

POLITECNICO DI MILANO
Master of science in Computer Engineering
Dipartimento di Elettronica e Informazione



BTicino S.p.A

bticino

Performance Evaluation and Queuing
Network Model of the SCS BUS in
BTicino S.p.A domotic system
(My Home System)

Supervisor: Prof. Paolo Cremonesi
Co-supervisor: Prof. Fabio Salice
Company supervisor: Ing. Gioel Asseconi

Thesis of Master degree :
MAJID SHOKROLAHI YANCHESHMEH

Academic year 2011-2012

to my

MOTHER and FATHER

with love

Abstract

For homeowners and designers seeking distinctive style and the latest in home technology, *My Home system*(MHS) is a home automation system of **BTicino S.p.A** which aims to increase comfort, safety, communication and control within the home. *MHS* is based on *BTicino SCS* (simple connection system) technology, all devices are connected together by a BUS cable (twisted pair cable), dedicated to exchanging information and to transporting the power supply voltage. All of the My Home devices are connected to the BUS cable. Each of the *My Home* applications can be controlled within the home via touch screens or wall switches, or outside the home via a telephone or computer. In *My Home system* there are different types of the control systems which the owner by considering of his needs can have them, such as Automation system, Thermoregulation system, etc. Each system has different devices and commands. These systems are connected to each other by using the SCS Bus as communication line between them.

Performance evaluation of the BTicino SCS bus and Queuing Network model of MHS system was concerned in this project to illustrate the bus occupancy in different kind of configurations. First we had to know the system very good to define queuing network model and parameters and then by using a simulator estimate the behavior of the system. Accuracy of the model has been compared by real data analysis of the sample system.

To aim the goals of the project some activities have been done. First by using *JMT*(JAVA modeling tool), *JSIMgraph* tool, we have designed and simulated queuing network model of the SCS BUS. Then, the results of the simulator have been compared with the result from real data of the system. In addition, after solving the model with MVA solution(*JMVA* tool), comparison between the former results and JMVA results have been done. Furthermore, for each specific installation of the system a specific model is needed, thus, we have developed a QN Model generator (BMG) framework in JAVA to prepare the specific model as input, XML file, for the JMT simulator. Finally, we expand the model for the tree SCS bus network structure

when other SCS buses are integrated in the system.

By comparing the result with real data, we could define quite accurate QN model of the system.

Sommario

Per proprietari di case e i designer l'obiettivo è quello di cercare uno stile distintivo e le ultime novità riguardanti la tecnologia per le case, My Home system (MHS) è un sistema domotico utilizzato da BTicino S.P.A. che mira ad aumentare il confort, la sicurezza, la comunicazione e il controllo all'interno delle case. MHS è basato su una tecnologia a sistema di connessione semplice (SCS) dove tutti i dispositivi sono collegati tra loro tramite un cavo BUS (cavo a doppiini intrecciati), dedicato allo scambio di informazioni e di trasporto della tensione di alimentazione. Tutti i dispositivi My-Home sono collegati al cavo BUS. Ciascuna delle applicazioni My Home può essere controllata all'interno della casa tramite uno schermo touch o attraverso interruttori a parete, mentre fuori casa può essere controllata tramite un telefono o un computer. In MHS ci sono diversi tipi di sistemi di controllo che il proprietario della casa in base alle sue necessità può decidere di avere, come ad esempio un sistema di domotica, di termoregolazione, ecc. Ogni sistema ha diversi dispositivi e comandi. Questi sistemi sono connessi attraverso il Bus SCS di cui prima, linea di comunicazione tra loro. Una valutazione delle performance del Bus SCS è stata studiata in questo progetto per illustrare l'occupazione del Bus in diversi tipi di configurazioni. Esistono due tipi di carico, generati dai dispositivi sul Bus SCS: il carico dovuto all' User Action (UA), quando l'utente inizia ad utilizzare il sistema e il Background Load (BL), quando i sistemi scambiano informazioni tra loro. Per prima cosa è stato necessario conoscere il sistema in dettaglio in modo da poter essere in grado di definire il modello di rete di accodamento e i rispettivi parametri, dopodiché utilizzando un simulatore è stato possibile stimare il comportamento del sistema. L'accuratezza del modello è stata verificata dividendo il modello generale in un sotto modello, il che ha reso la valutazione più facile. Per raggiungere gli obiettivi del progetto, alcune attività sono state fatte. In primo luogo mediante JMT (strumento di modellazione JAVA) e JSIM-graph è stato progettato e simulato il modello di rete di accodamento del BUS SCS. A questo punto i risultati del simulatore sono stati confrontati

con i risultati ottenuti dai dati reali del sistema. In aggiunta, dopo aver creato il modello con un algoritmo MVA (attraverso lo strumento di JMVA) è stato effettuato il confronto tra i risultati precedenti e i risultati JMVA. Inoltre, poiché per ogni specifica installazione del sistema è necessario un modello specifico, è stato sviluppato un framework generatore di modelli QN in JAVA per preparare lo specifico modello come input, attraverso un file XML, per il simulatore JMT. Si è deciso, poi, di espandere il modello per i BUS SCS ad albero, quando altri bus SCS sono integrati nel sistema. L'ultimo step è stato la definizione del modello analitico del sistema. Confrontando il risultato con i dati reali è stato possibile definire un modello QN del sistema abbastanza accurato.

Acknowledgements

I would like to express my deep-felt gratitude to my advisor, Prof. Paolo Cremonesi, and special thanks to my co-supervisor, Prof. Fabio Salice in the Department of Electronics and Informatics at The University of Politecnico di Milano in Italy, for their advice, encouragement, enduring patience and support.

I also wish to deeply thank my supervisor, Ing. Gioel Asseconi, in BTicino S.p.A company, for all his kind support, help and advices which made it possible for me to keep the work going on. Special thanks also to "Stratos 0" group members, in BTicino S.p.A company; Elisabetta Bianchi, Andrea Moro, Stefano Alessio and Stefano Frigerio, who were abundantly helpful and offered invaluable assistance, support and guidance. Not forgetting to my best friends, Charles Salvestroni and Rocco Davide Dato, who always have been there.

I would also like to convey thanks to BTicino S.p.A company, for giving me this opportunity to do my thesis there and providing me what were needed. My special thanks to Ing. Marco Catuzzo, who has defined the project "Stratos", for providing his support, comments and additional guidance to the completion of this work.

Additionally, I would like to thank The University of Politecnico di Milano, especially Como Campus, the Department of Electronics and Informatics, professors and staff for all their hard work and dedication, providing me the means to complete my degree.

And finally, I wish to express my love and gratitude to my family; for their understanding and endless love, through the duration of my studies.

NOTE: This thesis was submitted to my Supervising Committee on the July 4, 2012.

Contents

Abstract	I
Sommario	III
Acknowledgements	V
1 Introduction	1
1.1 My Home System(MHS)	1
1.1.1 Systems	2
1.2 Motivation	2
1.3 Project goals	3
1.4 Thesis contributions	3
1.5 Thesis organization	4
2 BTicino S.p.A. Blocks	5
2.1 Introduction	5
2.1.1 Levels	5
2.2 Structure of the Frame	7
2.2.1 Brief frame	7
2.2.2 Extended frame	8
2.2.3 Collisions	10
2.2.4 Bus Access Time	10
2.3 SCS information	11
2.4 My Home Automation system	12
2.4.1 Architecture of the system	13
2.4.2 Automation commands	13
2.5 MY HOME Thermoregulation System	15
2.5.1 Operation mode	15
2.5.2 Thermoregulation's commands	19
2.6 My Home Safety Burglar Alarm System	19
2.6.1 Operation mode	20

2.6.2	Burglar Alarm's Commands	21
2.7	My Home Sound Diffusion system	21
2.7.1	Brief description of functions	21
2.7.2	Sound Diffusion's Commands	22
2.8	My Home Energy Management system	22
2.8.1	Notes of the system	23
2.8.2	Energy Management commands	24
3	BACKGROUND - RELATED WORK	25
3.1	Queuing Network Model	25
3.2	Network simulation	27
3.3	Java Modeling Tool (JMT)	28
3.3.1	JSIM model simulation	29
3.4	Getting the Real data (Log file)	32
3.4.1	The SCS viewer	32
4	Queuing network Model	35
4.1	Introduction	35
4.2	Load in the system	36
4.2.1	User Action Load	36
4.2.2	Background Load	36
4.3	Queuing Network Model	39
4.4	Modeling the SCS BUS using JMT	41
4.4.1	Define the Classes	44
4.4.2	Distribution of the Workloads	46
4.5	Service Time in the BUS	47
4.6	How to Simulate the Time Access to the SCS BUS	48
4.6.1	The basic components and their parameters	49
4.7	Results and Accuracy of the Model	50
4.7.1	Throughput	51
4.7.2	Utilization	54
4.7.3	Response Time	54
4.7.4	System Response Time	57
4.8	Mean Value Analysis - using JMVA	57
4.8.1	Throughput	58
4.8.2	Utilization	60
4.8.3	System Response Time	60
4.8.4	Residence Time	60
4.9	Tree SCS BUS network	64
4.9.1	QN Model for Level 3 and 4	65

4.9.2	QN Model for Level 3, 4 and Burglar Alarm	66
4.10	Real Data Analysis	68
4.10.1	Real data Analysis - Thermoregulation Background Load	68
5	BTicino QN Model Generator (BMG)	75
5.1	Introduction	75
5.2	JMT XML file	76
5.3	BMG application	76
5.3.1	Selecting the systems in the MyHome system	77
5.3.2	Getting Configuration	78
5.3.3	Choosing the type of commands	79
6	Conclusion and future works	81
6.1	Future works	82
	References	83

List of Figures

2.1	Complex system	6
2.2	Levels in My Home system	7
2.3	Point to Point frame	9
2.4	Broadcasting frame	10
2.5	Collision example	11
2.6	Thermoregulation system schema	16
2.7	Energy Management system schema	23
3.1	The starting window of JMT	29
3.2	The starting window of JMT	30
3.3	SCS Viewer Interface	33
3.4	Log file of the SCS bus - taken by SCS viewer	33
4.1	MyHome system - SCS BUS	40
4.2	MyHome system - SCS BUS	41
4.3	Simple QN Model for Level 3	43
4.4	Uniform Distribution	49
4.5	Throughput Result- simplified model to Extended class	53
4.6	Throughput Result- simplified model to Extended class	55
4.7	Utilization Result - simplified model to Extended class	56
4.8	System response time - Extended class and Sound class	57
4.9	JMVA Throughput result - textual	58
4.10	Throughput Results Comparison between JMVA and JSIM - simplified model to Extended class	59
4.11	Utilization Results Comparison between JMVA and JSIM - simplified model to Extended class	61
4.12	System Response Time Results Comparison between JMVA and JSIM - simplified model to Extended class	62
4.13	Residence Time Results Comparison between JMVA and JSIM - simplified model to Extended class	63
4.14	Tree SCS BUS Network	64

4.15	QN model for Level 3,4	65
4.16	QN model for Level 3,4 and Burglar Alarm	67
4.17	Real data - Thermoregulation	69
4.18	Real data - Thermoregulation , Utilization	70
4.19	Real data - Thermoregulation , Utilization of a period 15 minutes	70
4.20	Real data - Distribution of data in a period of 15 minutes	71
4.21	QN model -Thermoregulation BL	71
4.22	JSIM Results - Thermoregulation Background Load	72
5.1	JMT Architecture	76
5.2	Flowchart of BMG	77
5.3	BMG Interface	78
5.4	First step:Selecting the systems in the MyHome system	79
5.5	Second step:Getting Configuration	79
5.6	Third step:choosing the type of commands	80

List of Tables

1.1	Systems in My Home system	3
2.1	Automation's commands- Shutter system	14
2.2	Automation's commands- Lighting system	14
2.3	Thermoregulation's commands	19
2.4	Burglar Alarm's commands	21
2.5	Sound Diffusion commands	22
2.6	Energy Management- METER commands	24
2.7	Energy Management- LOADS commands	24
4.1	Background Load in Thermoregulation System	38
4.2	Components and their model	42
4.3	Transmission time and bus occupancy for the three types of the frames	44
4.4	defined parameters for the SCS bus	50

Chapter 1

Introduction

“Simplicity is the ultimate sophistication.”

