ISSN 1392 - 1215 ELEKTRONIKA IR ELEKTROTECHNIKA. 2004. Nr. 4(53)

T 115 MEDICINOS TECHNOLOGIJA

Home Environment Control System for the Disabled: User Interface Testing

P. Serafinavičius, V. Lauruška
Department of Electronics, Šiauliai University,
141 Vilniaus Str., Šiauliai, Lithuania, tel. +370 41 595835, e-mail: paulius_s@splius.lt

Introduction

Home environment control and automation system can be useful not only for healthy people. Recently more attention is paid to assistive technologies which help to integrate disabled people. Specially developed systems enabling communication and living environment control for the disabled can be examples of such technologies.

Disabled people often have some difficulties using traditional home environment controls or they are unable to use it at all. Home environment control system can help them to increase the quality of their life. This allows them to feel more independent. The developed system is designed for severe physically disabled people having speech and language impairments. It can be used to make more accessible environment control for the elderly.

The majority of similar systems are expensive and/or not enough adaptable for persons with different types of disabilities. It results in decreased system functionality and usability. The technologies used in designed system are the same as have been used for healthy people. Only the interface should be specialized for the disabled.

Hardware of the system

We used a technology called Smart Home (or Domotics) in the system. This technology is based on a network which has been available more then 10 years for heating, ventilation and air conditioning (HVAC) applications. Smart Home technology is dedicated for command, control and supervision home environment. Mostly it includes safety alarms (smoke, fire, gas), HVAC but excludes telecommunication and entertainment events. Smart home devices are connected to a network. The type of the network is different from a telecommunication's network.

Several types of Smart Home's network standards such as X10, LonWorks, BatiBus, EHS and EIB are available. They are incompatible in most of the cases. So there was necessary to select one of them as a basis for the system we developed. We selected EIB (European Installation Bus Association) due to its advantages [2] and it is being used for distribution of control signals in home environment.

The proposed system allows the integration of the functions that a disabled user should have an access too. This includes not only home environment control (lighting, heating, shutters, motorized door and windows) but also entertainment equipments (TV, VCR, DVD and Audio systems) and security (alarm for detection of intruders, safety alarms). A universal infrared transmitter has been used for entertainment equipment control. A PC has been used for the user interface implementation. General block diagram and more detailed description of the system can be found in [1].

Software of the system and the user interface

The principle of user interface of the system is based on scanning OS MS Windows (9x, NT, 2000, XP) control elements (mostly pushbuttons) on a dialog-based menu. A pushbutton is active for a short timeout called activation time. User can press a key to execute an associated command related with to this pushbutton. This period is can be set up programmably. The next pushbutton is focused if there is no control signal from the user, etc. The focus is made by setting the focus rectangle to the object using tabulation key index sequence. The user activates a focused pushbutton by an ON-OFF switch. It is a separate key which is connected to the standard keyboard parallel to the "space" key, or modified keyboard with this one key only. The main function of the user interface is to scan Windows menus with control options. When the user presses the switch the appropriate data is being sent to EIB network or infrared transmitter depending on the selected option command. There have been used EIB Falcon run-time components to enable the full bus access from a Windows application. When EIB device gets a data telegram, it executes an action given by the telegram and after has finished it sends a response telegram to the PC which tells was the action successful or not. According to the response telegram PC displays an appropriate message to the user. So the user is always informed about the current state of each controlled device. In the same way he could know if another user has controlled any device on EIB network.

The user interface can be implemented in two or three-levels dialog structure with pushbuttons: The 1^{st} level – selection one of all available rooms in a flat or house; the 2^{nd} level – selection one of all available controlled objects (e. g. lights, door, window, etc) in the selected room; the

3rd level – selection one of all available commands (e. g. open, close, switch on, off, etc) in the selected object. (e. g. room 1 -> window -> open). The principle algorithm of the users interface is shown in figure 1. It represents only the 1st level – room selection because other two levels – object and command selection look similar. We have implemented in user interface prototype n=10. That means that there are available for control no more than 10 rooms, 10 objects in each room and 10 commands for each object. It is enough in most of the cases. We can enlarge each menu level without destroying the current structure.

The scanning process begins when user pushes the "Start" button. The button is active for a timeout t_a. If user presses the button while it is in active state (not later than after t_a) then the scanning process is stopped. Otherwise the "Room 1" pushbutton becomes focused and if it has been selected the objects menu in "Room 1" appears on the screen. If it hasn't been selected then focuses the next one and so on. Of course this method of user interface using only one input key and objects scanning is slow enough but this is the simplest way of implementation and no specific, expensive hardware is needed for it. We must to make some usability metrics according to ISO 9241-11.2 committee draft to make the interface more efficient [3]. Activation time t_a evaluation is one of the most important measurements in this design. The activation time must be set such value that it shouldn't slowdown task completion process too much and the user should be able to execute a required command in time.

