in the spring
Tslo – Start of daylight hours in the summer
Tslf – End of daylight hours in the summer
Tseo – Start of daylight hours in the autumn
Tsef – End of daylight hours in the autumn
Tszo – Start of daylight hours in the winter
Tszf – End of daylight hours in the winter
Ec – Norm for light intensity (500lx)
Ecs – Critical norm for light intensity (300lx)

Fig. 1 Lighting control algorithm

Choice of a hardware and architecture of the system

The embedded systems are a combination of computer hardware, software and electromechanical parts, connected in such a manner, that they can perform one or a combination of functions [7]. The end user cannot program systems of this type i.e. he can make choices, related to the functionality of the system, but he cannot change this functionality through addition or change of the software [8, 9]. The embedded systems are usually implemented by means of the less powerful computers that can assure the functionality of a system or equipment. The use of embedded systems of the type FPGA (Field Programmable Gate Arrays) is related to a great flexibility and integration of software and hardware from the design phase and there is a possibility for changes in the functionality by the end user [10]. The process of embedding of a certain project on FPGA can be separated in few successive processes – implementation of the project, synthesis, positioning and creation of interconnections. At first the process can be presented schematically or by algorithm (fig. 1). The next step is the simulation of the process on RTL (register-transfer level) level. This is the first of series simulation processes that are made on the different levels of abstraction of the process and lead to the real physical programming of the FPGA. After this the RTL simulation must be presented as a series of bites that can be programmed on the FPGA. An intermediate process is the synthesis and it is also presented as a series of steps. Firstly the algorithm of the process, taken in consideration is converted to a device netlist format and then the resulting file is converted to hexadecimal code. Finally the hexadecimal code that has been created is transferred to the physical FPGA device. The functional simulation is made after the synthesis and before the physical programming of the device. This step is necessary to ensure correct logical functionality. On this stage, the simulation is a longer process. After positioning and routing, a binary program file is created for device configuration [6].

The choice of FPGA embedded system for realization of the intelligent lighting control system, presented in the current paper is not accidental – this type of embedded systems are just a chip, in which by means of appropriate software the whole functionality of the system is set. This leads to avoidance of the drawbacks of control systems, built with programmable logical controllers and namely big boards with multiple electronic elements and possibility for errors and also higher prices of the systems.

The architecture of the intelligent lighting control system proposed is built in compliance with the topology of the existing power lines that supply the lighting system of every room in the building. Usually the connection between the control (base) unit and the lighting system of each room has a point to multiple points topology. That means that the base unit controls the lighting systems in multiple rooms according to the situation. The distance between the base unit and the control module (FPGA chip) for a particular room can be up to 400 meters. Different configurations of the system architecture are possible. In the case of the control system, presented in the current paper power line modems will be implemented in the base unit as well as in every room in the building. Except for the communication capabilities, each room module will be able to perform control functions. That means that the data flow between the lighting system in a room and the base unit is significantly reduced. On the other hand the control units of every room send information to the base unit for the lighting conditions at a given moment.

The communication is bidirectional. The messages, exchanged between the base unit and the control unit of each room are:

- Each control unit works as a communication controller and exchanges information about the controlled values with the base unit;

– Turns on/off groups of luminaires, controls the position of the shutters – opened/closed;
– Monitors and collects data on:
  - status of group of luminaires on/off;
  - duration of operation of the groups of luminaires;
  - number of switching procedures for a group of luminaires;
  - malfunction of a luminaire in the group;
  - light flux level;
  - signal of presence detector;
  - position of the shutters.

The base unit is a main component for monitoring and control of the operation of the lighting systems in a building and its main function is to keep information about the operation of the control units for a defined period of time.

