

ELEVATOR SYSTEM WITH DESTINATION CONTROL AND FPGA IMPLEMENTATION OF DISPATCHING ALGORITHMS

Daniel M. Muñoz, damuz@unb.br
Carlos H. Llanos, llanos@unb.br
University of Brasília, Department of Mechanical Engineering
Grupo de Automação e Controle, GRACO
Campus Universitário Darcy Ribeiro, Zip code 70910-900 , Brasília, D.F., Brazil.
Leandro Coelho, leandro.coelho@pucpr.br
Pontificial Catholic University of Paraná, PUCPR
Industrial and Systems Engineering Graduate Program, LAS/PPGEPS
Imaculada Conceição, 1155, Zip code 80215-901, Curitiba, Paraná, PR., Brazil.
Mauricio Ayala-Rincón, ayala@unb.br
University of Brasília, Department of Computer Science
Campus Universitário Darcy Ribeiro, Zip code 70910-900 , Brasília, D.F., Brazil.

Abstract. *Elevator systems for vertical transport of passengers are implemented in modern high-rise buildings. Conventional elevator systems use up and down hall-call buttons for calling an elevator service, a group of local control systems (LCS), one per elevator, and an Elevator Group Control System (EGCS) which selects the most suitable elevator for attending a hall-call. The performance of an elevator system is measured by means of several parameters such as the power consumption, average waiting time of passengers, among others. Conventional elevator systems have critical problems, such as, uncertainty about destination calls and number of passengers waiting for an elevator, as well as, a high computational cost. Nowadays, Field Programmable Gate Arrays (FPGAs) devices have been widely used in automation systems, given that reconfigurable architectures can be effectively applied to the realization of complex systems, with good performance and flexibility, keeping the requirements of every process or products in a customized way. In order to improve the performance of the elevator system the Destination Control System (DCS) was introduced in last years. The DCS uses destination keypads in the halls allowing the controller to know the destination floor and the exact number of passengers waiting for a car. This paper proposes a distributed approach for an elevator system, which comprises a network of Destination Control Systems (one per floor), an EGCS based on Fuzzy Logic (FEGCS) running on a CPU and an RS-485 network for connecting several Local Control Systems (LCS), developed on FPGAs, each one of them implementing several dispatching algorithms. The proposed architecture take advantage of the efficient grouping of passengers provided by DCS and the flexibility provided by the FPGAs, allowing to implement, simultaneously, different scheduling strategies according to the traffic situations in the building.*

Keywords: *Elevator Group Control System, FPGA, Fuzzy Logic*

1. INTRODUCTION

Efficient elevator systems are important to improve the users comfort in modern high-rise buildings. Conventional elevator systems consist of an Elevator Group Control System (EGCS) connected by a network to several Local Control Systems (LCS) and several up and down hall buttons for calling the elevator. Both, the EGCS and the LCS are commonly implemented by microprocessors. The EGCS receives the hall-calls and selects the most suitable elevator for attending them.

The problem of finding the best solution is very hard; in fact, the elevator problem has an *NP-hard* computational complexity (Kim *et al.,* 1998). In conventional systems the schedule of the elevators is more difficult due to the uncertainties about the desired floor and the number of passengers waiting in the halls. Thus, there is no deterministic solution of the conventional elevator problem and therefore, an approximate best solution should be found.

In commercial buildings the elevator system must deal with different flux of passengers at a given time. This factor lead to the necessity of designing a flexible control having the possibility to change dynamically the control strategy (Kim *et al.,* 1998). Many studies have been carried out to solve this problem. Approaches based on fuzzy logic methods which are capable to prescribe a correct assignment to a car given a new *hall-call* are described in (Ujikara and Amano, 1994). Also, techniques based on discrete-state Markov chains and Markov decision problem formulations have been presented in Pepyne and Cassandras (1997) and Cho *et al.* (1994). Others techniques based on artificial intelligence, neural networks and genetic algorithms are presented in Kim *et al.* (1998), Siikonen and Leppala (1991), Siikonen (1997) and Fujino *et al.* (1997). Recently works use adaptive optimization algorithms, improving the performance of the EGCS (Luo *et al.,* 2008), (Tanaka *et al.,* 2005) and (Tanaka *et al.,* 2005b).

In order to avoid some uncertainties, recent elevator systems use *hall-call* buttons for selecting the desired floor in the halls, allowing the group controller to know the desired floor and having a good approximate number of the expected passengers waiting for a call, before the user boarding the car. This methodology, called Destination Control

System (DCS), provides important information that can be used for guiding passengers with the same destination to the same cars, improving the performance of the elevator service (Sorsa *et al.,* 2005).

The performance of an EGCS is measured by means of several parameters such as the power consumption of the elevator system, average waiting time of passengers, among others. A good elevator service is directly related with large energy consumption and therefore implies high functional operational cost.

Commonly, the EGCS are implemented in microprocessors devices and perform a large number of operations that require a high computational cost. Current advances in VLSI technology raised the density integration fast enough for allowing the designers to develop directly in hardware several algorithms commonly implemented on software, thus, obtaining an expressive processing speed-up. Modern Field Programmable Gate Arrays (FPGA) provide hundred of thousand of logic elements, allowing the hardware design of complex algorithms in a flexible way exploring the parallel capabilities in order to decrease the computation time by performing simultaneous calculations.

This paper proposes a flexible architecture for elevator systems comprising several DCSs, one per hall, interconnected to an EGCS based on Fuzzy Logic (FEGCS) and several Local Control Systems (LCS) implemented on FPGAs in which three dispatching algorithms are described. The FEGCS selects the most suitable elevator to attend a new *hall-call*, as well as, selects the most suitable dispatching algorithm to run in each elevator, according to the traffic situation.