Activation time experimental evaluation

One of the most powerful methods of evaluating a design is to use a controlled experiment [3]. Using this method has been evaluated the activation time t_a. For this is needed to select a task. The same task will be used for testing with different users. When choosing a task for this evaluation it is reasonable to choose the task which takes the shortest time to complete it. In our case the task is to open the door in the first room (fig. 2). This task consists of three user actions: selection the first room from the rooms menu, selection the door from the controllable objects in this room and selection the command to open the door from the commands menu for the selected object. Each action is executable by a pushbutton which stands in the first position. That means a time to complete the task t_c can be expressed in an ideal conditions:

$$t_c = kt_a + mt_r; \qquad (1)$$

where t_r = 0.2 s – human reaction time to a visual signal [3]; k = 1 - the number of steps needed to execute specified task; m = 3 - the number of menu levels; and t_c = 1 sec when t_a = 0.4s for the given task.

A thin arrow in the fig. 2 shows the direction of the next activated pushbutton after activation time has been passed. A bold rectangle shows the executed option and a bold arrow shows the next menu level is opened.

This task as described above has been tested with 5 users. One of them is physically disabled – can't walk and have difficulties to move his hands.

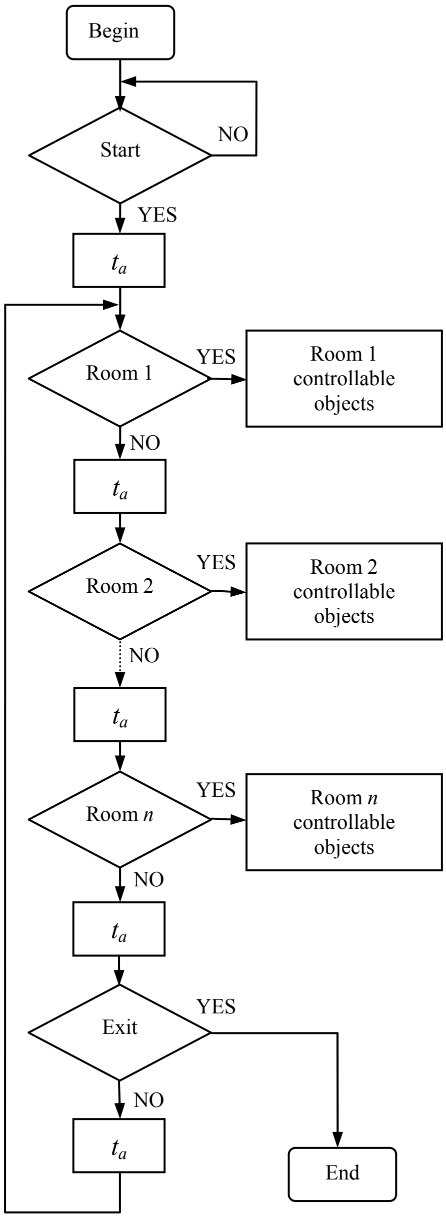


Fig. 1. Principle algorithm of user interface. t_a – activation time, n – maximum number of pushbuttons in the same level menu level not including "Start" and "Exit" buttons

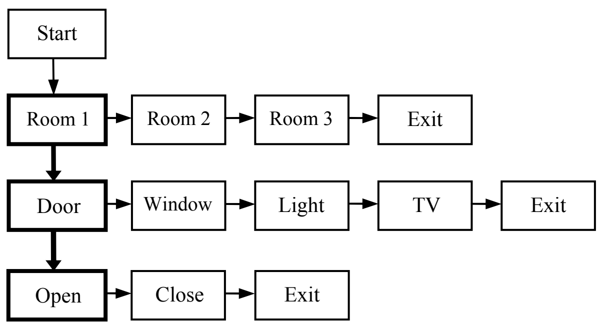


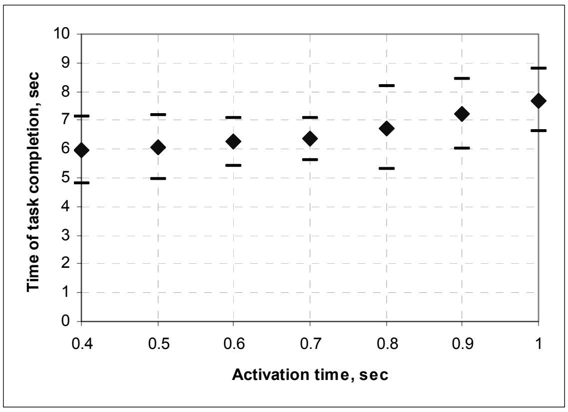
Fig. 2. The task completion diagram

Each user was familiarized and trained to work with this interface before the test. Each user was tested 5 times with the same activation time. For this evaluation was specially designed the interface program using Microsoft

