KNXsim: Simulator Tool for KNX Home Automation Training by Means of Group Addresses

Juan A. Gómez-Pulido[1](✉) and Alberto Garcés-Jiménez[2]

[1] Department of Technologies of Computers and Communications, Universidad de Extremadura, 10003 Cáceres, Spain
jangomez@unex.es

[2] Department of Computing Sciences, Universidad de Alcala, 28805 Alcala de Henares, Spain

Abstract. The growth of home automation makes it necessary to train qualified personnel in the knowledge and use of the most important standards, among which KNX is the leader in Europe. The programming of home automation services with KNX is based on the concept of group addresses, which allow defining the behavior of the domotic devices for a previously defined facility. This training is complex and usually requires a physical domotic facility where the previously programmed design can be tested, which makes the learning process difficult for the students. In this article we show the development and characteristics of a multi-platform simulator that recreates the real operation of an automated facility for any programming scheme defined by the student, validating it by means of a virtual installation that includes different devices usually involved in home automation. This simulator has allowed the generation of a wide range of cases of use in the training of the most usual domotic services.

Keywords: Simulation · Home automation · KNX · Group address

1 Introduction

Home automation (or Domotics) [7] is a technological area that deals with the development and integration of automated systems in all types of facilities, with the aim of efficiently managing services such as security, energy, communications, comfort, etc. This area of knowledge is undergoing a fast technological growth and social deployment, with a strong projection in innovation. For this reason, academic institutions include its study in practically all engineering degrees: industrial, telecommunications, computer science, electronics, etc.

In the early 1990s, Europe promoted the “European Installation Bus” (EIB) home automation standard, which evolved into the EIB Konnex standard, better known as KNX [1]. It is an open standard, oriented to the technical management of buildings and homes. Most of european domotized facilities are deployed by using the KNX standard.

J.-L. Guisado-Lizar et al. (Eds.): SIMUtools 2023, LNICST 519, pp. 34–43, 2024.
https://doi.org/10.1007/978-3-031-57523-5_3

KNX allows for highly scalable designs and can be used in installations of any size, large or small. KNX facilities are very flexible, as they allow components from different manufacturers to be connected. The compliance of domotic devices with this standard is managed by the KNX Association [3], which currently has more than 500 members in 190 countries.

The design of a domotic system based on KNX architecture can be planned schematically and the devices involved can be programmed manually. However, this procedure is complex, not very flexible and subject to failures. It is important to note that the operation of a KNX system depends on how the so-called Group Addresses (GA) are used and assigned. The concept of GA is fundamental, as its understanding will enable engineers to design a correct domotic system.

The automation of the design and programming processes of a KNX system is mostly done by means of the Engineering Tool Software (*ETS*) [5], which relies on GAs for the definition and operation of the domotic services. This powerful tool allows to design any KNX configuration and to program it; however, it lacks the ability to simulate the designed system for validation before programming the real facility. In other words, we will only know that the domotic services will work according to our specifications when we program it in the real environment. This limitation makes the possibility of using simulators attractive.

This paper shows the development of the *KNXsim* simulator, which allows to simulate the real behavior of domotic services as they have been programmed using GAs. This simulator was mainly designed for training purposes, in order to facilitate the understanding of the concept of group addresses and its experimentation without the need of having the physical home automation facility.

2 Related Works

ETS is the main software tool devoted to the design and programming of KNX-based home automation systems. It is a powerful and professional tool, but lacks the ability to perform simulations. To fill this gap, KNX Association developed the *KNX Virtual* [2] tool. This software performs a simulation of the system, although it has two limitations in regards to a comprehensive academic purpose. On the one hand, *KNX Virtual* can only run under the Windows operating system, so computers with the Linux O.S. (quite common in academic, student and laboratory environments) would not be able to install this simulator. On the other hand, the visual environment is not very user-friendly, as it consists of a set of rectangles that need to be conceptually associated with the different home automation devices, which have to be imagined. In addition, *KNX Virtual* requires a previous study in its use and does not facilitate the understanding of the concept of group addresses, a basic concept in this domain.

KNX simulator [4] is a powerful online simulation tool designed to learn KNX through practice without the need of having physical devices. Its beautiful visual environment and high level of user interaction facilitates the learning process. This tool is mainly focused on the selection of real home automation devices, their connectivity and simulated operation. However, the design of the installation to be programmed (including the definition of the group addresses) still requires the use of the *ETS* tool.