2. ELEVATOR SYSTEM

Basically a conventional elevator system is composed of quadruplets <*h,c,e,g*>, where *h* is the set of hall-call buttons, *c* the set of *car-call* buttons, *e* the set of elevators and *g* the group controller. In general, the elevator group controller (EGCS) must select the more suitable elevator for a passenger(s) requirement(s) and each elevator is controlled by a Local Control System (LCS). The EGCS is connected to the *hall-call* buttons and to the LCSs through a communication network, as shown in Fig. 1.

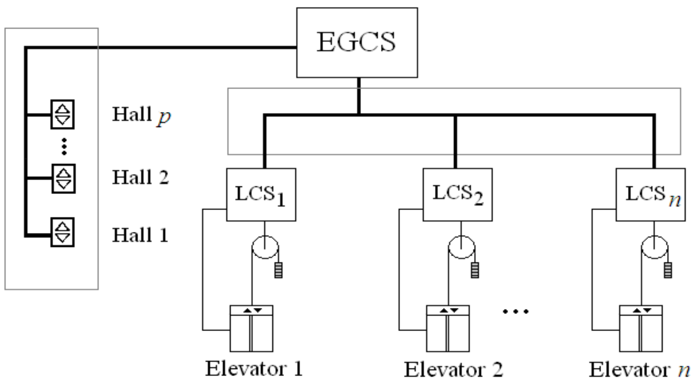

Figure 1. The conventional elevator system

The process to assign an elevator is represented by the following steps:

a) A passenger who wishes to go to the floor *x* press a *hall-call* button indicating the desired direction and stand waiting until the elevator moves to the floor where the *hall-call* occurred.

b) The *hall-call* is transmitted to the EGCS, which is currently running a given algorithm. The EGCS selects the most suitable elevator for attending the solicitation, taking into account information about the current traffic and the current state of each elevator. Different approaches for implementing the EGCS have been proposed, as will be presented in Section 3.

c) The EGCS transmits the solicitation to the selected LCS through a communication network. The LCS operates the traction engine and receives the sensors position signals from the shaft. When the current position of the elevator is the floor where the *hall-call* occurred, the elevator stops and the door is opened.

d) The user enters into the car and press the desired floor *x.* The new solicitation is registered by the LCS, closing the door and then it operates the traction engine moving the elevator into the desired direction. Finally, when the floor *x* is reached the elevator stops and the door is opened.

The process to determine the best elevator for attending a *hall-call* is more difficult when using conventional systems, given that uncertainties about the desired floor and the number of passengers waiting in the halls are presented. This problem is solved by using a Destination Control Systems (DCS), in which the *car-calls* are eliminated and then each hall in the building has a button panel for selecting the desired floor before the user boards the car (Sorsa *et al.,* 2005). Elevator systems using DCS works as follow.

a) A passenger in hall *y* who wishes to go to the floor *x* press the respective desired floor button from a panel. The new *hall-call* composed by a duple <*y,x*> is transmitted through a network to the EGCS.

b) The EGCS processes the new *hall-call* <*y,x*> taking into account the current traffic, the number of passengers in the hall and the desired floor. The EGCS, which runs a given algorithm, selects the suitable elevator sending the respective number of that elevator through the network, displaying it in the panel. Thus, the passenger knows which one of the elevators must wait in order to improve the service.

c) The EGCS transmits the solicitation to the selected LCS through a communication network. The LCS operates the traction engine and receives the sensors position signals into the shaft. When the current position of the elevator is the floor where the *hall-call* occurred (y), the elevator stops and the door is opened.

d) Users waiting for the same elevator enter into the car and then the LCS closes the door. Afterward, the traction engine is operated moving the elevator in the desired direction. Finally, when the floor x is reached the elevator stops and the door is opened.

As explained above the DCS allows the EGCS to know previous information about the number of passengers waiting for an elevator and their desired floors. These facts avoid some uncertainties of the conventional systems, allowing the EGCS to calculate more exactly the estimated waiting time of the passengers in order to perform suitable control strategies, identifying more effectively the traffic patterns in the building and guiding passengers with the same destination floors to the same cars.

3. RELATED WORKS

Modern elevator systems must improve the traffic handling capacity. Many studies have been done regarding methodologies and algorithms for scheduling the elevators in heavy traffic situations. Recent trends in EGCS involve not only new control strategies, but also new structural changes such as the use of double-deck or multi-cars in the same shaft and the use of Destination Control Systems (DCS), (Markon *et al.,* 2008).

Double-deck elevator systems use two cars stacked on each other and drive them by the same traction engine, increasing the transportation and saving space occupied by the elevators. Sorsa *et al.* (2005) discuss the effect of the DCS to Double-deck elevators using a genetic algorithm controller, resulting in a high handling capacity. Recently, Zhou *et al*. (2007) and Yu *et al*. (2007) describe a Double-deck elevator using Genetic Programing with learning reinforcement and Ant Colony Optimization, respectively. Both approaches demonstrate the effectiveness of the Double-deck for different traffic situations.

A Multi-car elevator system has two or more cars in the same shaft. This system promises a significant improvement of the transportation in high-rise buildings, reducing the space occupied by elevators in the building. Ikeda *et al.* (2007) and Kuroda *et al.* (2008) present different algorithms for controlling multi-car elevator systems, showing that multi-car system performs better than conventional systems; however, new control strategies are needed and the scheduling of the elevators becomes more difficult for heavy traffic situations.