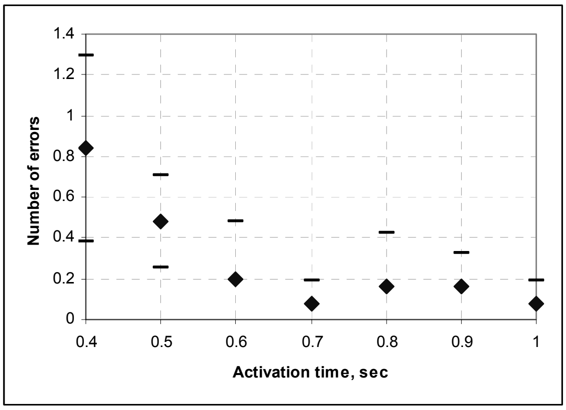
Visual C++. It logs to a file all the user actions and the time which has been passed since pressing the “Start” button. The time to complete the task and the number of errors the user has made has been determined by using this log data sequence. Measurements are done with different activation times which were taken 0.4, 0.5, 0.6, 0.7, 0.8, 0.9 and 1 seconds.

Results and Discussion

The activation time evaluation results are shown in figure 3.



a



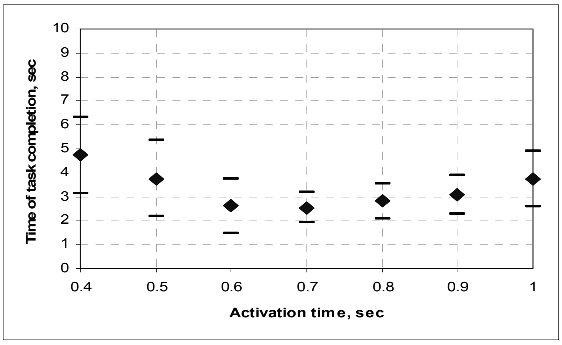
b

Fig. 3. *a* – time of task completion value vs. activation time value (‘◆‘ – time to complete the task; ‘▬’ – standard deviation of time to complete the task); *b* – number of errors value vs. activation time value (‘◆‘ – mean number of errors the users has made; ‘▬’ – standard deviation of errors)

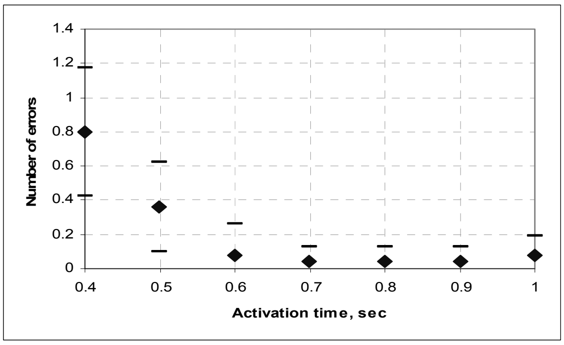
These two diagrams show how the activation time affects the task completion time and the number of errors made by the users. The shortest time is reached when activation time is set to 0.4 seconds. But this is not tolerated due to the highest mean number of errors. An error is counted in case if user presses a “wrong” key. Wrong key means that is executed a command or opened a menu level which is not necessary to complete the given task. The task was to open a door in the room no. 1 in the evaluation. The sequence of user actions according to fig. 2 should be: “start -> room1 -> door -> open”. If the sequence is: “start -> room2 -> exit -> room1 -> door -> open” than one error - selection of room 2 will be counted. Actions which are taken to correct an error are not counted as errors. In this case “exit” after executing “room 2” is not an error. Empty scanning cycles are not counted as errors too, because it affects the overall task completion time. So according to the results of the evaluation the activation time should be set to 0.7 sec to ensure the interface efficiency for most users in most of the cases due to the lowest mean number of errors and enough short activation time.

We noticed that the majority of errors are similar. They look like: “start -> room1 -> window -> exit -> door -> open” or “start -> room1 -> door -> close -> open” in the action sequence. That means the users aren’t able to execute a required command in time with all the measuring values of activation time. That happens due to the timer counted value that does not reset after any execution. It results in decreased activation time for the next command. There was necessary to modify the user interface program that after each executed menu option reset of the timer counted value would be done.

After the interface program modification the evaluation of activation time must be done repeatedly ensuring how the modification affects the results. The evaluation conditions also must be the same as in a previous one. The results of modified interface evaluation are shown in figure 4.



a



b

Fig. 4. *a* – time of task completion value vs. activation time value after interface modification (‘◆‘ – time to complete the task; ‘▬’ – standard deviation of time to complete the task); *b* – number of errors value vs. activation time value after interface modification (‘◆‘ – mean number of errors the users has made; ‘▬’ – standard deviation of errors)

It is evident that the task completion time reduces in all of the evaluation cases. Optimal activation time is also 0.7 seconds as it was before the modification. The average reduction of time to complete the task is 49.1% and the average reduction of number of errors is 41.4% in comparison with the results provided by the first evaluation. This is a quite significant affect to overall interface efficiency.