In this sense, unlike *KNX simulator*, *KNXsim* facilitates the learning of KNX home automation systems under the paradigm of group addresses without the need of other tools, such as *ETS*. This way, *KNXsim* provides a conceptual and academic approach as a first step for a better knowledge that will facilitate the later use of *ETS* and/or *KNX simulator*.

3 Methodology

3.1 Domotic Approach

The simulator proposes a house with a particular domotic architecture, not defined by the user in terms of its elements and physical connections. Therefore, the simulator was designed to allow operational, but not constructive, flexibility.

This design approach was chosen because, as mentioned above, the simulator was developed mainly for training purposes. It is intended to facilitate the understanding of the concept and handling of group addresses for home automation, which is perfectly covered by a rigid physical architecture, as long as it is broad enough to cover the most common domotic services.

This approach allows to generate a wide set of use cases of group addresses, where different operational approaches for the same domotic purpose can be explored. In this way, training of group address is simplified if the same physical home automation structure is always used as a starting point.

3.2 Architecture of the Domotic House

Figure 1 shows the domotic architecture of the facility that will be used by the simulator, in terms of its functional elements and connections. In this case, the installation represents a home.

There are several types of elements:

– Sensors. The sensors are the domotic devices that receive information from the environment (temperature, luminosity, touch, etc.) and send the corresponding values to the domotic bus. Thus, the domotic system will make the appropriate decisions according to the way it was programmed.
– Loads. The loads are electromechanical devices that can be controlled by a home automation system for purposes of climate control, security, comfort, etc. For example, loads are lights, alarms, taps, doors, blinds, etc. We remark that the loads are not domotic devices, but devices that are controlled by the domotic devices.
– Actuator. 8-channel binary output switch. This device controls the on/off switching of the loads according to the values received from the sensors.
– Buses. Electrical power line and KNX bus. The first one feeds electrically the actuators and the loads. The second connects the sensors and the actuator to enable communication among them.