Although, the Double-deck and Multi-car elevator systems promises a dramatic improvement in the handling capacity, the most commercial elevator system is the single elevator with DCS, avoiding complex control algorithms and mechanical modifications in the structure of the shaft. Several contributions for improving the performance of elevator systems using DCS have been made. Beielstein *et al.* (2003) presents an elevator system with DCS using a group controller with evolution strategies and Sorsa *et al.* (2005) demonstrates that in up-peak traffic situations the handling capacity is increased over 20-30% by using control systems with DCS. Tanaka *et al.* (2005) and (2005b) present a dynamic optimization solution for elevator systems using DCS, showing improvements in the handling capacity in comparison with traditional systems and algorithms.

Fuzzy logic methodologies have been studied for identifying traffic patterns in order to design control strategies in a dynamic approach according to the current traffic situation. In Siikonen (1997) and Kim *et al.* (1998), fuzzy systems were applied in a multi-objective control, in which the users comfort and the power consumption are important factors that should be improved. In the Siikonen model, a Fuzzy Elevator Group Control System (FEGCS) implements a forecasting method and a traffic pattern recognition. Kim *et al.* (1998) implements two FEGCS, the first one uses traffic characteristics for identifying traffic patterns and the second one uses the Average Waiting Time (AWT) of passengers as evaluation criteria for selecting the most suitable elevator. Fuzzy Neural Network (FNN) for optimizing multiple objectives in order to implement the EGCS with DCS is presented in Tai *et al.* (2008).

In summary, most of the previous works describe elevator systems in which the ECGS, implemented by microprocessors, must performs a large number of operations for running the control algorithm, limiting the flexibility of the system for performing, simultaneously, different strategies. FPGA implementations of dispatching algorithms could provide more flexibility, allowing the elevator system to reconfigure dynamically different scheduling strategies according to the traffic situations in the building.

In our previous works, FPGA implementations of dispatching algorithms were presented. Muñoz *et al.* (2008) and (2008b) describe a novel architecture for a single-car elevator system using the traditional up and down *hall-call* buttons. The architecture consists of a distributed controller in which the Local Control Systems (LCSs), implemented on FPGAs, perform several dispatching algorithms, each one of them is suitable in a specific traffic situation. A FEGCS, running on a CPU, identifies the traffic patterns, selects the most suitable elevator for attending each *hall-call* and selects the most suitable algorithm to be executed in each elevator.

In this work we use the same distributed approach presented in Muñoz *et al.* (2008) to implement an elevator system; however, the architecture considers the use of DCSs. The new approach takes into account not only the traffic patterns, but also the information about the desired floor and the load capacity of each elevator. The dispatcher of the elevators was performed effectively by using DCS, given that data provided from *hall-calls* and elevators allow for a better and easier organization than with conventional elevator systems. The FPGA implementation of dispatching algorithms provides high flexibility given that different strategies can be setting-up for specific traffic situations and new complex algorithms can be easily implemented by reconfiguring the architecture. Also, the FPGA implementation allows the LCS to perform simultaneous computations, increasing the performance when using high-speed elevators and, simultaneously, reducing the number of calculations required by the group controller.

4. IMPLEMENTATIONS

In this section the proposed elevator system architecture and its main components are presented. The architecture considers the use of Destination Control Systems (DCS) and a group control algorithm based on fuzzy logic. The Local Control Systems (LCSs) were implemented on FPGAs, performing three dispatching algorithms that were described using the Very High Description Language (VHDL).

The overall elevator system architecture for p halls and n elevators is shown in Fig. 1. It consist of a set of DCS connected to an RS-485 communication network. In the same network a set of LCSs are connected, each one of them implemented on an FPGA, in which are described, directly in hardware, different dispatching algorithms. All this components (DCSs and LCSs) are connected to the network using slaves nodes that allow to convert the RS-485 standard to an RS-232 standard. A FEGCS, running on a CPU, take the advantages of the use of DCSs performing two main process: (1) identification of traffic patterns in order to select the most suitable algorithm to run in each elevator and (2) dispatching the elevators taking into account several variables that will be described below.

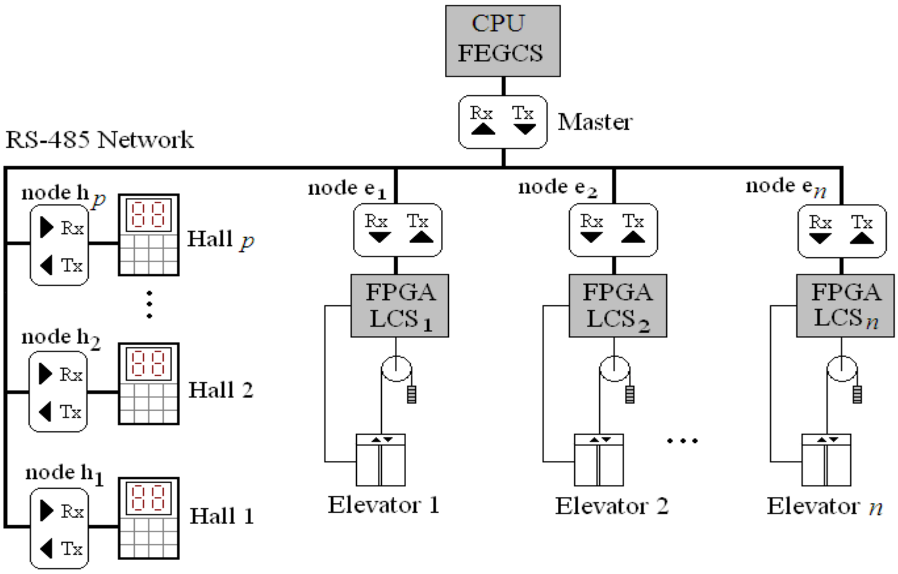

Figure 2. The proposed architecture

The RS-485 network allows up to 32 devices to communicate at distances up to 1200 meters. The number of devices (nodes) of the network can easily be extended using repeater modules. Therefore, the proposed architecture is flexible in terms of scalability, allowing to set up elevator systems with a large number of floors and elevators. Notice that this is a novel architecture in the following aspects:

a) The FPGA implementation of the LCSs improves the flexibility of the elevator system, given that the dispatching algorithms can be actualized in real time. This aspect allows the elevators to perform, simultaneously, different algorithms according to the traffic situations in the building.