Leonardo da Vinci

This thesis has been done under a funded internship activity in BTicino S.p.A company as starting part of the "Stratos" project which has been defined to analyze and improve performance of the MHS system and validate the configuration of plant and scenario. The goal of this thesis activity is to make Queuing Network model for single and multiple SCS BUS and evaluate performance of the system according to a given configuration of a plant. To do these, having good knowledge of the MHS system is necessary and in the following and next chapter an introduction to MHS system has been given.

1.1 My Home System(MHS)

My Home System (MHS) is the domotic system of BTicino S.p.A and over the past few years, Home Automation Systems (Domotic system) have received a great deal of attention. For homeowners and designers seeking distinctive style and the latest in home technology, *My Home* [12] is a home automation system which aims to increase comfort, safety, communication and control within the home.

My Home is based on *SCS* (simple connection system) technology of BTicino S.p.A, all devices are connected together by a BUS cable (an electrical conductor that makes a common connection between several circuits), dedicated to exchanging information and to transporting the power supply voltage. All of the *My Home* devices are connected to the BUS cable -an unshielded twisted pair cable. Each of the My Home applications can be

controlled within the home via *Touch screens* or wall switches, or outside the home via a telephone or computer.

1.1.1 Systems

In *My Home* system there are different types of the control systems which the owner by considering of his needs can have them:

- Automation system(lighting control, shutter control)
- Thermoregulation system
- Burglar Alarm system
- Sound Diffusion system
- Energy Management system

Each system has different devices and commands which are explained later. These systems are connected to each other by using the *SCS* Bus as communication line between them. The system is made up of devices which communicate via the SCS BUS wire; some devices, such as the controls and actuators, perform basic functions while others, such as the *TOUCH SCREEN*, perform advanced functions. As well as allowing the devices to communicate, the BUS transports their power supply. The communication between devices is through sending the frames in Bus. Inside *My Home* commercial offer, there are many different systems where devices communicate using *SCS* protocol. For this reason there are some common rules that allow messages not to be misunderstood. The systems are shown in table 2.1.

1.2 Motivation

All of the My Home devices are connected to the BUS cable. Each of the *My Home* applications can be controlled within the home via touch screens or wall switches, or outside the home via a telephone or computer. Knowing the capacity of the system regarding to the usage of the system is important to be investigated, while we know in *My Home system* there are different types of the control systems which the owner by considering of his needs can have them, such as Automation system, Thermoregulation system, etc. Each system has different devices and commands and they are connected to each other by using the SCS Bus as communication line between them.

-	My Home Subsystem
1	Lighting-Automation
2	Electrical system supervision(Energy Management)
3	Thermoregulation
4	Anti-Intrusion(Burglar Alarm)
5	Luminous/alarms signaling(Hospital system)
6	Video door entry system
7	Access control
8	Multimedia (Sound Diffusion)
9	Auxiliary
10	Integration Devices

Table 1.1: Systems in My Home system

In the filed of estimating system capacity, not much research has been conducted. On the other hand, real-time communication of My Home system with the QoS concept and also high-performance installation of a plant were concerned to do this project.

1.3 Project goals

Modeling My Home system to estimate occupancy of the SCS Bus network is the main goal of this project. Hence, performance evaluation of the SCS bus was concerned in this project to illustrate the bus occupancy in different kind of configurations.

When an installer wants to decide how to install and configure a plant, a framework which uses a methodology to estimate system capacity is needed to help the installer by some advices according to anticipated result from simulation.

1.4 Thesis contributions

The contribution of this thesis can be summarized as follows:

- By using JMT(JAVA modeling tool) [23], JSIMgraph tool, we have designed and simulated queuing network model of the SCS BUS.
- The results of the simulator have been compared with the result from real data of the system.

- After solving the model with MVA solution(JMVA tool), comparison between the former results and JMVA results have been done.
- For different kind of the configuration, we have developed a QN Model generator framework in JAVA to prepare the specific model as input, XML file, for the JMT simulator.
- Later on, Extended models for the tree SCS bus structure when other SCS buses are integrated in the system have been presented.
- Finally, an Analytical model of the system has been presented.

1.5 Thesis organization

While this chapter has given an overview of the motivation and scope of this research on Queuing Network of SCS Bus network, as well as a brief introduction, the remainder of this thesis is organized as follows:

- Chapter 2 presents a thorough overview of BTicino Blocks and My Home System structure.
- Chapter 3 explains the background and related work. This chapter also presents a thorough overview the tools have been used for this project.
- Chapter 4 describes the procedure of formulating the Queuing Network model, accuracy of the model by comparing the result with real data and introducing the extended Model for tree SCS BUS network.
- Chapter 5 Introduce the BMG application which has been developed to generate the models according to given configuration of MHS.
- Chapter 6 concludes by summarizing the contributions and the direction of future research based on this thesis is also discussed.

Chapter 2

BTicino S.p.A. Blocks

"Will you look back on life and say, "I wish I had," or "I'm glad I did"?"

Zig Ziglar

2.1 Introduction

In general approach, domotics system has to cover installations composed by more buildings belonging to the same complex. As it can be easily understood, everything concerns *video door entry* system and access control system has to work in such a system. Also other systems, however, can be included in common areas. A generic example of a complex system is represented in the figure 2.1.

2.1.1 Levels

My Home system is divided in levels each one representing the topology of the system. From now on, the system will be called MyHome System (MHS). Each level has a representing number; this number increases going from common areas to private flats. These are the main levels involved:

0. Back-bone : level 0 or L0: generally called back-bone
1. Building : level 1 or L1: generally called building;
2. Raiser: level 2 or L2: generally called raiser;
3. Apartment: level 3 or L3: generally called flat(Apartment);
4. Private raiser: level 4 or L4: generally called private raiser.

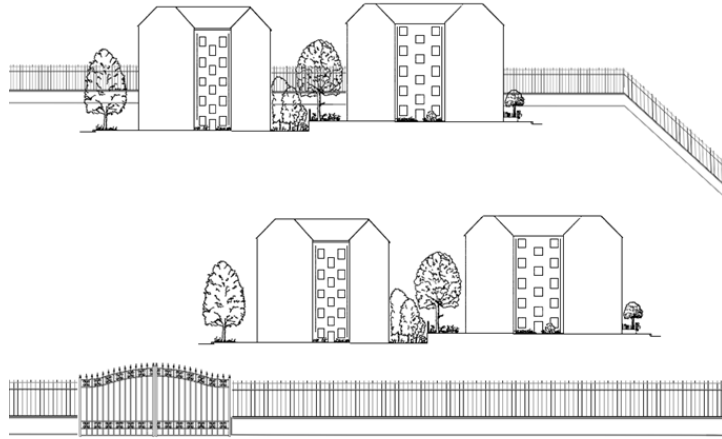


Figure 2.1: Complex system

System commands and functions can work across many levels. For example, supervision functions work with L0 to L4. A MHS is not necessarily made up of all levels. L3 is usually present. Level interface: Two different levels may be linked through a level interface LI, which is an object that can perform a set whatever of the following functions:

- galvanic separation of levels
- technological separation (e.g. 8wire/2wire)
- command filtering in order to transmit selected messages only
- frame modification (e.g. address changing)
- audio-video link across different levels
- command routing

These are the possible level interfaces:

- LI1: interface between L0 and L1 \Rightarrow building interface;
- LI2: interface between Lx and L2 \Rightarrow raiser interface;
- LI3: interface between Lx and L3 \Rightarrow flat interface;
- LI4: interface between L3 and L4 \Rightarrow private riser interface.

Each Level can be composed by a complete system, apart from *video door entry* and access control systems.

Different levels are separated by a *Level-Interface*; this device decide how (and if) transmit every received message to the other level or other Bus, similar to Router in LAN network.

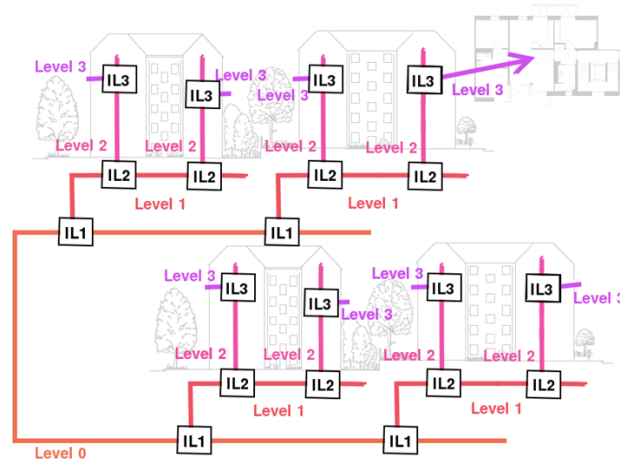


Figure 2.2: Levels in My Home system

2.2 Structure of the Frame

The frames are the basic unit of SCS communication system. In fact, they contain , information about the sender, recipient, and the type of command to be executed. Each frame is formed by a number of bytes depends on the type of system that must handle and the kind of information that should be transmitted. Within the same frame, between a byte and the next, one bit is transmitted. This bit has the logic value "1" and then a duration of 104 microseconds. The frames are divided mainly into two types:

- Brief frame
- Extended Frame

2.2.1 Brief frame

The short frame, also called addressable reduced, are used by systems that require a limited number of information to be exchanged and are formed

by a number of devices below 100 (Automation system, Intrusion system, energy management system)[1].

The short frame is composed of 7 bytes, so that its length is 7904 ms (7 x 10 x 104 microseconds + 6 x 104 microseconds, due to the 6 bits transmitted between a byte and the next). The structure of the short frame is as follows:

STX	DES	LOC	CMD	ARG	CKS	ETX
-----	-----	-----	-----	-----	-----	-----

- STX is the Start Frame, whose hexadecimal code is A8;
- DES is the address of which is for the frame;
- LOC is the address of the sender;
- CMD is the type of command to be transmitted;
- ARG is the argument of that command;
- CKS is the checksum of data sent (XOR of previous bits);
- ETX is the End Frame, whose hexadecimal code is A3.

2.2.2 Extended frame

The extended frame, also called extended addressing, are used by systems that require multilevel BUS, in which for the large number of devices or for reasons of distance, it is necessary to separate the implant on more than one BUS isolated. The extended frame is composed of 11 bytes, so that its temporal duration is 12.48ms. The structure of the extended data frame is as follows:

STX	DES1	DES2	Des3	LOC1	LOC2	LOC3	CMD	ARG	CKS	ETX
-----	------	------	------	------	------	------	-----	-----	-----	-----

- STX is the Start Frame;
- DESX represents the address of which is addressed to the frame;
- LOCx represents the address of the sender;
- CMD is the type of command to be transmitted;
- ARG is the argument of that command;
- CKS is the checksum of the data sent;
- ETX is the End Frame.