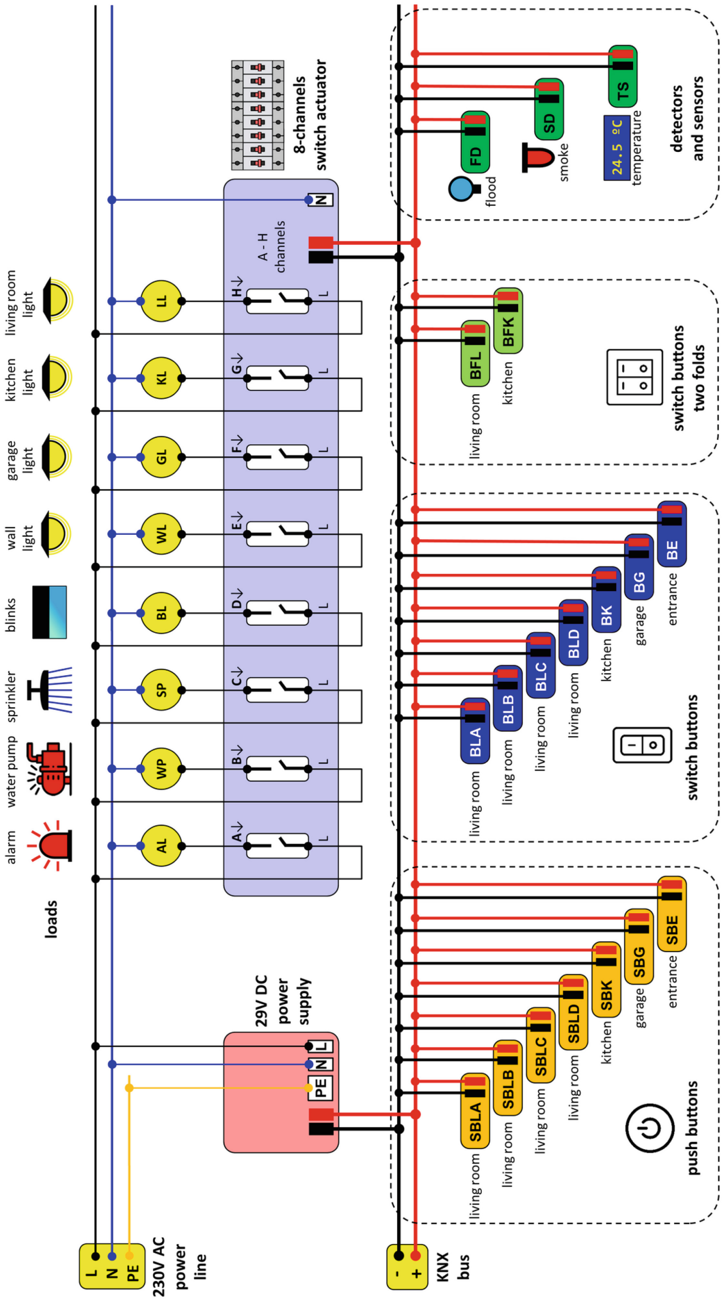

Fig. 1. Domotic architecture considered by the simulator.

Table 1. Devices involved in the domotized house.

ID	Device
Loads:	
AL	alarm
WP	water pump
SP	sprinkler
BL	blinks
WL	wall light
GL	garage light
KL	kitchen light
LL	living room light
Sensors:	
FD	flood detector
SD	smoke detector
TS	temperature sensor
Switch buttons:	
SBLA	switch button A in living room
SBLB	switch button B in living room
SBLC	switch button C in living room
SBLD	switch button D in living room
SBK	switch button in kitchen
SBG	switch button in garage
SBE	switch button in entrance
Push buttons:	
BLA	push button A in living room
BLB	push button B in living room
BLC	push button C in living room
BLD	push button D in living room
BK	push button in kitchen
BG	push button in garage
BE	push button in entrance
Push buttons 4-fold:	
BFL	push button-four in living room
BFK	push button-four in kitchen

Table 1 lists the visible devices involved in the user's interaction with the home automation system.

Figure 2 shows the domotic house from the end user's point of view, where all the sensors and loads are visible. The simulator uses this view for the validation

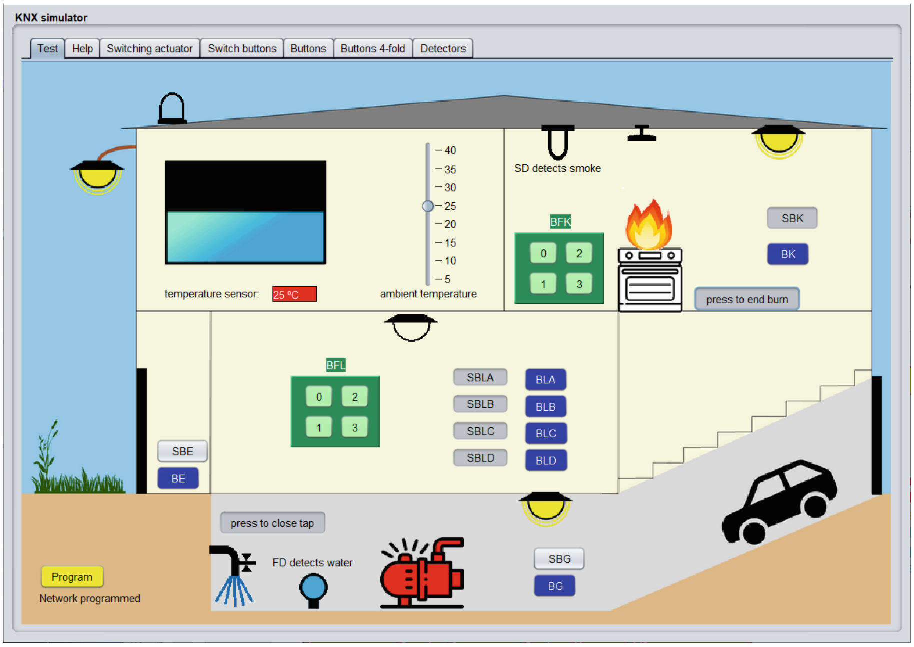

Fig. 2. Simulator view of the domotic house.

of the domotic behaviour as the group addresses have been programmed. The simulator provides, for this view, a set of buttons that allow the user interaction to control the house, as well as the necessary mechanisms to define environmental events that condition the domotic behavior, such as ambient temperature, smoke generation, water flooding, etc.