b) The distributed control approach proposed in this architecture allows the FECGS to decrease the number of computations required for implementing the group controller. It can be explained given that the FEGCS schedules the elevators by choosing the most suitable LCS to attend a new *hall-call* or group of new *hall-calls*; however, each LCS performs the final control, according to the selected algorithm, choosing the order in which the *hall-calls* will be attended.

c) By accomplishing this architecture, the elevator system keeps on working when the FEGCS is out of service. In this case, each LCS will work according to the latest algorithm selected by the FEGCS.

4.1 Fuzzy Elevator Group Control System

The group controller proposed in this architecture is based on fuzzy logic and was implemented in order to identify traffic patterns and to select the more suitable elevator to attend the *hall-calls*. The FEGCS is composed of three main modules, as follow: (1) *traffic identifier module*, (2) *dispatcher module* and (3) *mapping module*. The *traffic identifier module* is executed every 15 minutes, up-loading the current traffic situation in the building. The *dispatcher module* and the *mapping module* are executed whenever a new building event occurs. Figure 3 presents the proposed FEGCS, that was accomplished by using the Xfuzzy tool and synthesized in Java language (Xfuzzy 2003).

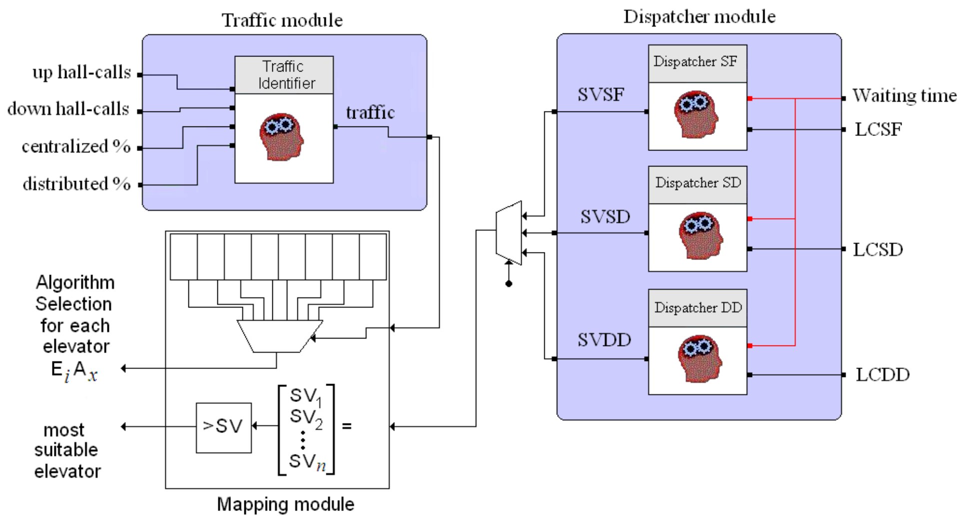

Figure 3. The Fuzzy Elevator Group Control System

4.1.1 The traffic module

The *traffic module* was implemented taking into account the model proposed by Kim *et al.* (1998). This module receives four input variables that represents traffic characteristics in commercial buildings and identifies the current traffic situation, selecting one among five possible traffic patterns. The input linguistic variables of this module are shown in Tab. 1 and are calculated taking into account the information provided by the DCSs. Table 2 describes the traffic patters used in this work. Each one of them presents different characteristics, traffic intensities and occurs at different time in commercial buildings.

Table 1. Input variables of the Traffic Module

Input variable	Description
Up hall-calls	Number of *hall-calls* from the lobby to others floors
Down hall-calls	Number of *hall-calls* from all floors to the lobby
Centralized incoming %	Ratio of the number of *hall-calls* with the same desired floor to that of all *hall-calls*. It is a value percentage of diversity of the desired floor of the passengers.
Distributed outgoing %	Radio of the number of *hall-calls* from different floors, except the most crowded floor, to that of all *hall-calls*. It is a percentage value of the floors diversity which the *hall-calls* are solicited.

Table 2. Traffic patterns in commercial buildings

Traffic pattern	Description
Up peak	Early in the morning the passengers come into a building beginning the office activities. Traffic intensity is large
Business time	Before and after noon, majority inter floors-calls. Total traffic intensity is medium
Heavy traffic	Anytime, inter floor-calls, up and down hall-calls. Total traffic intensity is large
Down peak	Passengers go out from the building closing the office activities. Traffic intensity is medium/large
Inactive time	Night, total traffic is small

4.1.2. The dispatcher module

This module was implemented in order to select the most suitable elevator for attending a *hall-call*. This module receives linguistic variables related to the waiting time and the load capacity of each elevator and must be executed n times (n elevators) in order to find a suitable value (*SV*) for each elevator. The elevator with the larger *SV* is selected for attending a *hall-call*.

Waiting time of passengers and handling capacity are important variables for implementing group controllers, being directly related to comfort and performance of the elevator system. Table 3 describes the input variables required for computing the suitable value of the elevators.