The transmission between SCS devices is performed using different types of frames. The command can be sent to a single device (*point to point* communication) or in *broadcasting* to a set of devices (using area, group or general frames). In case of *point to point* transmission, the receiving device sends an ACK after having correctly received the frame. In the broadcast case, receiving devices do not answer with an ACK message. Broadcast frame is sent on the bus three times. Basically the system uses short frames; however in order to introduce new functions, extended frames are being

progressively introduced.

These three types of frames are explained as the following :

Point to point frame(PP): A point to point frame is a frame sent from a device addressed to another. The receiving device is expected to send an acknowledge message in order to inform the sender of the message that the frame was successfully received. The acknowledge message is a single byte equal to the fixed value of 0xA5 and must be received by the transmitter within a 1,664ms period after the transmission was ended. If no ack is received the frame must be repeated up to 8 times;

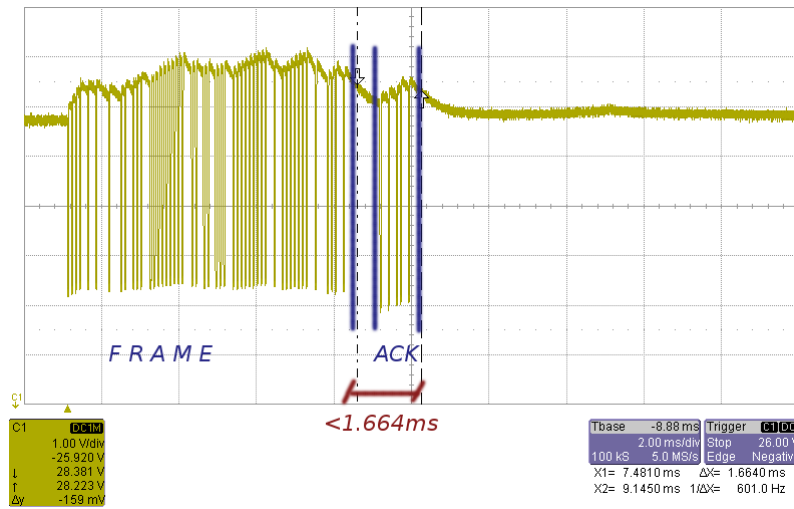


Figure 2.3: Point to Point frame

if no ack is received after the 8th repetition, the device is considered out of order or not present. Before each repetition, the transmitter MUST wait its free-Bus time before transmitting. Brief point to point frame timing:

$$9.56\text{ms} + T_{\text{waitFreeBus}} \Rightarrow 14.5\text{ms} < T < 46\text{ms}$$

Brief Broadcasting frame(BB): A broadcasting frame is sent from a device without any confirm of the reception. The frame is repeated 3 times at a 3.12ms interval. Brief Broadcasting frame timing:

$$30\text{ms} + T_{\text{waitFreeBus}} \Rightarrow 35\text{ms} < T < 62\text{ms}$$

Extended Broadcasting frame(EXT): This type of frame is long frame and broadcasting frame. Note that extended frames can only be trans-

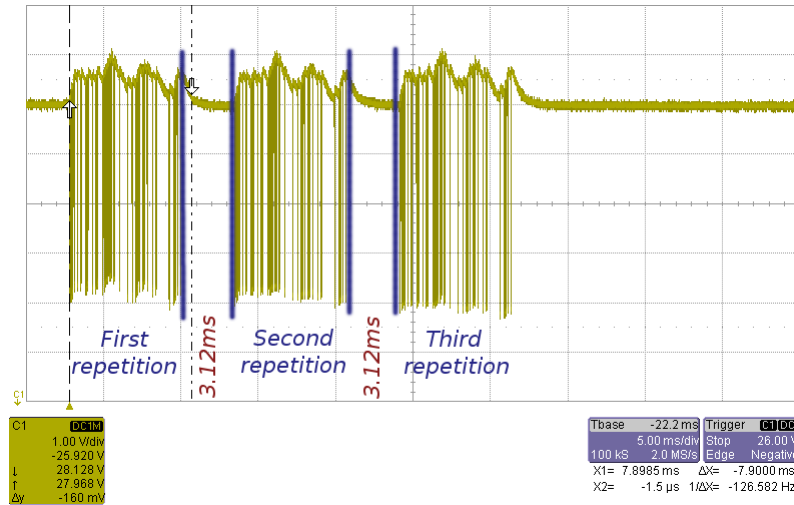


Figure 2.4: Broadcasting frame

mitted using broadcasting method! Extended Broadcasting frame timing:

$$43.5 + T_{waitFreeBus} \Rightarrow 48.5ms < T < 80ms$$

2.2.3 Collisions

Inside SCS protocol there's a system to resolve possible collisions [6]. A collision is a situation where more than one device try to transmit at the same time or when a disturb occurs during transmission. If the device transmitting don't read the same it is writing a collision is detected and the free-bus time is waited again before restarting the transmission of the frame. In figure 2.5 it is an example where a artifact is generated on the bus during a transmission.

2.2.4 Bus Access Time

The transmission protocol in BTicino SCS bus is the CSMA/CA. Therefore, each device before transmits a frame on the bus, must wait a random time between $5.2ms$ and $31.72ms$ (to avoid collision). Based on the type of device, there are different cases:

- Bus access time for sender devices (commands, scenario commands, special commands, etc.)
- Bus access time for receiver devices (dimers, actuators, etc.)

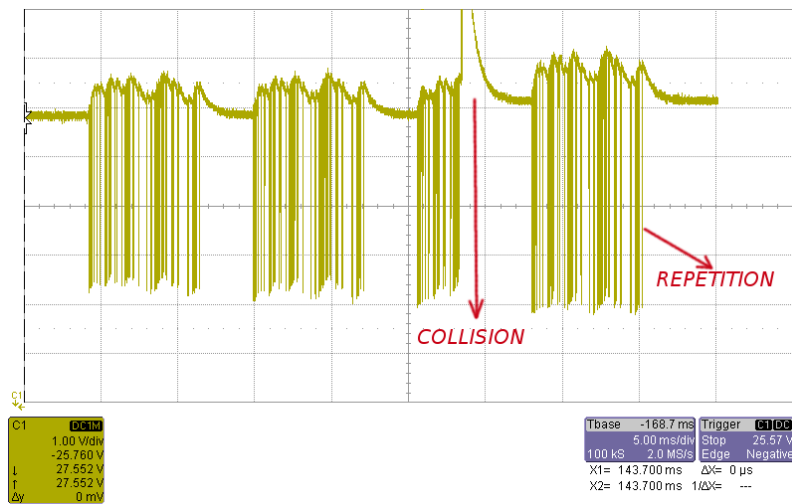


Figure 2.5: Collision example

- Bus access time for special devices (scenario modules, SCS-SCS interfaces, IR receivers, radio receivers, contact interfaces, etc.)

2.3 SCS information

SCS - Power Supply: The Supply is given through a power supply that normally ranges from 18 to 28 volts. Devices and wires/plugs connected to a power supply is named segment. The total segment sinking depends from the used power supply: typically 1.2A or 0.6A.

SCS - Voice: Over SCS cables is available a low quality mono-channel audio. The covered band is 300Hz-3,4KHz and the signal is sent base-band. The maximum signal on bus is 660mV peak-to-peak while typically it is 10-30mVrms while normal speech.

SCS - Audio: Over SCS is available an High quality stereo audio channel having these features:

- 20Hz - 20KHz @3dB band
- FM modulation using 70KHz of deviation at maximum input
- Left modulated at 5.5MHz
- Right modulated at 6MHz
- Max amplitude 100mVrms

SCS - Video: The video signal supported by SCS has these features :

- PAL-BG (625 lines - 576 visible, interlaced)
- AM modulated at 38.9MHz with amplitude 560mVpp

Data: Data are exchanged over a half-duplex channel having bit-rate of 9600 bit/sec.

2.4 My Home Automation system

The SCS Automation system [7] allows you to manage functions in a simultaneous and integrated way. These functions are:

- Lighting control
- Activation of rolling shutters, fans and exhausters
- Air-conditioning and heating
- infrared controls
- Scenarios

Thus, it consists of a series of devices (for controlling and actuating all the loads in a house).

These elementary devices, simple controls and actuators, are used with other devices which allow you to achieve advanced functions, which represent the fundamental principle of a domestic automation: demotic units, telephone dialing devices, control panels, visual and audible signal and much more.

Naturally, the SCS automation system can be combined with all SCS systems (burglar alarm, video door-entry systems...). It also includes an interface with traditional control devices for serial ports (RS-232) or USB, telephone lines and interfaces for specific buses, e.g. EIB.

The SCS Automation system is basically an open system which provides a number of possibilities for expanding and changing the functions required, hence it is intended for service/industrial sectors in which it is often necessary to have an extended management of complex functions such as, lighting and motor-driven rolling shutters.

However, the potentialities of this system are also evident in a residential environment because it allows you to increase comfort and safety conditions, in addition to an extended automation of traditional functions such as the light controls and the electric activation of rolling shutters. The possibility to be interfaced with traditional controls, combined with structural wiring

simplicity, makes this system suitable for being used in historical and prestigious environments, hence specific artistic restrictions are required.

2.4.1 Architecture of the system

The basic system includes *control* and *actuating devices*. *Control* devices basically replace traditional devices such as *switches*, *two-way switches* and *push buttons*, but they can also execute new and more complex functions; instead, *actuators* are devices which, as with traditional relays, control the connected load after giving a specific command. There is almost no need to say that all system devices are connected in parallel to the *SCS* cable with removable terminals. The required cable must be sheathed, unshielded and depolarized. The *actuators* must be connected to the BUS cable and to the power line of the relevant controlled load.

Control devices also include items such as scenario units (item *F420*), which allow you to simultaneously use several actuators selected by the user. As a result, it is possible to program the (self-learning) device with a series of controls which allow the user to set up a specific environmental condition (*scenario*). The condition will be re-created when the corresponding activation key on the scenario unit is pressed. As mentioned above, the Automation system also makes it possible, via the interfaces, to use traditional devices or PC platforms, thus allowing for the opening and flexibility of external systems.

These interfaces include two types:

- *Level one interface*: represented by a device to which traditional devices such as thermostats, control devices, wind detectors etc., are connected.
- *Level two interface*: represented by a device which allows the connection via an RS232 connector to a PC used to control and monitor the conditions of the BUS logic devices.

2.4.2 Automation commands

Following the logical subdivision used for the SCS Automation system, different types of command can be generated and interpreted, according to the set of devices involved. Control devices (senders) allow for the activation of actuators (receivers) with the following procedures:

Point-point controls: Controls directed to one actuator identified by an "environment number" and by a "light point number."