Table 2 shows the group addresses architecture of the house. As we said before, *KNXsim* considers a fixed architecture of a domotic system. This requires the definition of a table of group addresses (in our case, of two levels), which assign to each address a meaning in terms of the operability of the associated load. The group addresses have been classified into three domains of home automation services: lighting, security and comfort.

3.3 Simulator Workflow

Figure 3 shows the workflow developed for the simulator. The user of the tool has two options: to design the home automation application and to test its operation.

The user designs the home automation application by modeling the scene, i.e. designing the links between the actuator channels and the sensors by means of the corresponding group addresses. Once this modeling is done, the group addresses are automatically sent to the home automation devices (switch actuator, buttons, and detectors), which generate signals that indicate valid events. If the group address selections are incorrect or they have not yet been selected, these signals

Table 2. Two-level group addresses architecture of the house.

Main group	*Meaning*	Address	*Meaning*	Group address
0	*Lighting*	0	*On light WL*	0/0
		1	*Off light WL*	0/1
		2	*On light GL*	0/2
		3	*Off light GL*	0/3
		4	*On light KL*	0/4
		5	*Off light KL*	0/5
		6	*On light LL*	0/6
		7	*Off light LL*	0/7
1	*Security*	0	*On alarm AL*	1/0
		1	*Off alarm AL*	1/1
		2	*On water pump WP*	1/2
		3	*Off water pump WP*	1/3
		4	*On sprinkler SP*	1/4
		5	*Off sprinkler SP*	1/5
2	*Comfort*	0	*Up blinks BL*	2/0
		1	*Down blinks BL*	2/1

are not sent. Therefore, the simulator will only act on the domotic system when it receives valid events.

The result of these events on the domotic house is only generated when the test order is received by the user, and this order will only arrive if there are valid events. This will prevent the simulator from activating non-configured or erroneous applications.

The simulator generates the results by visualizing the effects that the domotic devices and the environmental conditions generate on the loads. An undesired result can be achieved if the configuration of the group addresses for the designed scene is not correct, in which case this scene has to be redesigned by selecting the correct set of group addresses.

4 Results

The simulator was programmed using the Java language [6] and the Netbeans [8] graphics library. In this way, the application can run independently of the operating system, which favors its dissemination in any academic installation.

Currently, *KNXsim* is used as a teaching tool in the lab practices of the Domotics subject, that belongs to the Degree on Telecommunication Engineering of the University of Extremadura, Spain. In addition to the application installed on the laboratory computers, students have a wide range of practice scripts to implement use cases in the programming of home automation services. Each

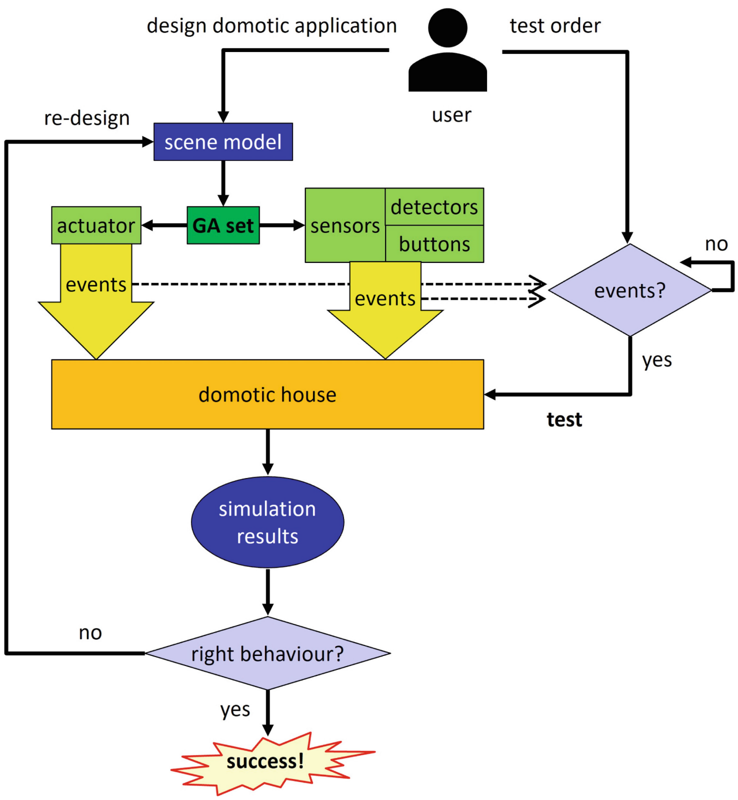

Fig. 3. Simulator workflow.

practice is based on some specifications, with which the student must program the appropriate group addresses for each domotic device. This programming is not deterministic, but the specified result must be so. In this way, the student's understanding and creativity are favored in order to reach the same end.