Table 3. Input variables of the Dispatcher module

Input variable	Description
Waiting time	Estimated value of the total waiting time of the passengers. (waiting time in halls plus waiting time into the cars)
Load capacity same floor (LCSF)	Number of vacancies available in the elevator when it reached the floor from which the *hall-call* was solicited in case of the desired floor has been previously assigned to the elevator.
Load capacity different floor same direction (LCSD)	Number of vacancies available in the elevator when it reached the floor from which the *hall-call* was solicited in case that the desired floor is different from the previously assigned to the elevator and has the same direction than the current movement direction.
Load capacity different floor different direction (LCDD)	Number of vacancies available in the elevator when it reached the floor from which the *hall-call* was solicited in case that the desired floor is different from the previously assigned to the elevator and has different direction that the current movement direction.

The input variables of the *dispatcher module* can be calculated according to several building events stored in registers. Equation (1) illustrates the matrix *NP* of calls for a building whit p floors. Rows (indexed by h_i) represent the floor from which a *hall-call* was given; columns (indexed by f_j) represent the desired floor of the *hall-call* and the value stored, *NP(i,j),* represents the number of passengers waiting in the hall i going to the floor j. From this matrix, the flux of passengers in the building and the input variables of the *traffic module* can be computed.

$$\begin{array}{llllll} \langle h|f\rangle & f_1 & f_2 & f_3 & \cdots & f_p \\ h_1 & 0 & \square & \square & \cdots & 3 \\ h_2 & \square & 0 & \square & \cdots & \square \\ h_3 & \square & 1 & 0 & \cdots & \square \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ h_p & 10 & \square & \square & \cdots & 0 \end{array} \tag{1}$$

Other important values related to the building events are the stop number and the load capacity of each elevator. Equation (2) shows an example of the programed calls register for a building with p floors and n elevators, in which the elevator 2 has been programed for stopping in the second, third and p^{th} floors. Notice that the order in which these floors are attended is not shown in the equation. Equation (3) shows the load capacity register of the elevators, in which nine vacancies are available for elevator 1 and two vacancies are available for elevator 2. The values stored in Eq. (1), (2) and (3) are actualized when a new *hall-call* is requested or a *hall-call* is attended.

$$\begin{array}{llllll} \langle E|f\rangle & f_1 & f_2 & f_3 & \cdots & f_p \\ E_1 & \square & \square & \square & \cdots & \square \\ E_2 & 0 & 1 & 1 & \cdots & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ E_n & \square & \square & \square & \cdots & \square \end{array} \tag{2}$$

$$\begin{array}{lllll} \mathit{Elevator} & E_1 & E_2 & \cdots & E_n \\ \mathit{loading\,capacity} & 9 & 2 & \cdots & \square \end{array} \tag{3}$$

The *dispatcher module* is composed of three fuzzy systems, each one of them is activated depending on the direction of the *hall-call* requested with respect to the current direction of movement of each elevator.

a) *Dispatcher SF*: receives the estimated waiting time (WT) and the load capacity (LCSF, see Tab. 3) as inputs and returns the Suitable Value (SV) of an elevator when the desired floor of the requested *hall-call* have been previously stored in the matrix of calls ("1" value in the (*i,j*) position of Eq. 2).

b) *Dispatcher SD*: receives the estimated waiting time (WT) and the load capacity (LCSD, see Tab. 3) as inputs and returns the suitable value (SV) of an elevator when the desired floor of the requested *hall-call* has not been stored previously in the elevator ("0" value in the (*i,j*) position of Eq. 2); however the direction of the requested *hall-call* is the same than the current direction of movement of the elevator.

c) *Dispatcher DD*: receives the estimated waiting time (WT) and the load capacity (LCDD, see Tab. 3) as inputs and returns the suitable value (SV) of an elevator when the desired floor of the requested *hall-call* has not been stored previously in the elevator ("0" value in the (*i,j*) position of Eq. 2) and the direction of the requested *hall-call* is the opposite to the current movement direction of the elevator.

Equations (4) to (6) show the estimation of the overall waiting time (WT). This variable is computed taking into account the hall waiting time and the trip waiting time. *SN* represents the stop number programmed for each elevator, *DN* is the number of driving floors and *NP(i)* is the expected number of passengers boarding or leaving the elevator in each stop.

$$WT = drive_{time} + stop_{time} \tag{4}$$

$$drive_{time} = DN \cdot nominalspeed_{time} + (SN+1) \cdot speedup_{time} + (SN+1) \cdot speeddown_{time} \tag{5}$$

$$stop_{time} = \sum_{i=1}^{SN} NP(i) \cdot transfer_{time} + SN \cdot openingdoor_{time} + SN \cdot closingdoor_{time} \tag{6}$$

The load capacity (L*C*) of each elevator is calculated every time a new *hall-call* occurs. Equation (7) shows the LC computation, where TLC is the total load capacity of the elevator and *currentNP* represents the current number of passengers in the elevator. A negative value of *LC* means that the load capacity is not enough when the elevator reaches the floor from which the *hall-call* was solicited.

$$LC = TLC - [currentNP + \sum_{i=1}^{SN} NP_{inc}(i) - \sum_{i=1}^{SN} NP_{out}(i)] \tag{7}$$

Table 4 shows the rules of knowledge of the fuzzy *dispatcher module.* Notice that there are three fuzzy systems, each one of them using nine rules; however, only one fuzzy system is activated at the same time, according to the direction of the *hall-call* with respect the movement direction of the elevator.

Table 4. Knowledge rules of the *dispatcher module*

Waiting time (sec)	LCSF			LCSD			LCDD		
	negative	**zero**	**positive**	**negative**	**zero**	**positive**	**negative**	**zero**	**positive**
small	medium	vlarge	vlarge	medium	large	vlarge	medium	large	vlarge
medium	small	large	medium	small	medium	large	vsmall	medium	small
large	vsmall	medium	small	vsmall	small	medium	vsmall	small	vsmall

4.1.3. The mapping module

The *mapping module* receives as input the decisions from the *traffic module* and the *dispatcher module,* and according to the current traffic situation selects one among three algorithms implemented in the LCSs. Also, the *mapping module* selects the elevator with the largest suitable value in order to attend a new *hall-call*.