Conclusions

The most important criteria – the activation time was evaluated when investigating this kind of user interface. The results of both evaluations show that value of the activation time 0.7 seconds should be quite good according to number of errors and the shortest task completion time. The evaluated results show that approximately twice is reduced the specified task completion time and the number of errors by 41.4% when using modified user interface (timer value reset after each executed option).

Acknowledgements

This research was carried out during the project PACS of the Eureka program with the co-financing of Lithuanian Research and Studies Fund.

References

1. **Serafinavicius P., Lauruska V.** Adaptive Smart Home System // Human-Computer Interaction. Theory and practice (Part II). ISBN 0-8058-4931-9. - LEA Publishers, 2003. - Vol.2 P. - 133-137.
2. **Kubilinskas E., Lauruska V.** Smart home system for physically disabled persons with verbal communication difficulties // Assistive Technology – Added Value to the Quality of Life. AAATE'01. ISSN 1383-813X - IOS Press, 2001; P.300-304.
3. **Dix A.**, **Finlay J., Abowd G., Beale R.** Human-Computer Interaction. Second Edition. ISBN 0-13-239864-8 - Prentice Hall; P.25-26, 199-205, 416

Pateikta spaudai 2004 03 05

P. Serafinavičius, V. Lauruška. Neįgaliajam skirtos namų aplinkos valdymo sistemos sąsajos tyrimas // Elektronika ir elektrotechnika. – Kaunas: Technologija, 2004. – Nr. 4(53). – P. – 5-8.

Namų aplinkos valdymo sistema buvo suprojektuota panaudojant standartinį namų automatizavimo tinklą, asmeninį kompiuterį ir infraraudonųjų spindulių siųstuvą. Ji leidžia fiziškai neįgaliems žmonėms atlikti reikiamus veiksmus gyvenamojoje aplinkoje (įjungti šviesą, atidaryti duris, langą, įsijungti pageidaujamą televizijos kanalą ir pan.). Buvo ištirta sukurtos sistemos vartotojo sąsaja, įgyvendinta vienu paprasčiausių pozicionavimo metodu – objektų skenavimu per dialogų pagrindu sudarytą trijų lygių meniu. Tyrimo tikslas – optimalios objekto aktyvinimo trukmės nustatymas, nes ji yra vienas pagrindinių kriterijų, lemiančių bendrą sistemos efektyvumą. Rezultatai parodė, kad po 0,7 sekundės aktyvinimo tirti vartotojai padarė vidutiniškai mažiausiai klaidų ir pateiktą valdymo užduotį įvykdė greičiausiai. Tyrimas buvo pakartotas su patobulinta vartotojo sąsaja (laikmečio apskaičiuoto laiko nulio nustatymas). Tai labai pagerino rezultatus. Il. 4, bibl. 3 (anglų kalba; santraukos lietuvių, anglų ir rusų k.).

P. Serafinavičius, V. Lauruška. Home Environment Control System for the Disabled: User Interface Testing // Electronics and Electrical Engineering. – Kaunas: Technologija, 2004. – No. 4(53). – P. – 5-8.

Home environment control system was designed using a standard home automation network, a PC and an infrared transmitter. It makes home environment for the disabled more accessible. The design of user interface was tested which was implemented using objects scanning method on the dialog-based 3-level menu. The goal of the investigation was to evaluate optimal object activation time which is one of the most important criteria affecting the overall system effectiveness. The results show that tested users make less errors and do the task faster when activation time is 0.7 second. The evaluation was done repeatedly with modified user interface (timer value reset after each executed option). The results were got significantly better. Ill. 4, bibl. 3 (in English; abstracts in Lithuanian, English and Russian).

П. Серафинавичюс, В. Лаурушка. Исследования интерфейса пользователя системы управления домашнего окружения для инвалидов // Электроника и электротехника. – Каунас: Технология, 2004. - № 4(53). – С. 5-8.

Система управления дома спроектирована, используя стандартную сеть автоматизации дома, персональный компьютер и передатчик инфракрасных лучей. Эта система создает условия для инвалидов управлять окружающей средой (включить свет, открыть окно, включить телевизор и прочее). Был исследован интерфейс реализован методом сканирования меню потребителя. Цель исследований - определение оптимального времени, необходимого для активизации клавиша. Результаты показали, что оптимальное время активации – 0,7 сек. Исследования были повторены с модифицированным интерфейсом (ресетирование таймера), которого результаты улучшились умеренно. Ил. 4, библ. 3 (на английском языке; рефераты на литовском, английском и русском яз.).

DOI: 10.5755/j02.eie.10929