The following list shows some of the use cases considered in the practices for the students according to the timetable of the subject. These cases were selected as they cover basic domotic services present in many homes:

- Lighting control based on user interaction.
- Lighting control based on door opening.
- Alarm control based on smoke, flood or intrusion detection.
- Sprinkler control based on fire detection.
- Water pump control based on the detection of flooding in the garage.
- Blinds opening degree control according to the outdoor temperature.

Nevertheless, other use cases different from those listed and many other combinations can be configured from the same domotic devices and loads available

Table 3. Features comparison of KNX simulator tools.

Feature	KNXsim	KNXsimulator	KNX virtual
Realistic graphical interface	Yes	Yes	No
Platform	All	online	Windows only
Supports group addresses	Yes	No	No

in the simulator. This will favor the development of projects where the student will autonomously create home automation services of interest to the user.

Table 3 shows a comparison of the simulator with the alternatives described in Sect. 2 in terms of basic operational characteristics from a didactic point of view, which is the main purpose of KNXSim. The simulator stands out in two features: it has been developed to be used under any operating system that supports Java (e.g. Windows and Linux) and to use group addresses as a fundamental element for designing home automation applications.

5 Conclusions

The *KNXsim* simulator was developed to facilitate the learning of the design of domotic systems based on the KNX standard through the concept of group addresses. This tool fills a gap in *ETS*, the main software devoted to the programming of these systems: *ETS* does not allow a simulation prior to the programming of the physical installation. This means, on the one hand, that anomalous behaviors can only be detected if we have access to the real system once it has been programmed; on the other hand, debugging times are longer, as a real validation is always required. Other tools allow prior simulation, but are more difficult to use or have a more professional approach.

In addition to allowing a simulation of domotic behavior, KNXsim was developed with the purpose of facilitating training in domotic programming by means of group addresses. This simulator, intended for academic environments, presents a user-friendly environment, easy to understand and operate, and a set of manuals and practice scripts for the student.

The current experience is that students understand the concept of group addresses and become familiar with KNX-based systems better and more quickly. Nevertheless, we want to go further into this educational area, proposing some indicators to analyze the impact of this tool on student learning regarding to traditional learning methods. Also, we want to analyze the usability facts of real professional users in the domotic area.

Acknowledgements. The authors thank the support of Fundacion General CSIC and the State Research Agency, Spain, under the contracts 0551-PSL-6-E (iFriend project) and PID2022-137275NA-I00 (X-BIO project) respectively, by the resources used for developing the simulator.

Author contributions. Juan A. Gómez-Pulido: Conceptualization; Methodology; Investigation; Software; Computational; Writing; Supervision. Alberto Garcés-Jiménez: Ideas; Testing; Writing-review.

References

1. KNX Handbook for Home and Building Control, 6th edn. KNX Association (2015)
2. ETS6 and KNX Virtual for testing and learning about KNX. Technical report, KNX Association (2023)
3. KNX Association: Mission and Objectives (2023). https://www.knx.org/knx-en/for-professionals/What-is-KNX/Our-mission/. Accessed Sept 2023
4. KNX Simulator, S.L.: KNX Simulator (2023). https://www.knxsimulator.com. Accessed Sept 2023
5. Meier, M., Szczensny, C.: The Ultimate Guide to KNX Programming: One of fastest, easiest and cheapest ways to learn KNX programming and ETS 5. Voltimum (2018)
6. Sharan, K., Davis, A.: Beginning Java 17 Fundamentals: Object-Oriented Programming in Java 17 (2022). https://doi.org/10.1007/978-1-4842-7307-4
7. Spivey, D.: Home Automation For Dummies. O'Reilly (2015)
8. Wielenga, G.: Beginning NetBeans IDE: For Java Developers. Apress (2015)