5. RESULTS

The Local Control System (LCS) architecture has been described in VHDL using the Xilinx ISE 10.1 development tool. All the algorithms are based on Finite State Machines (FMS) and are parameterizable by number of floors, allowing the LCS to be setting-up for different commercial buildings.

5.1 Local Control System

Figure 4 shows the hardware architecture of the LCS. Several communication components can be observed in which the action commands from the FEGCS are received and interpreted. When the elevator attends a *hall-call*, these communication components create and send a state command composed of the current position and the current movement direction of the elevator.

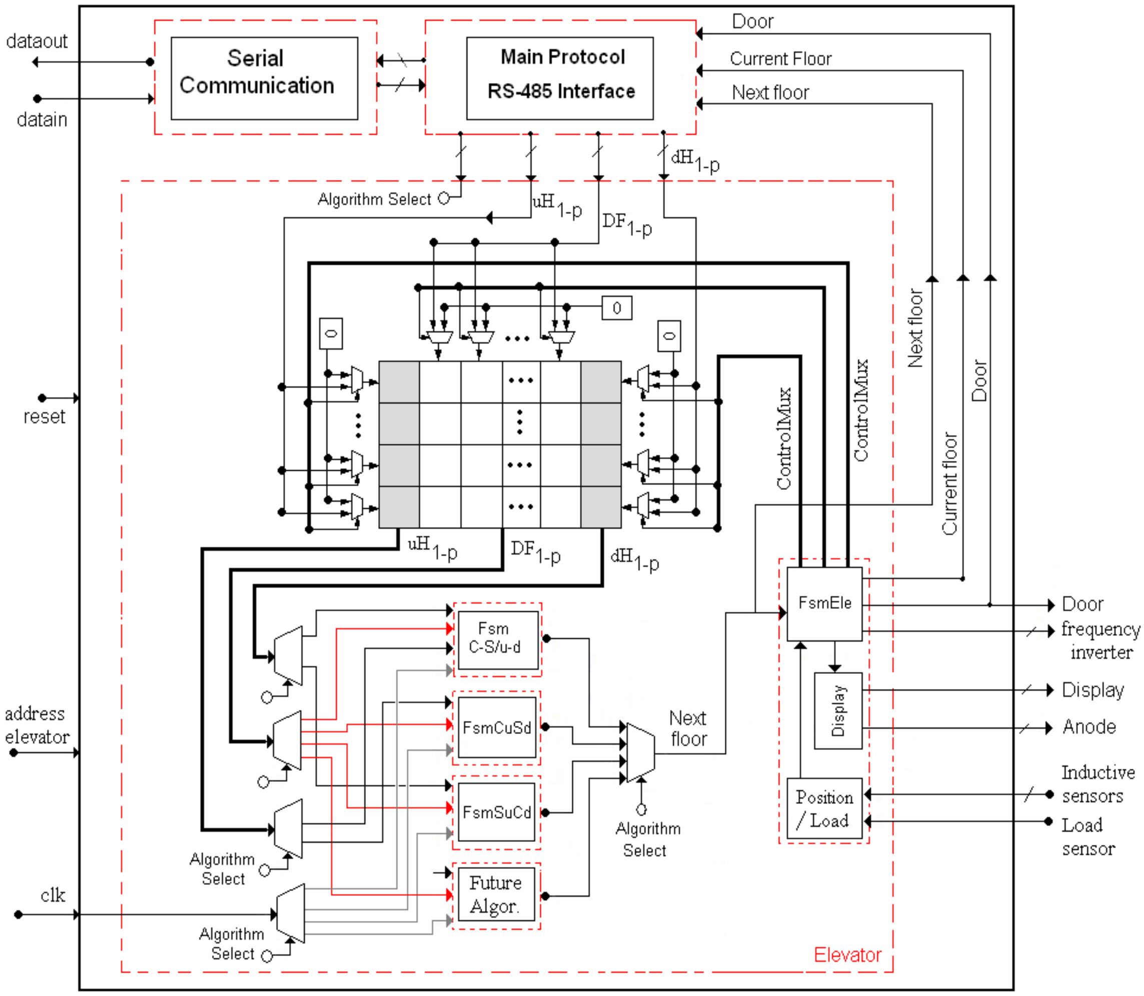

Figure 4. The FPGA LCS architecture

The LCS implements three dispatching algorithms, each one of them is suitable for different traffic situations. These algorithms select the next floor to be visited by searching the *hall-calls* from three registers. The *uH-register* stores *hall-calls* in the up direction, the *dH-register* stores the *hall-calls* in the down direction and the *DF-register* stores the desired floors. The dispatching algorithms were described using a Finite State Machine (FSM) approach.

The next floor to be visited is sent to the *Fsm_Ele* component which implements the interface with the elevator. The *Fsm_ele* controls several instrumentation, such as, the frequency inverter of the motor engine, the elevator door, displays and also receives signals from the inductive sensors in the shaft and sensors in the elevator in order to estimate the current position and number of passengers in the elevator, respectively.

Notice that, on the one hand, this architecture provides flexibility by executing, simultaneously, different strategies in order to attend the *hall-calls*. On the other hand, the number of computations in the FEGCS are decreased, given that the order in which the calls must be attended is computed by the LCS, according to the current algorithm selected.