Command	GEN/AMB	PP	GR
Rise up (Stop + up)	2 BB	2 BB + 2 PP	2 BB + 2n BB
Stop	1 BB	BB + PP	BB + n BB
Rise (Stop + down)	2 BB	2 BB + 2 PP	2 BB + 2 n BB

Table 2.1: Automation's commands- Shutter system

where PP is Point to Point frame, BB is Brief Broadcasting frame and EXT is Extended Broadcasting frame.

command	GEN/AMB	PP	GR
On	1 BB	BB + PP	BB + n BB
Off	1 BB	BB + PP	BB + n BB
Dimmer 10 On %	1 BB	BB + PP	BB + n BB
Dimmer 10 Off %	1 BB	BB + PP	BB + n BB
Dimmer 10 Reg up	1 BB	BB + PP	BB + n BB
Dimmer 10 Reg down	1 BB	BB + PP	BB + n BB
Dimmer 100 On %	1 EXT	2 EXT	EXT + n EXT
Dimmer 100 Off %	1 EXT	2 EXT	EXT + n EXT
Dimmer 100 Reg up	1 EXT	2 EXT	EXT + n EXT
Dimmer 100 Reg down	1 EXT	2 EXT	EXT + n EXT

Table 2.2: Automation's commands- Lighting system

where PP is Point to Point frame, BB is Brief Broadcasting frame and EXT is Extended Broadcasting frame.

Room controls: Controls directed to all the actuators identified by the same environment number.

Group controls: Controls directed to all the actuators identified by the same group number, although they may belong to different environments.

General controls: Controls directed to all the actuators of the system.

The point-point control always requires a confirmation to make sure that the message has been received (ACK) by the receiver.

The commands of the shutter are shown in table 2.2.

The commands of the lighting system are as the following: Where **n** is the number of activators in a group.

2.5 MY HOME Thermoregulation System

In my Home system there is a Temperature control system [11] with zone control which gives the possibility to manage temperature in each environment (room) depending to the user's needs. It can be used:

- **WHEN THE SYSTEM IS PRESENT FOR ZONE MANAGEMENT:** a system is preset for zone management when the solenoid valves for the management of each zone can be installed on the distribution collector.
- **WHEN THE SOLENOID VALVES USED ARE:** The solenoid valves that manage the zones can be of two types:
 - With ON / OFF contacts
 - With open / close contacts

The MY HOME temperature control system can also manage the circulation pumps.

- **IN SYSTEMS WITH:** Radiators, Fan-coil, Radiant panels
- **WITH SYSTEMS OF:** Heating, Cooling, Both
- **IN CENTRAL HEATING SYSTEMS WITH DISTRIBUTION COLLECTOR FOR EACH HOME:** In central heating systems where each Home is fitted with its own distribution collector, it is possible to:
 - By installing a meter on the collector input, measure the amount of heat used
 - By installing solenoid valves, to manage the different zones of the Home

2.5.1 Operation mode

Periodically (every 15 minutes), the central unit sends at the probe the commands with the indication of the temperature to set and the operating mode (heating or cooling). The probe answers at central unit command (or a request of a control device) sending on the bus the measured temperature of the environment; in this way on the central unit is possible show the temperatures in the various zones constantly updated. Then the probe actives or deactivates the actuators based on the program set by the central unit, and based on the local setting (chosen by the user through a knob)

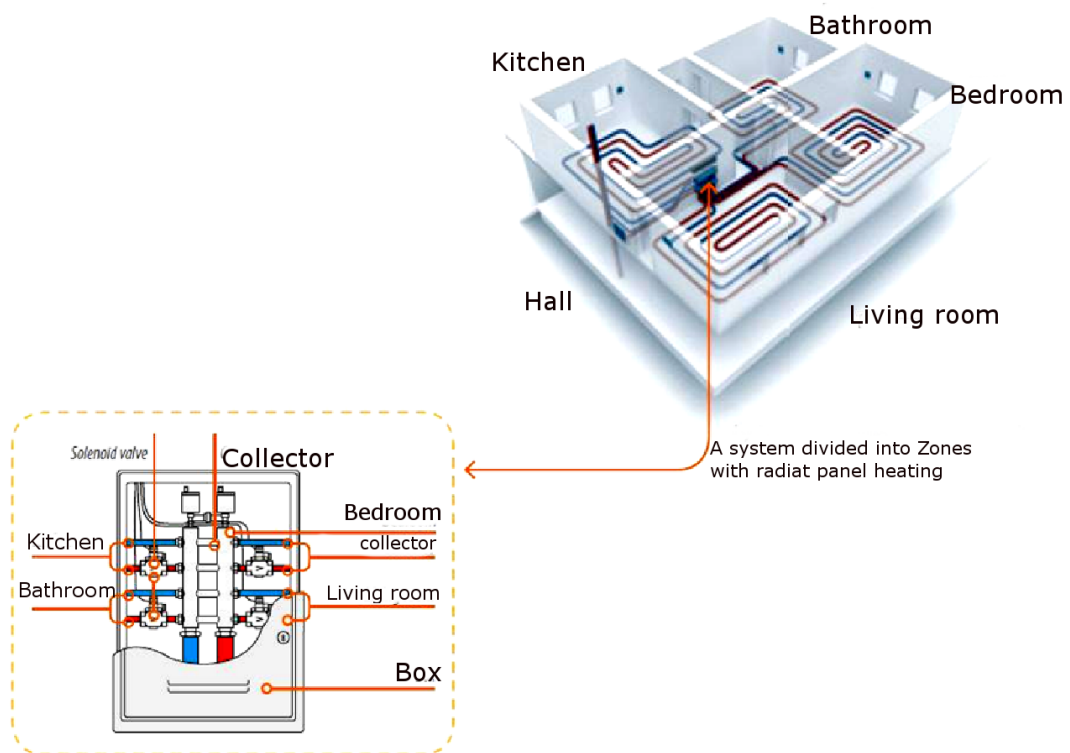


Figure 2.6: Thermoregulation system schema

and on the temperature detected. This "distributed" management of the system permit at the probe to continue to set the temperature in the own zone although a malfunctioning of central unit; in this way the probe works as thermostat based on the last temperature setting received.

Probe: Probes must be installed in each zone, so that the temperatures of the rooms can be detected and if necessary locally changed, in relation to the temperature set to the control unit.

The central unit: The central unit programming, provided by graphic display, permits to set the system (instead of the setting via PC) and to modify the operating mode of the system. It's equipped of a management software with a menu that permits to the user to choose the operating mode (automatic, manual, scenario ...) show the temperatures in the various zone, show and modify the various profile of daily temperature and weekly program, while the remaining voices of menu (protect by password) are reserved at the installer and permits the

access at the system (zone configuration, diagnostic, setting for the remote control of the system ...)

The central unit can work as heating or cooling. The passage from the heating mode to cooling mode (and vice versa) is done manually by the user, that can decide according to the weather conditions (variable from year to year) when active the necessary functions. It's provided the possibility to connect a contact in input for the remote control (commutation from the antifreeze mode to automatic mode and vice versa) through telephonic actuator. The central unit manages no more than 99 different zones (with a Master probe for each zone, more eventually slave probes); in addition, it performs collecting function and alarm / events management and has the possibility to realize a diagnostic test of the system.

Temperature sensors: They are sensors that not control a zone but that are delegate only at the measure of a temperature and send it on the bus. They can be used as generic thermometer, to measure for example the extern temperature or the temperature of sanitary water. The difference from the slave probe (whose behavior it's the same of thermometer), it's can manage also the negative temperature.

Working principle: A Master probe can control a zone with a maximum of 9 actuators, 8 slave probes and 9 pumps. The knob (if present) permits the local setting of the temperature (within the limits of +/- 3°C respect at the value setting by the central unit), the forcing in antifreeze / thermic protection and OFF.

The Speed key permits to choose the desired speed if fan-coil actuator are used. The management of the modality is done according to the follow standards:

- The OFF mode has the maximum priority, both is locally selected and is remote selected; this means that if a probe is locally force in OFF, it's not possible command it from remote, however is always possible force in OFF a probe both through remote and locally in any moment.
- The antifreeze / thermic protection mode can be forced only if the probe is not in OFF; both local forced and remote, has priority on the ON mode (heating / cooling), than if the local selector is in antifreeze is not possible set in ON the probe form remote, whereas if is set in antifreeze from remote remain also if locally is selected one of the ON positions (-3 ... +3); from remote the antifreeze commands and

heating protection are different; in local are not represented by the same position at the selector (*): if the last set mode was heating the probe work in antifreeze, if was in cooling the probe works in heating protection.

- To avoid that a local forcing limit the control possibility form remote of a probe, it's expected an unlocking command that cancel the forcing in OFF or local antifreeze / heating protection; when receive this command, the probe back to work a the last mode setting from remote and it's possible modify this mode as if there was no local forcing; the probe follow the program setting by the central unit until it's operated on the selector.
- The probe read always the frame that set the modality, also if for local forcing can not executed immediately; for example, if you receive a antifreeze command while it's forced in OFF locally, it remains in OFF, but pass in antifreeze as soon as the local force is terminated; at the same mode the frames that set cooling or heating are always recognized, maintaining aligned the probe during the forcing phase too.
- At the reset, the probe is in antifreeze with reference temperature +7°C (except a local forcing in OFF that has priority).

The loads management has the following rules:

- The loads implementation (switching on or off) is completely controlled; the probe sends the command and receive the state update from the actuators checking if the command is correctly executed; in the case an actuator not respond (in 10 seconds), the command is repeated and, once the probe not receive an answer, is set in OFF sending at the central unit the malfunctioning situation.
- Where is requested a pump piloting, the actuator switching on (valve opening) is executed (eventually after a delay) by the "Zone ON" signaling for pump actuator, while the switching OFF (valve closing) can be done only after request and get the (permission) of switching off of the pump; if the pump not answer, after two communication attempts the probe is OFF, it does 2 others attempts, signal the malfunctioning at the central unit and force the valve in closing.

Command	Frames
Display State	6 BB + 4 EXT (*7 BB + 5 EXT)
Change position of knob probe	1 EXT
Change zone's setting	2 EXT + 2 BB + 2 n BB + (1BB + 1EXT)
Change mode of central	2 EXT + $\frac{1}{4}$ Background Load
Change speed of fan coil (by probe)	1 EXT

Table 2.3: Thermoregulation's commands

where PP is Point to Point frame, BB is Brief Broadcasting frame and EXT is Extended Broadcasting frame.

2.5.2 Thermoregulation's commands

2.6 My Home Safety Burglar Alarm System

This system is designed to protect home by activating alarm in case of a danger by means of audible, luminous and telephone alarm [8]. It is equipped with some sensors which are sensitive IR detectors and magnetic contacts. This system has an separate SCS Bus which is connected to the main bus of the level three where other systems are using the shared bus.

This system is constituted by complex electronic equipment such as to detect and report:

- attempts of undue intrusion into the protected areas system is activated;
- tampering with system devices on or off;
- Technical alarm.

The planning setup of the system must include the following parameters:

- determination of the area to be protected
- position, number and type of detectors
- location of the control and information repetition devices
- position, number and type of local alarm devices
- position and interconnections with broadcasting devices, if any
- classification of the cables and type of laying

Detectors represent the characteristic element of the system; therefore they must be selected according to the performances to be achieved and to the structure of the environment to be monitored.

In particular, it is necessary to give special attention to the location of the access points (doors, windows, skylights, etc.). The various detecting devices are all in constant communication with a unit, which represents the heart of the system. Apart from constantly verifying the absence of trespassing signals, the unit controls the alarm conditions, if any, by performing all the operations established by the user during the system setup.

Other devices let you interface the system with transmission media which can remotely signal when the alarm goes off. Audible signal devices (alarm sirens) are also part of the system. The system can also separate the control, in other words it can divide the area to be protected into different zones. By doing so, it is also possible to use the system when in the presence of people by activating the system only in unattended places. It is possible to have a maximum of eight independent zones.