Results

For realization of the control modules for the individual rooms, based on FPGA chips, the following VHDL code has been created:

```
-- Create Date:    16:14:47 06/27/2011
-- Module Name:    Osvetlenie - Behavioral
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;
use IEEE.numeric_std.all;
--library UNISIM;
--use UNISIM.VComponents.all;
entity Osvetlenie is
   Port ( clk : in  STD_LOGIC;
                                         reset : in  STD_LOGIC;
                                         Tm : in std_logic_vector (10 downto 0);
        Sp : in  STD_LOGIC;
        Sl : in  STD_LOGIC;
                                 Sz : in  STD_LOGIC;
                                Ystr,Yrto,Yroo,Yrof : out STD_LOGIC;
                                Upir : in  STD_LOGIC;
                                Eect: in std_logic_vector (10 downto 0);
        Se : in  STD_LOGIC);
end Osvetlenie;

architecture Behavioral of Osvetlenie is
constant Tspo: integer :=7;
constant Tspf: integer :=20;
constant Tslo: integer :=6;
constant Tslf: integer :=21;
constant Tseo: integer :=8;
constant Tsef: integer :=19;
constant Tszo: integer :=9;
constant Tszf: integer :=17;
constant Tro: integer :=8;
constant Trf: integer :=17;
constant Ec: integer :=500;
constant Ecs: integer :=300;
type statetype is (idle, send1);
  signal state_reg, state_next: statetype;
begin
process (clk, reset)
  begin
    if reset='1' then
      state_reg <= idle;
    elsif rising_edge(clk)  then
      state_reg <= state_next;
    end if;
  end process;

process(state_reg,Upir)
  begin
              case state_reg is
      when idle =>
                               if (Upir = '1')then
                               if(Sp ='1')  then  --or Sl = '1')
                               --Ysp <=1;
                               if ( (std_logic_vector(to_unsigned(Tspo, 10))<= Tm) and (Tm <=std_logic_vector(to_unsigned(Tspf, 10))))then ---fix ;
                               Ystr <='1';
                               state_next <=  send1;
                               else
                               Ystr <='0';
                               state_next <=  send1;
```

```
                              end if;
                              end if;
                              if(Sl ='1')  then
                              --Ysp <=1;

                              if( (std_logic_vector(to_unsigned(Tslo, 10))<= Tm) and (Tm <=std_logic_vector(to_unsigned(Tslf, 10)))) then ---fix
                              Ystr <='1';
                              state_next <=  send1;
                              else
                              Ystr <='0';
                              state_next <=  send1;
                              end if;
                              end if;
                              if (Se = '1') then

                              if ( (std_logic_vector(to_unsigned(Tseo, 10))<= Tm) and (Tm <=std_logic_vector(to_unsigned(Tsef, 10)))) then ---fix
                              Ystr <='1';
                              state_next <=  send1;
                              else
                              Ystr <='0';
                              state_next <=  send1;
                              end if;
                              end if;
                              if (Sz = '1' or Sz = '0') then

                              if ( (std_logic_vector(to_unsigned(Tszo, 10))<= Tm ) and (Tm <=std_logic_vector(to_unsigned(Tszf, 10)))) then ---fix
                              Ystr <='1';
                              state_next <=  send1;
                              else
                              Ystr <='0';
                              state_next <=  send1;
                              end if;
                              end if;
                              else
                              Ystr <='0';
                              Yrof <='1';
                              Yrto <='0';
                              Yroo <= '0';
                              state_next <=  idle;
             end if;
              when send1 =>
                        if (Eect > Ec) then
                        Yrof <='1';
                        Yrto <='0';
                        Yroo <= '0';
                        elsif (Eect >Ecs) then
                        Yroo <= '1';
                        Yrto <='0';
                        Yrof <='0';
                        else
                        Yrto <='1';
                        Yroo <= '0';
                        Yrof <='0';
                              end if;
                              state_next <=  idle;

              end case;
end process;
end Behavioral;
```

The VHDL code has been successfully compiled and the simulation results for the operation of the system proposed are shown on fig. 2. These results are correct and show that the room modules are expected to operate normally.