5.2 The Dispatching Algorithms implemented in hardware

As shown in Fig. 4 each LCS allows to setting-up the logic of control by choosing one of the dispatching algorithms implemented in the FPGA. The algorithms are selected according to the traffic patterns (see Tab. 2) identified by the *traffic module* and may be present at different times, suggesting the use of different strategies for attending the *hall-calls*. The implemented algorithms are based on the collective principle and the selective principle, as explained in Muñoz *et al.* 2008. Figure 5 shows the behavior of the implemented algorithms and the desired traffic situation for each one of them. The situations presented in Fig. 5 are used by the *mapping module* (see Section 4.2.3) in order to select the algorithms to run in each elevator. For example, when an up peak traffic is presented it could be useful to execute a collective approach during the up trip, leaving the passengers in their desired floors, and then when the elevator has visited the highest desired floor, a selective approach is executed during the down trip in order to go expressly to the lobby (see Fig. 5b). In the same way, the selective up / collective down algorithm is selected when the traffic pattern is the down peak (Fig. 5c). Notice that multiple traffic patterns are presented and different algorithms can be executed simultaneously increasing performance of the elevator system.

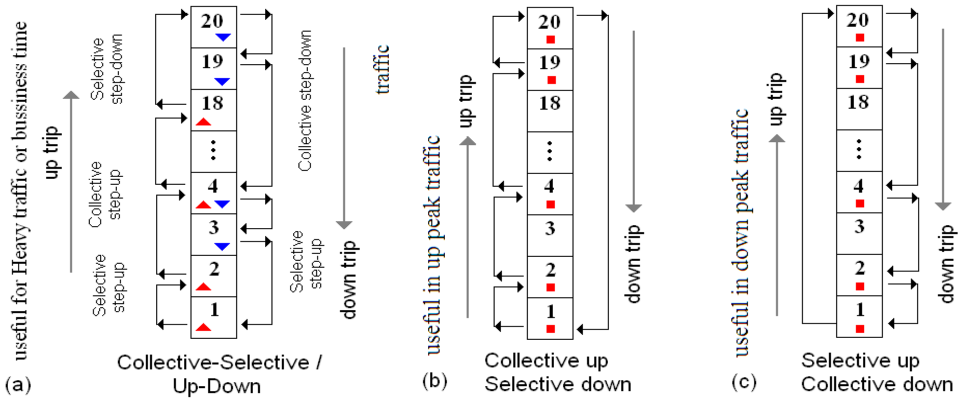

Figure 5. Dispatching algorithms implemented in hardware

Figure 6 shows the synthesis results of the main components of the LCS. Some important parameters in the implementation are the FPGA resources consumption (Slices and Look up Tables-LUTs) and the performance (Frequency) of the proposed circuits. This results demonstrate the correctness and effectiveness of the hardware implementation of the dispatching algorithms.

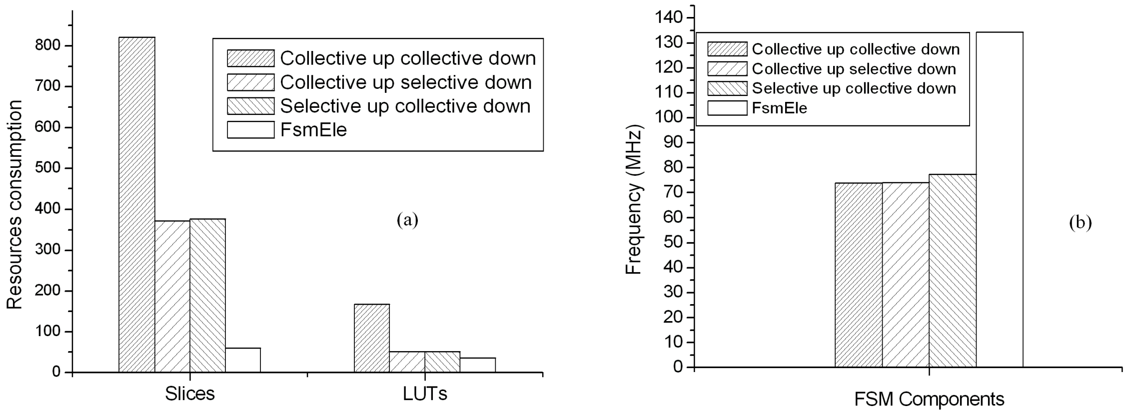

Figure 6. Synthesis results for Spatan3E FPGA

6. CONCLUSIONS

In this work, an architecture for implementing elevator systems using destination control systems (DCS), a distributed controller based on fuzzy logic and FPGA implementation of dispatching algorithms was presented. The Fuzzy Elevator Group Control System (FEGCS) runs on a CPU, receives the *hall-calls* from the DCS and selects the most suitable elevator to attend each *hall-call*. Also, the FEGCS identities traffic patterns in the building, using the information provided by the DCS, in order to select the most suitable dispatching algorithm to run in each Local Control System (LCS), from the different algorithms that were implemented on the FPGAs.

The proposed architecture takes advantage of the efficient grouping of passengers provided by the DCS and the flexibility provided by the FPGA, allowing to implement, simultaneously, different scheduling strategies according to the traffic situations in the building. The use of DCS avoids some uncertainties presented in conventional elevator systems allowing the group controller to perform more effectively and efficiently than with conventional elevator systems, given that relevant knowledge, such as, the overall waiting time of passengers, the expected number of passengers waiting in the halls and the expected number of passengers going to the same floor can be estimated with more accuracy than with conventional systems.

As future works we pretend to validate the FEGCS for different traffic situations using a hardware-in-the-loop simulation in which the LCSs, implemented on the FPGAs, are used as a model of an elevator system. Also, we pretend to explore the capabilities of population based optimization algorithms in the EGCS implementation. Evolutionary methodologies for implementing the EGCS can be justified given that the topology of possible fitness functions for the elevator problem are highly non-linear, multimodal and they change dynamically with the traffic situations.