2.6.1 Operation mode

Periodically there are some frames which are transmitted in the SCS bus of the Burglar alarm but whenever there is any command lunched by user, the frames after passing in the mentioned Bus, go in the level 3 Bus as well. For each device there are two point to point frames each 4 seconds which are sent by the mean of background load in the burglar alarm.

- *Tamper control cycle*: In order to verify the proper functioning of the bus and of the devices, the unit periodically queries (every 4 seconds) all the objects connected to the bus so that it will be possible to become aware of trespassers cutting or tampering with cables within a maximum time period of 4 seconds. Every device, when queried, responds with the tamptest unless its tamper device is in operation. The operational devices (siren, telephone dialing device, relays...) go off if they do not receive a periodical request within 10 seconds.
- *SYSTEM LIMITS*: The maximum number of devices is 40 inasmuch as the system queries every 100ms per 4 seconds.

Command	Frame in BUS Burglar Alarm	Frames pass through level 3
insertion	3 PP + 4 BB + 1 EXT	3 PP1 + 4 BB + 1 EXT
Insertion + error insertion	3 PP + 12 BB + 4 EXT	3 PP1 + 12 BB + 4 EXT
deactivation	3 PP + 5 BB + 1 EXT	3 PP1 + 5 BB + 1 EXT
Alarms	1 PP + 3 BB + 2 EXT	3 BB + 2 EXT
choking	5 PP + 2 BB + 1 EXT	5 PP1 + 2 BB + 1 EXT
choking+ Insertion(TS3.5")	8 PP + 4 BB + 2 EXT	8 PP1 + 4 BB + 2 EXT
choking + errors Insertion (TS3.5)	8 PP + 16 BB + 6 EXT	8 PP1 + 14 BB + 5 EXT + 8 EXT

Table 2.4: Burglar Alarm's commands

where PP is Point to Point frame, BB is Brief Broadcasting frame and EXT is Extended Broadcasting frame.

2.6.2 Burglar Alarm's Commands

2.7 My Home Sound Diffusion system

This system is the one dedicated to multimedia distribution into the house [10]. In particular the treated signals are: VOICE: A mono audio signal with low bandwidth (300Hz-3KHz) transmitted in base-band; AUDIO: A stereo audio signal with High quality (20Hz-20KHz) transferred upon modulated carriers; VIDEO: A video signal with low spatial definition (PAL). Interactions are expected with video door entry system because both systems manage voice and video signals. The possible components which are needed are as following :

- Audio/Video mixers
- Sound sources
- Control Devices
- Sound Amplifier
- Loud Speaker

2.7.1 Brief description of functions

Multimedia systems can be distinguished into three different categories: - Flat - Audio multiplied - Audio/video multiplied. Flat systems are characterized by a unique audio (HiFi) and a unique Video signal at a certain

moment in the whole plant.

Audio multiplied system has always an unique Video signal in the whole installation while can have different audio (HiFi) signal in different branches. Audio/video multiplied system are plants where a whatsoever branch can have a different audio (HiFi)/video signal couple from another. It has to be well noticed that voice signal is always the same in all the system whatever its configuration is.

2.7.2 Sound Diffusion's Commands

Commands	Frame in BUS
Switching on PP	8 BB
Switching on AMB	6 BB + 2nAMB BB
Switching on GEN	6 BB + 2n BB
Switching off PP	2 BB
Switching off AMB	1 BB + nAMB BB
Switching off GEN	2 BB + n BB
Change track	2 BB
Increase Volume PP	2 BB
Increase Volume AMB	1 + nAMB BB
Increase Volume GEN	1 + n BB
Request State and Volume AMB from TS	2 BB + 2 nAMB BB
Request paging	2 BB + n BB
Source cycles	6 BB
Request Frequency Radio	2 EXT

Table 2.5: Sound Diffusion commands

where PP is Point to Point frame, BB is Brief Broadcasting frame and EXT is Extended Broadcasting frame.

2.8 My Home Energy Management system

The Energy Management system [9] is designed to avoid the increasingly usage of electricity when frequent use can trigger the electricity meter due to overload and therefore disconnecting the power.

This system controls the amount of absorbing electricity by the current load inside of the house and automatically disconnecting of the load which are not so important. The usage amount is controlled by a specific unit which triggers activators when the load is close to overload condition. More ever it is possible to manage the energy system by the having a time plan which

is settled up by the user to have better utilization of the power contract during low price time-period.

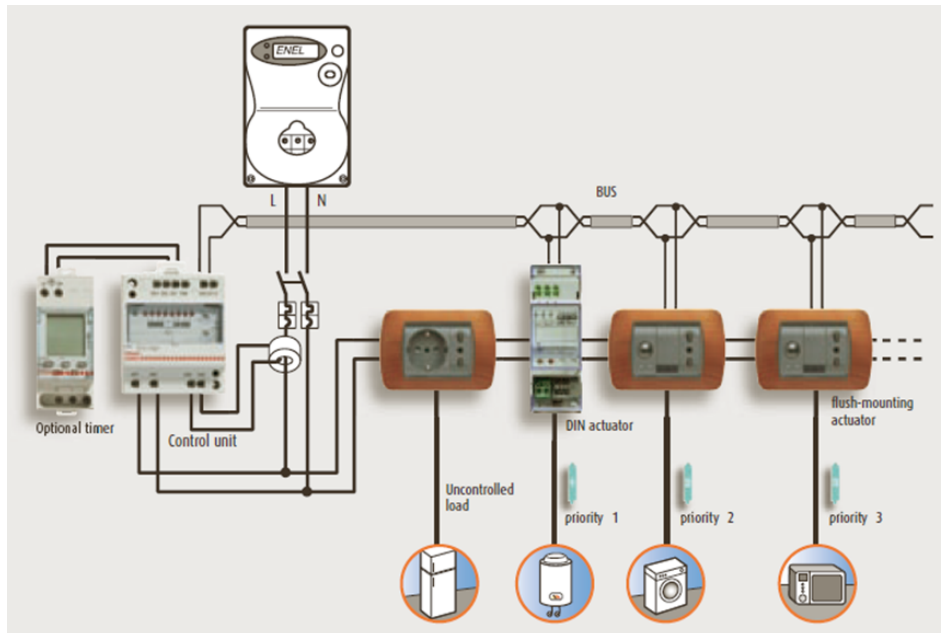


Figure 2.7: Energy Management system schema

2.8.1 Notes of the system

A central Energy Management can fuel a maximum number of four actuators and a display panel. In the case where the number of loads to be controlled is greater than four, is necessary to insert into a power supply system similar to that used for the automation system. In case the supply center is no longer fed from the mains at 230V ac, but it also exploits the power 27V dc bus, absorbing a maximum current of 10mA. In this case the system may provide up to 16 actuators, which absorb a maximum current of 320 mA totals, a value very shorter than the SCS feeder.

The maximum number of connectable devices on the bus depends on the absorption total of the same and the distance between the connection point and the power supply.

If the system shares the same cable system automation, then the calculation of the limit number of units must be done keeping in mind the general absorption of the devices of both systems.

It 'still need to keep in mind that in the presence of a power supply, the

maximum number of actuators of the Energy Management System which can be connected to the bus is equal to 16, while in the absence of the power supply (The voltage is supplied from the Central Bus Management Energy) that number drops to 4. The operating temperature of the system is between -5°C and 40°C .

2.8.2 Energy Management commands

- METER (F520-F521 frame per TA): TA sarebbe la linea di misura (ovvero il toroide).the commands are shown in table 2.7.
- LOADS ,as it shown in table 2.8.

Command	Frame in BUS
Request instantaneous consumption	3 EXT (+ 1 EXT each 30 s)
Request consumption of partial current day	3 EXT
Request consumption of partial current month parziale mese corrente	3 EXT
Request monthly cumulative consumption	4 EXT
Request daily profile of hourly	20 EXT
Request monthly profile on a daily basis	46 EXT
Request monthly averages of hourly	20 EXT
Request Profile last 12 months	48 EXT

Table 2.6: Energy Management- METER commands

where PP is Point to Point frame, BB is Brief Broadcasting frame and EXT is Extended Broadcasting frame.

Command	Frames in BUS
Enabling /Disabling	2 EXT
forcing /Fine forcing	2 EXT
State request	2 EXT
Request Totalizer	6 EXT
Reset Totalizer	6 EXT
Differential current level	2 EXT

Table 2.7: Energy Management- LOADS commands

Chapter 3

BACKGROUND - RELATED WORK

“ My religion consists of a humble admiration of the illimitable superior spirit who reveals himself in the slight details we are able to perceive with our frail and feeble mind.”

Albert Einstein

In this chapter some background information are provided for the reader which helps him to understand the rest of documentation better. Queuing network and the tools which have been used in this project are described.

3.1 Queuing Network Model

During the many years ago, Computer and communication systems have been studied as a network of queues[16]. According to [17] Queueing network modelling is a particular approach to computer system modelling in which the computer system is represented as a network of queues which is evaluated analytically. A network of queues is a collection of service centers, which represent system resources, and customers, which represent users or transactions. Here there are some definition in queuing network:

Class A class is a group of customers with each customer of that group having the same workload intensities (A_c , N_c , or N and Z_c) and service demand ($S_{c,d}$). A class MUST define the service demand ($S_{c,d}$) at each device used [17] [24].

Throughput Throughput is the number of customers serviced by a device. x_j is device throughput [17] [24].

Response Time Response Time is the time a customer spends at a device r_i [17][24].

System Throughput System Throughput is the number of customers serviced by the system within a given period of time. $X = \sum_j x_j$ [17][24].

System Response Time System Response Time is the sum time a customer spends at each device. $R = \sum_i r_i$ [17][24].

Device Utilization Device Utilization is a ratio between the busy time (time a device is being used) and the available time of the device. This ratio describes the extent of utilization of a device. Utilization is defined as less than equal to 1.0; therefore, 0.5 utilization means half of the device's possible capacity is being used [17][24].

Arrival Rate Arrival Rate is the rate of customers arriving at a queue per unit time [17][24].

Think Time Think Time represents the time a user spends before submitting a customer [17][24].

Queue Length Device Queue Length is the number of customers waiting for service in a particular queueing center [17][24].

A QN can be open, closed, or mixed depending on its customer classes: open if all classes are open, closed if all classes are closed, and mixed if some classes are open and some are closed [13].

Open Model A Queueing Network is open if customers arrive from outside the system. Arrival rate(λ) denotes arrival intensity.

Closed Model A Queueing Network is closed if customers are generated inside the system (such as in a terminal system). In closed classes, the number of class c terminals (N_c) and think time for each class c terminal denote workload intensity (Z_c).

Mixed Model systems that have both open classes (external input) and closed classes (such as cycle events of keyboard input) [13].

Mean Value Analysis(MVA) Mean Value Analysis (MVA) [13][17] is a technique for analyzing queueing networks that represent computer systems. MVA provides a developer with important performance measures such as Response Time, Throughput, Queue Length, and Device Utilization. Instead of solving a set of simultaneous linear equations to

find the steady state probability of being in each system state, from which performance metrics (e.g., system throughput, device utilizations, user response times) can be derived, MVA is a simple recursion. MVA calculates the performance metrics directly for a given number of customers, knowing only the performance metrics when the number of customers is reduced by one. The recursion is intuitive, efficient, and elegant. Its impact on the field of analytical performance evaluation has been huge.