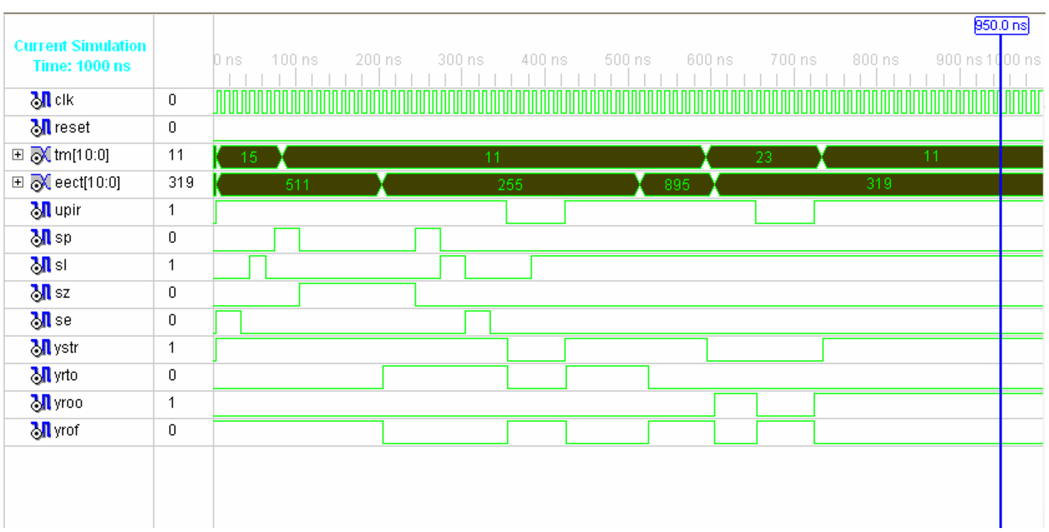


Fig. 2 Simulation of the operation of the proposed lighting control system

Conclusions and further investigations

The intelligent lighting control system for educational buildings, proposed has a great potential both for energy savings and for improving the comfort of the occupants. There are still a lot of further investigations and improvements of the system that are subject of a future work, but the fist step is made and it is successful. So far the base unit is implemented by means of logical programmable controllers (PLCs) and can function normally (previous work of our team). New parts of the proposed system are the control units for the rooms. They have not been physically realized yet, but it is only a matter of physical programming to finish this task. Of course, because of the flexibility of the FPGAs there is an intention for further complication of the algorithm proposed, as well as addition of functionalities to the control system in order to create a complete building automation system. Another task is the interconnection of the control units with the base station and real physical experiments and monitoring of the system operation.

Acknowledgements:

The results, published in this paper, are related to the topic and investigated for scientific project № 112pd055- 8, financed by the Technical University of Sofia

References:

[1]. Guidebook on Energy Efficient Electric Lighting for Buildings, Annex 45, Aalto University School of Science and Technology, ISBN 978-952-60-3229-0, ISSN 1455-7541;

[2]. **Heschong L., R. Wright, S. Okura,** Daylighting Impacts on Human Performance in School, Journal of the Illuminating Engineering Society, Summer 2002, p. 101-114;

[3]. **Pachamanov A.**, Energy Savings and Lighting Techniques (Energy efficient lighting), “Avangard- Prima”, 272 pages. ISBN 978-954-323-251-2, Sofia 2007;

[4]. **Pachamanov A., R. Pachamanov**, Method for Lighting Control in Buildings, Patent application № 106331/21.01.2002;

[5]. **Georgiev V., I. Petrinska**, Efficiency Estimation of Lighting Control Systems according to the Expenses for Lighting in Public Buildings, XII National Lighting Conference with International Participation 2007, 10-12 June 2007, Varna, Bulgaria, p.126-131;

[6]. **Sulaiman N., Z. Obaid, M. Marhaban, M. Hamidou**, Design and Implementation of FPGA-Based Systems – a Review, Australian Journal of Basic and Applied Sciences, 3(4): 3575-3596, 2009, ISSN 1991-8178, ICSInet Publication

[7]. **Chu, Pong P**., “RTL Hardware Design using VHDL”, John Wiley, 2006, ISBN 978-0-471-72092-8

[8]. **Roth, Charles H**. et al, “Digital systems design using VHDL”, CL-Engineering, 2007, ISBN 978-0534384623

[9]. **Mladenov, V., Kr. Filipova, S. Petrakieva, B. Dimov, F. Uhlmann,** Analysis of Signal Competition in Asynchronous Ultra High-Speed Digital Circuits, Przeglad Elektrotechniczny (Electrical Review), Issue 11, 2007, ISSN 0033-2097, Poland, pp. 197-200.

[10]. **Filipova Kr., I. Costov**, Design of electronic hardware using VHDL and Petri networks described with the example of Baseband and LMP layers in Bluetooth, IEEE, Catalog N 04EX830, 27th ISSE, Annual School. Lectures, ISBN 0-7803-8422-9, 2004, 13-16 May, Sofia.