7. REFERENCES

Beielstein, T., Ewald, C-P., Markon, S., 2003, “Optimal Elevator Group Control by Evolution Strategies”, Proceedings of the International Conference on Genetic and Evolutionary Computation, Chicago, USA, pp. 1063-1974.

Cho, Y., Gagov, Z., Kwon, W., 1999, “Elevator Group Control with Accurate Estimation of hall-call Waiting Times”, IEEE Proceedings of International Conference on Robotics and Automation, Vol.1, Detroit, USA, pp. 447-452.

Fujino, A., Tobita, T., Segawa, K., Yoneda K., Togawa A., 1997, “An Elevator Group Control System with Floor-Attribute Control Method and System Optimization using Genetic Algorithms”, IEEE Transactions on Industrial Electronics, Vol.44, No 4, pp. 546-552.

Ikeda, K., Suzuki, H., Markon, S., Kita, H., 2007, “Traffic-sensitive Controllers for Multi-Car Elevators; Design, Multi-objetive Optimization and Analysis”, Proceedings of the Annual Conference of the Society of Instrument and Control Engineers, Kagawa, Japan, pp. 2655-2662.

Kim, C., Seong, K.A., Lee-Kwang, H., Kim J.O., 1998, “Design and Implementation of a Fuzzy Elevator Group Control System”. IEEE Transactions on Systems Man and Cybernetics Part A, Vol.28, No. 3, pp. 277-287.

Kuroda, Y., Nakata, M., Yamagushi, M., 2008, “A Problem and Its Solution for Multi-Car Elevator Group Control”, Proceedings of the International Conference on Circuits/Systems, Computers and Communications, Yamaguchi, Japan, pp. 693-396.

Luo, F., Lin, X., Xu. Y., Li, H., 2008, “Hybrid Elevator Group Control System Based on Immune Particle Swarm Hybrid Optimization Algorithm with Full Digital Keypads”, IEEE Proceedings of the International Congress on Intelligent Control and Automation, Chongqing, China, pp. 1482-1487.

Markon, S., Aoki, K., Nakawaga, M., Sudo T., 2008, “Recent Trends in Elevator Group Control Systems”, Proceedings of the International Conference on Circuits/Systems, Computers and Communications, Yamaguchi, Japan, pp. 697-700.

Muñoz, D., Llanos, C., Ayala-Rincón, M., van Els, R., 2008, “Distributed Approach to Elevator Group Control Systems using Fuzzy Logic and FPGA Implementation of Dispatching Algorithms”, Journal of Engineering Applications of Artificial Intelligence, Vol.21, pp. 1309-1320.

Muñoz, D., Llanos, C., Ayala-Rincón, M., van Els, R., 2008, “FPGA Implementation of Dispatching Algorithms for Local Control of Elevator Systems”, IEEE Proceedings of the International Symposium on Industrial Electronics, Vol.21, Ouro Preto, Brazil, pp. 1997-2002.

Pepyne, D. and Cassandras C., 1997, “Optimal Dispatching Control for Elevator Systems during Up-peak Traffic” IEEE Transactions on Control System Technology, Vol.5, No 6, pp. 629-643.

Siikonen, M.-L. And Leppala, J., 1991, “Elevator Traffic Pattern Recognition”, Proceedings of International Congress of the Fuzzy Systems Association, Brussels, Belgium, pp. 195-198.

Siikonen, M.-L., 1997, “Elevator Group Control with Artificial Intelligence”, Private Communication, Helsinki University of Technology, Research Reports A67, Finland, 32p.

Sorsa, J., Hakonen, H., Siikonen M-L., 2005, “Elevator Selection with Destination Control System”, Elevator World, Vol.54, No. 1, pp. 148-155.

Tai, J., Yang, S., Yan, H., 2008, “Dispatching Approach Optimization of Elevator Group Control System with Destination Floor Guidance using Fuzzy Neural Network”, Proceedings of the International Congress on Intelligent Control and Automation, Chongqing, China, pp. 7085-7088.

Tanaka, S., Uraguchi, Y., Araki, M., 2005, “Dynamic Optimization of the Operation of Single-car Elevator Systems with Destination Hall call Registration: Part I. Formulation and Simulations”, European Journal of Operational Research, Vol.167, pp. 550-573.

Tanaka, S., Uraguchi, Y., Araki, M., 2005, “Dynamic Optimization of the Operation of Single-car Elevator Systems with Destination Hall call Registration: Part II. The Solution Algorithm”, European Journal of Operational Research, Vol.167, pp. 574-587.

Ujira, H. and Amano, M., 1994, “The Latest Elevator Group-control System”, Mitsubishi Electric Advance, Vol.67, pp.10-12.

Xfuzzy3.0 User Guide, 2003, “Fuzzy Logic Design Tools”, 21 August 2009, <http://www.imse.cnm.es/Xfuzzy>.

Yu, L., Zhou, J., Mabu, S., Hirasawa, K., Hu, J., Markon, S., 2007, “Double-deck Elevator Croup Supervisory Control System using Genetic Network Programming Ant Colony Optimization with Evaporation”, Journal of Advanced Computational Intelligence and Intelligent Informatics, Vol.11, No 9, pp. 1149-1148.

Zhou, J., Yu, L., Mabu, S., Hirasawa, K., Hu, J., Markon, S., 2007, “Double-deck Elevator System using Genetic Network Programming with Reinforcement Learning”, IEEE Proceedings of the International Conference on Evolutionary Computation, Singapore, pp. 2025-2031.

8. RESPONSIBILITY NOTICE

The authors are the only responsible for the printed material included in this paper.