3.2 Network simulation

Simulation modeling is becoming an increasingly popular method for network performance analysis. Generally, there are two forms of network simulation [13]:

- Analytical Modeling
- Computer simulation

Analytical modeling is conducted by a mathematical analysis that characterizes a network as a set of equations. The main disadvantage is its overly simplistic view of the network and inability to simulate the dynamic nature of a network. Thus, the study of a complex system always requires a discrete-event simulation package, which can compute the time that would be associated with real events in a real-life situation. A software simulator is a valuable tool, especially for today's network with complex architectures and topologies. Designers can test their new ideas and carry out performance-related studies, thus freeing themselves from the burden of "trial and error" hardware implementations.

A typical network simulator can provide the programmer with the abstraction of multiple threads of control and inter-thread communication. Functions and protocols are described either by finite-state machine, native programming code, or a combination of the two. A simulator typically comes with a set of predefined modules and a user-friendly GUI. Some network simulators even provide extensive support for visualization and animation [13]. In this thesis the JMT tool[4] has been used to model and simulate the queuing network of the SCS bus of the *BTicino S.p.A.* system.

3.3 Java Modeling Tool (JMT)

In the manual of the JMT [2] has been explained that The Java Modeling Tools (JMT) [21] is a free open source suite consisting of six tools for performance evaluation, capacity planning, workload characterization, and modeling of computer and communication systems. The suite implements several state-of-the-art algorithms for the exact, asymptotic and simulative analysis of queuing network models, either with or without product-form solution. Models can be described either through wizard dialogs or with a graphical user-friendly interface. The workload analysis tool is based on clustering techniques.

The suite incorporates an XML data layer that enables full reusability of the computational engines.

The JMT suite is composed by the following tools:

JSIMgraph A graphical user-friendly interface for the same simulator engine *JSIM* used by *JSIMwiz*. It integrates the same functionalities of *JSIMwiz* with an intuitive graphical workspace. This allows an easy description of network structure, as well as a simplified definition of the input and execution parameters. Network topologies can be exported in vector or raster image formats.

JMVA For the exact analysis of single-class or multiclass product-form queuing networks, processing open, closed or mixed workloads. A stabilized version of the Mean Value Analysis *MVA* algorithm is used. Network structure is specified by textual wizards. What-if analysis and graphical representation of the results are provided.

JABA It applies a simulation technique to solve a single station model, with finite (M/M/1/k) or infinite queue (M/M/1), and shows the underlying *Markov Chain*. It is possible to dynamically change the arrival rate and service time of the system.

JWAT Supports the workload characterization process. Some standard formats for input file are provided (e.g., Apache HTTP and IIS log files) customized formats may also be specified. The imported data can initially be analyzed using descriptive statistical techniques (e.g, means, correlations, histograms, boxplots, scatterplots), either for univariate or multivariate data. Algorithms for data scaling, sample extraction, outlier filtering, k-means and fuzzy k-means clustering for identifying similarities in the input data are provided. These techniques allow the identification of cluster of customers having similar characteristics.

The clusters centroids represent the mean values of the parameters of the classes (e.g., CPU time, n.o of I/Os, n.o of web pages pages accessed) that can be used for the workload parametrization. The figure 3.1 shows the starting window of the JMT. This starting window is

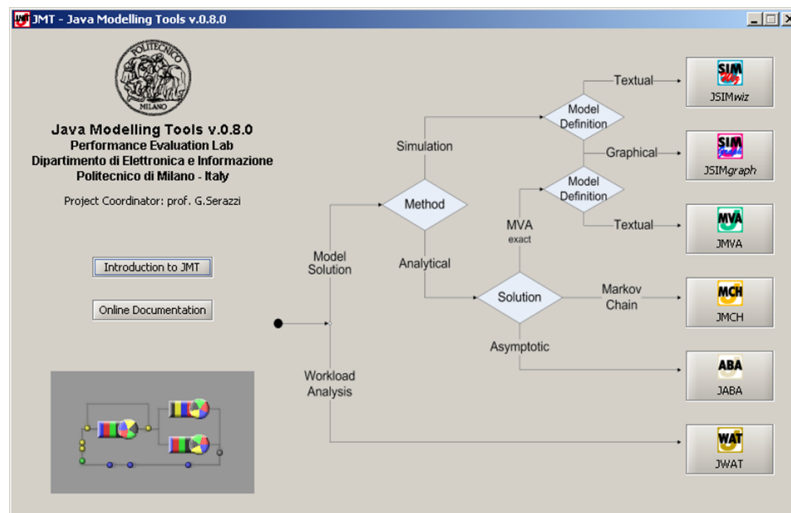


Figure 3.1: The starting window of JMT

used to select the application of the suite to be executed by clicking on the corresponding button. The flow chart should help the user to select the application that best fits its needs.

3.3.1 JSIM model simulation

As it mentioned in [4], JSIM the simulation module of the Java Modeling Tools (JMT), an open-source fully-portable Java suite for capacity planning studies. The simulator has been purposely developed to help both unexperienced and advanced users. Most of the difficult decisions that are needed in order to run simulations properly, such as the detection of the transient part of samples to be discarded, have been automated. The tool also provides guidance over the graphical design of the network and over the analysis and the plot of the results. What-if parametric analysis for parametric evaluation of complex systems are supported. Several features that increase the generality of the applications to capacity planning studies are provided, among them fork-join service centers, regions with finite capacity, state-dependent routing algorithms, priority classes and import of real workload distributions from log files.

As the goal of the project was to make model of the SCS Bus , the simulator has been used in this project. JSIMwiz and JSIMgraph both are the simulator part of the JMT and have the same engine. In this project during making models the JSIMgraph has been used and afterward for calling the JSIM engine to solve the model by the software which has been developed for generating the models, the JSIMwis GUI has been called. Here you can see an example of the JSIMgraph model which is for a LAN network which each part of the network is defined in the GUI model.

The figure 3.1 shows the starting window of the JMT. To understand how

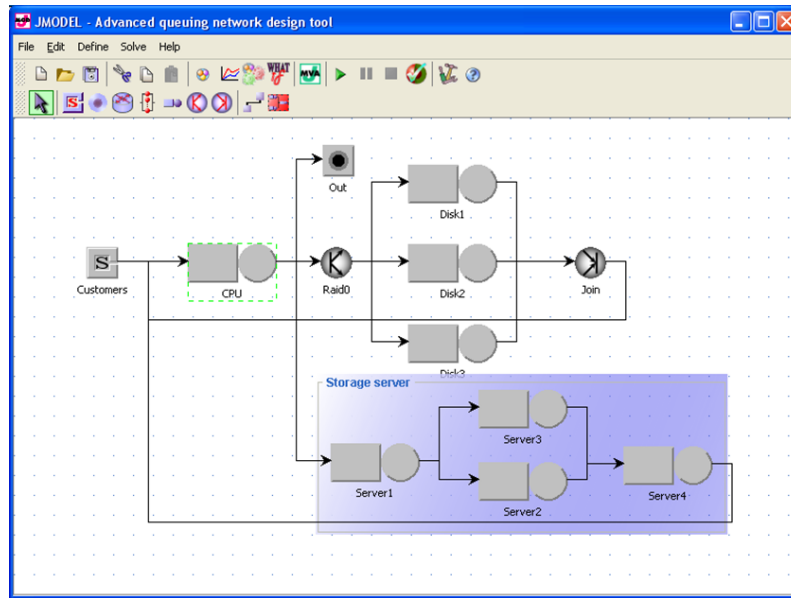


Figure 3.2: The starting window of JMT

this tool works here has been explained briefly how to define the model and the important parameters for the model, but for very complete information the reader can refer to the website of the JMT tool [4] where there are documentations of the project and also the forum where users could write their doubts about the tool. In each model, first the elements of the network should be simplified and be define in GUI of the JSIMgraph which is user-friendly to make it. Then for each elements the parameters should be define. The most important parameters which should be defines are :

1. Defining the classes of the system
 - (a) Closed classes
 - i. Population (number of the customers)

- (b) Open classes
 - i. Distribution of the arrivals
- 2. Priorities of the classes if exist
- 3. Service Time Distribution for delay and service centers
 - (a) Strategy
 - i. Load independent
 - ii. Load dependent
 - iii. Zero service time
 - (b) Service time distribution for each class
- 4. Routing strategies for each elements
 - (a) Random
 - (b) Round robin
 - (c) Probabilities
 - (d) Fastest service
 - (e) Least utilization
 - (f) Join the Shortest Queue
 - (g) Shortest response time
- 5. Queuing policy and capacity
 - (a) Queue policy
 - i. FCFS
 - ii. LCFS
 - (b) Drop Rule
 - i. Waiting queue
 - ii. BAS blocking
 - iii. Drop
 - (c) Capacity
 - i. Finite(number of customer)
 - ii. Infinite
- 6. Define performance indices

It should be mentioned that in the JMT, the distributions which are supported in JMT to define the distribution for arrivals and the service processes are :Burst(General, Modulated Markov Poisson Process MMPP2), Constant, Erlang, Exponential, Gamma, Hyperexponential, Normal, Pareto, Poisson, Student-T and Uniform.

The simulation result can be in two different ways which are as below:

Single simulation run result When the default simulation result by the given data has been requested.

WHAT-IF simulation result According to the manual of the JMT tool [44], a What-If Analysis consists of a series of simulations in which one or more control parameters are varied over a specified range. This allows the observation of system behavior under a spectrum of conditions, unlike the single simulation run where the system is observed under a fixed set of parameters. By default the What-If Analysis is not enabled. It must be activated explicitly and its parameters should be defined.

3.4 Getting the Real data (Log file)

For any performance analyzing it is important to check the result of the simulation with the real data which has been gathered from the system. In case of the SCS bus, the tool to log the system which is used is the *SCS viewer*.

3.4.1 The SCS viewer

This tool has been developed in *Bticino* to log the SCS Bus. By using this tool it is possible to see below information according to SCS bus:

- Frame
- Time (hh:mm:ss.ms)

The interface of the tool is shown in the figure 3.5.

This tool can be used for several goals such as sending special frame in the bus, writing codes and also an editor for editing the test files, but what was useful during this project was the logging feature.

In figure 3.6 you can see how it is the data provided by log file from SCS viewer. Where the last column is for the frames which have passed through the SCS bus.

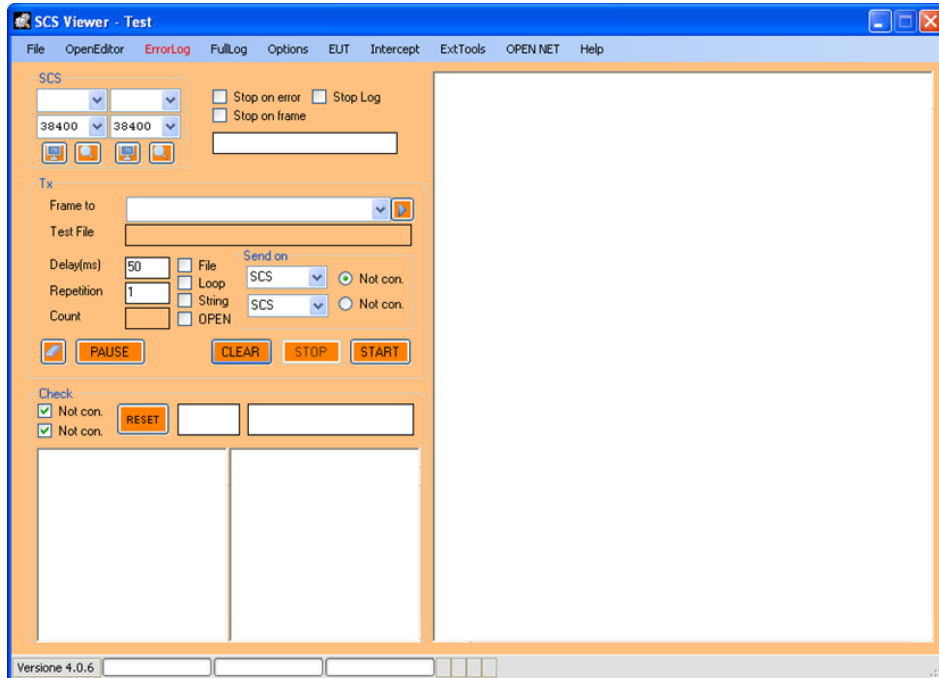


Figure 3.3: SCS Viewer Interface

```
RX COM1 - 31:12:2011 - 17:32:35.715 - D2000907316C2011 -  
RX COM1 - 31:12:2011 - 17:32:35.738 - D2000907316C2011 -  
RX COM1 - 31:12:2011 - 17:32:36.019 - D2000906173000FF -  
RX COM1 - 31:12:2011 - 17:32:36.035 - D2000906173000FF -  
RX COM1 - 31:12:2011 - 17:32:36.051 - D2000906173000FF -  
RX COM1 - 31:12:2011 - 17:36:45.305 - B502120A -  
RX COM1 - 31:12:2011 - 17:36:45.316 - B502120A -  
RX COM1 - 31:12:2011 - 17:36:45.324 - B502120A -  
RX COM1 - 31:12:2011 - 17:36:45.353 - B5021209 -  
RX COM1 - 31:12:2011 - 17:36:45.363 - B5021209 -  
RX COM1 - 31:12:2011 - 17:36:45.371 - B5021209 -  
RX COM1 - 31:12:2011 - 17:42:35.744 - D2000907316C2011 -  
RX COM1 - 31:12:2011 - 17:42:35.768 - D2000907316C2011 -  
RX COM1 - 31:12:2011 - 17:42:35.778 - D2000907316C2011 -
```

Figure 3.4: Log file of the SCS bus - taken by SCS viewer

Limits of the SCS viewer

This tool which we use for getting log has some limitations which are mentioned as the following:

- ***Limits in information:*** as it is mentioned above, the only information we can get from the log is the frame and time, there is no other information such as the type of the command, system utilization and etc.
- ***Time Accuracy:*** the time which is logged in the file is actually the PC time which is connected to the SCS bus to use the tool for logging, but the exact time (system time) of the system which frame on that time passed through SCS bus is different, therefore the time is not as accurate as it should be.
- ***Frame analyzer:*** to understand the meaning of each frame there is no any tool included in the SCS viewer and it should be done separately in another application. (for example in Excel which will be explained later). Understanding the frame is very important to analyze the parameters such as utilization which are useful for performance evaluation of the system.

Chapter 4

Queuing network Model

“Notice that the stiffest tree is most easily cracked, while the bamboo or willow survives by bending with the wind.”

Bruce Lee

4.1 Introduction

In this chapter we modeled the SCS bus system for level 3 of My Home system and then the model is extended for tree network and especially the load from other buses to level 3. Then we analyzed real data from the system which are provided in the LOG files and with the definition of the system some parameters have been chosen to compare the results. Later on, results from simulation is compared from the result from real system and the accuracy of model is discussed. At the end we have presented the Analytical model and methodology to compare it with the simulation.

To understand better the models, it should be mentioned that depending to the configuration of the My-Home system the parameters of the model should be changed, thus, an application has been developed to generate new model for new system considering the components in the planet which is explained in chapter 6.

As it has been explained in Chapter 2, there are different types of the systems such as Automation, Thermoregulation, Sound diffusion, Energy management and Burglar Alarm which depending to the need and request of the user can be installed in the planet, therefore the occupancy of the bus is related to number of the system an number of the devices which are installed in the planet and generates the load through the Bus.

4.2 Load in the system

Loads from each system is different, some systems generates the load with sending the frames through the Bus just when user start using the system, we call this load *User Action Load*.

Some systems not only have the former load when user used them, but also they should exchange information among them to keep the system updated and make sure *My Home system* works fine, therefore they have another kind of the load through the SCS bus which is not depend to the action from the user but it's communication among devices, we call it **Background Load**. Hence the load in the system or in the better words the occupancy of the SCS bus, in each portion of the time, is calculated by summing up the *user action load* on that moment and the *Background load* on that moment which it is the *Utilization of the bus*.

$$Occupancy_{BUS} = LOAD_{User} + LOAD_{Background}$$

4.2.1 User Action Load

When user wants to make a change (ex. turning on a light), a command should be used to send information(*frames*) to devices (ex. Activators). This information is included in frames which have been descried in chapter 2. To run a command, user should use some of the devices such as Command, Touch Screen, probe and etc. Each of these devices has different address and therefore the Time access to the bus for them is different.

In addition user can set scenario which will be launched automatically and it does user wills. Although the scenario will be run without any action from the user but in the definition of this thesis it has been considered as User action load as well, because has been defined by user and it's the same as user is doing each steps of scenario by himself.

4.2.2 Background Load

It is important to know why there are background loads and how they are generated. The background load is to keep the devices updated with the any changed happened in the system and also to check if any errors has been acquired in the system. There are three system ,that if they have been installed in the My Home system, they generate the background load. These systems are:

- Thermoregulation system

- Sound Diffusion system
- Burglar Alarm system

By the definition of the level 3 in My Home system, all these three systems are in level 3 and their frames pass through the SCS bus of the level 3. Each of these systems has been explained in the chapter 2, but here briefly there are explanations of their behavior regarding to the background load. The total background load is the summing up of the background load of each system.

$$BackgroundLoad_{TOT} = \sum_n BackgroundLoad_n$$

Where n presents the three system which make background load.

Thermoregulation Background Load

In this system there are *Master probes* and *Slave probes* and a central unit. The control and any change of this system can be done by the probes or by Touch screens. To keep the probes and touch screen updated by and any given change, some frames exchange between them in each 15 minutes. The background load for this system is based on the following parameters:

- Number of the Master Probes in the Zone
- Number of the Slave frame in the zone
- Number of the activators in the zone.
- Number of pompe activators.

To calculate the background load in Thermo regulation system ,it is needed to consider the worst case which all the parameters change each 15 minutes (therefore it insert a program to change all the parameters) or the best case when just they communicate to exchange the parameters of the temperature. It is estimated that in an hour there are one time worst case and three times the best case.

Therefore the value of the background load in the worst case is:

$$ThermoBL_{worst} = N_{Masterprobs} \cdot (12BB + 4EXT) + N_{slaveprobs} \cdot 6BB + N_{activators} \cdot 8BB + \sum_n NZU_n \cdot 8BB$$

-	Frame/h (worst)	Frame/h (best)
N master probe	12BB + 4ext *	8BB
N slave probe	Tx 6 BB	Tx 6 BB
Activators in zone	8 BB	8 BB
Pompa activators	8 BB x NZU **	8 BB x NZU **

Table 4.1: Background Load in Thermoregulation System

where:

BB = Broadcast brief frame and EXT =Broadcast Extended frame;

* Multiplied by 4 for the each 15 minutes;

** NZU number of the zone.

In our assumption we consider the 15 minutes load (not one hour) to estimate the good approximation of the Thermoregulation BL which is the simplified formula of the above one is as below :

$$ThermoBL_{Nframes} = N_{zones} \times 3 + \frac{3}{2}N_{Slave} + 2N_{Act} + 2[N_{Pompes} \cdot N_{zones}] \quad (4.1)$$

The thermoregulation system is a complex system, thus, there are two types of installations for thermoregulation system: -Normal Installation
-Huge Installation

With experiencing the real data from several system, the λ can be calculated as the following :

$$\lambda = \frac{ThermoBL_{Nframes}}{13(sec)} \quad (4.2)$$

where in this formula 13 second is the estimation time to send all the thermoregulation BL in each period of 15 minutes.

The below condition should be evaluated to recognize if the system is normal or huge :

$$\left\{ \begin{array}{ll} \lambda \leq 18 & \text{Normal Installation: Thermoregulation can be installed in L3} \\ \lambda > 18 & \text{Huge Installation: Thermoregulation should have its own SCS BUS} \end{array} \right\}$$

It is recommended to not install huge system as normal system in L3 because in each period of 15 minutes there will be a long time(e.g 30 seconds) that BUS is completely occupied by Thermoregulation BL, and if user use the system in these times, the system can not responde to any requests.

Sound Diffusion Background Load

In the sound diffusion system, the RDS are three extended frames in each second pass through the bus if there is a graphic and radio interface.

$$SoundDiffusionBL = \frac{3EXT.3600}{Totaltime(sec)} \quad (4.3)$$

Burglar Alarm Background Load

The background load (BL) for the Burglar alarm depends to the following parameters:

- sensors, Interfaces, (everything except the central)
- Central unit

For each device two frames every four seconds (one goes to the central unit and, one respond to the device).

$$AlarmBL = \frac{N_{devices} \cdot 2PP.3600}{4(sec)}$$

Note : the burglar alarm has a separated bus in level 3, and there is an interface between the bus of level 3 and the bus of the burglar alarm. The reason that this system has a separated bus is the power supply, by the reason to be able of adding another power supply an interface should be added and therefore another SCS bus is created, but this interface doesn't filter any User Action (UA) frame from the bus of Alarm system to bus of the level 3 and the frames which never goes to the level 3 are the background load (BL) of the burglar alarm. Hence in level 3 the total BL is just from the ThermoBL and SoundBL and the AlarmBL stay in its bus.

4.3 Queuing Network Model

In *My Home* system all the devices are connected to the SCS bus as it has shown in the figure 4.1 and 4.2. therefore we can say the *clients* in our model are the devices that generates the load, either *Background load* or *User Action Load*.

We mention again as we are interested in the occupancy of the SCS bus, we consider the SCS bus as our service center which other devices are connected

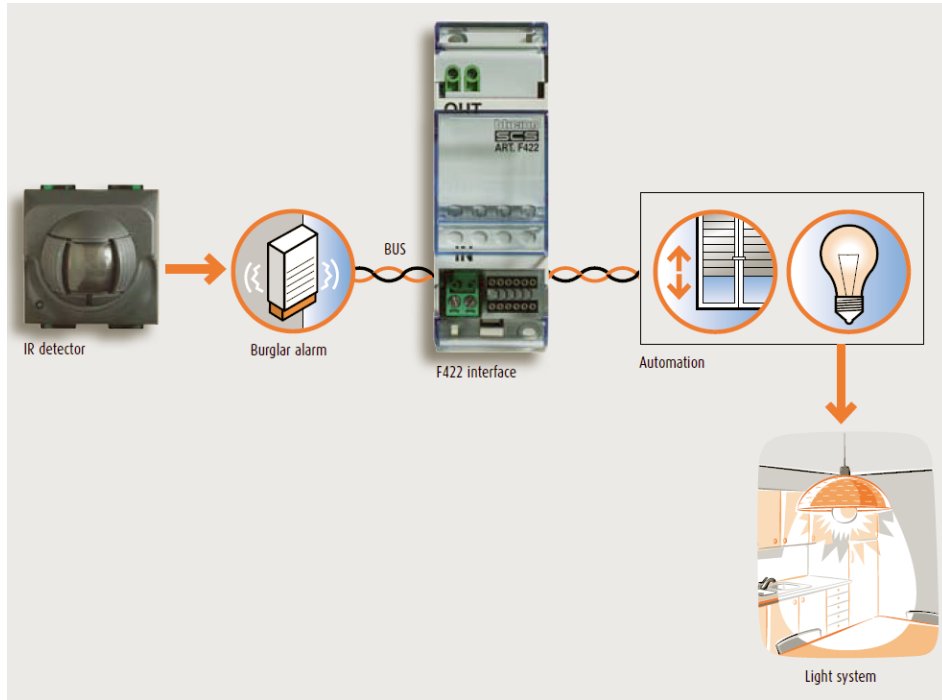


Figure 4.1: MyHome system - SCS BUS

to it. in addition between Buses we use *Interface* (similar to a Router) which separates buses and provides a way to use another power supply. For example, in Burglar Alarm system, which has a separated bus, all the Background Load remain in the SCS bus of Alarm system but all the User Action load pass through Level 3 , the SCS bus of Automation and Thermoregulation system.

the schema of the system can be drawn as figure 4.3 which is the basic and simplest ,but most used, model of My Home system. the schema can grows as tree shape network when we add other SCS bus for other levels. there are some rules related to number of buses could be connected to each other which has been explained in the system specification in chapter 3. We can define there are $N = (n_1, n_2, \dots, n_n)$ devices that are connected in the SCS Bus as service center S . In each exact moment just one device can use the BUS to transfer frame. As we know, we have three different type of the frames with different transmission time. each device to access to the bus should use the algorithm has been explained in chapter 3. According to the algorithm in the SCS bus CSMA/CA [22] method is used to avoid collision in the system. This time to access to the BUS can be assumed as

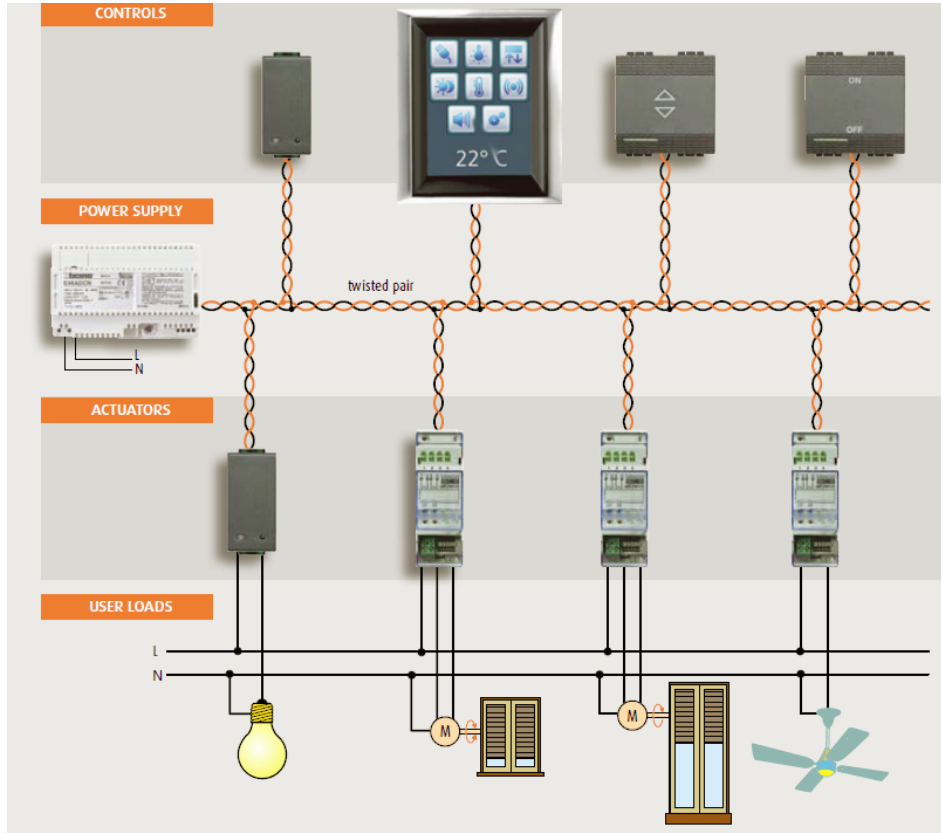


Figure 4.2: MyHome system - SCS BUS

random time between 5.2 ms to 31.72 ms where 5.2 is for device with highest priority (with address 00) and 31.72 is for the device with lowest priority (with address FF).



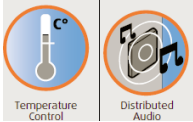




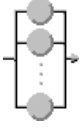

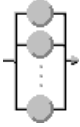
4.4 Modeling the SCS BUS using JMT

In this project first the level 3 SCS bus is modeled and it has been under consideration because it is the most crowded and occupied SCS bus in whole system. with this assumption we do not have any *Interface* in this model and we concentrate in single SCS Bus. With above explanation and assumption, we have some devices that are connected to each other using a twisted pair cable as Bus. Hence, we have a Delay center to simulate the devices (*Clients*) and a BUS (*service center*). To separate *Background load* from *User action load*, a **Mixed Queuing Network** is defined [15]. The open Model presents *Background load* and the Closed model presents *User*

action load.

First, we should model each component of the system into the queuing network elements. In chapter 2 the elements of each model in JMT have been described where they are: Source, Sink, Delay station, Service station and etc. In table 4.1 each component of system is modeled to one of the elements in JMT.

Table 4.2: Components and their model

 <p>SCS BUS</p>	 <p>Service Center</p>
 <p>Source of Background Load</p>	 <p>Source</p>
 <p>Sink for Open Model</p>	 <p>Sink</p>
 <p>Time Access to the BUS</p>	 <p>Delay Station</p>
 <p>Devices</p>	 <p>Delay Station</p>

The mixed model that we build, according to the elements described previously is shown in Fig4.5

One thing to notice is that at any moment just one device can use the SCS bus, thus it has to be defined either *Finite Capacity Region* or setting the queue length equal to one. As you can see in the figure above, the queuing is a mixed model where the open model is for the background load and closed model is for the user action load.

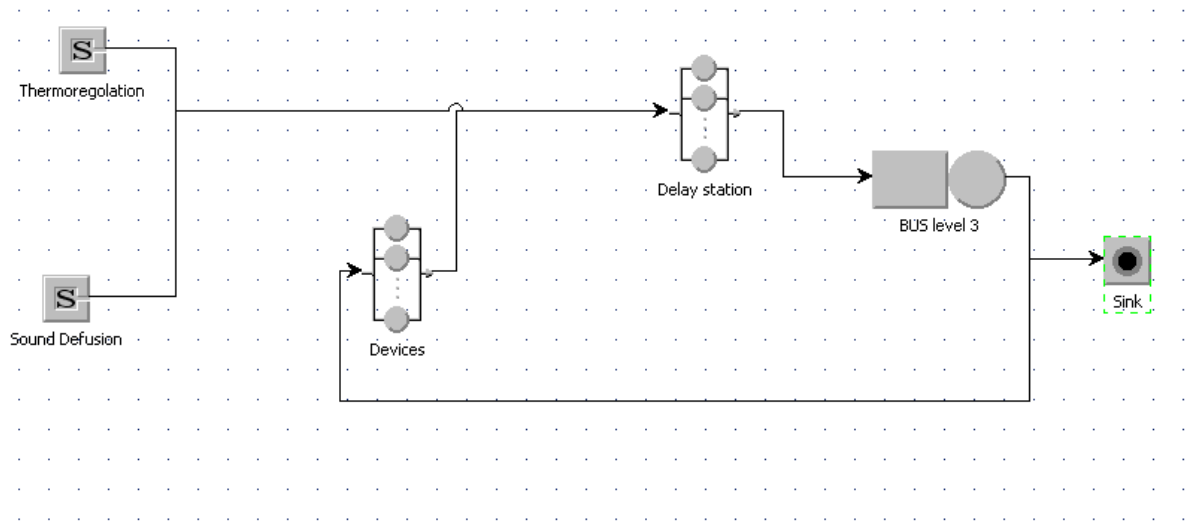


Figure 4.3: Simple QN Model for Level 3

- In the model, the Thermoregulation node presents the source node for BL from the thermoregulation system through the bus, and the same for the Sound Diffusion node.
- The devices node presents a delay station for the devices which make user action load, therefore it is the source node for the user action load.
- The Delay0 node is another delay station to simulate the delay each frame has to have before accessing to the bus and it is exactly before the bus which shows doesn't matter the frame is from user or from background load, in any case it should have a delay time to access to the bus.
- The BUS-Level3 node presents the service center which gives the service to the frames that is a media to communicate and in this project the capacity of this service center is important to be analyzed.
- The "Sink" presents the sink node for the open model.

Here there is a brief explanation about the strategy of the frame transition in the bus. In table 4.3 [5] the details of transmission time and bus occupancy for the three types of the frames is shown where the address of the device is "00":

-	T Frame	T Free BUS	Rx ACK	TOT	Occupancy /sec(percentage)
Point to Point	(7.9msec)	5.2msec	1.66msec	14.8	1.48
Broadcasting	(3 * 7.9msec+ 2 * 3.12msec)	(5.2msec)	0	35.1	3.51
Extended	(3 * 12.5msec+ 2 * 3.12msce)	5.2msec	0	48.9	4.89

Table 4.3: Transmission time and bus occupancy for the three types of the frames

4.4.1 Define the Classes

In queuing Network model, defining correctly the classes of the arrivals is very important. Having good knowledge about the system is the key point to define the good model with well-defined classes which describe the system in a fair way.

During the project, many strategies have been used to define the classes which are the main step to go through. The goal in queuing network is to minimize the number of classes and sum up the loads which are somehow the same as one class.

As it is explained above, there are two main loads come to the Bus. Hence, there are at least two classes which make completely different occupancy in the bus. Then the load for each of them should be clarify to see if for each of them just one class should be define or they should be in separated class. As has been explained in the chapter 2, there are different systems which can be included in the My-Home system that each of them has many commands. There are three types of the frame which are Point to Point, Broadcasting and Extended frame. From the table of commands for each system can be seen each command is a mixture of the number of the point to point, broadcasting or extended frames. there are several ideas to define classes.

The first idea was to define the classes as Commands , but it is not good because first of all the number of classes which should be defined are so many(each command one class) and second the simulator needed long time to provide the results.

Another idea is to group the commands which have more or less the same number of frames in each type of the frames and make a class for each of the group. Although in this way the number of the classes decrease ,compering to the first idea, but the number of the classes which should be define is that much high which we have the same problems as before.

The third idea is to define a class for each system, so the number of the classes is few but how to find a way to evaluate and analysis the workloads and the service time becomes a big problem to deal with, therefore it is difficult to get through this idea too.

The fourth and last idea is to define classes as type of the frame. This has some advantages as the followings:

1. Each command is a mixture of three types of the frame, therefore when a command is sent through the Bus it is easy to calculate the workload and its distribution. In another word we can have three type of $\lambda = \{\lambda_1, \lambda_2, \lambda_3\}$ and for each command it has an absolute number of λ_c .
2. Defining the service time in each station is important especially for the server (the BUS). If the classes have defined as frame types, then easily the *service time* can be define as a constant value because exactly we know how much time each frame occupies the bus , or in another word, how much is the bus utilization when a frame from type i is using the bus.
3. How to define the Background load is another issue. By this definition it is easy to define the background load because as specification of the system and the configuration information for each plant it is possible to calculate how many frames would be transferred through bus.
4. The minimum number of the classes can be achieved by this type of defining the classes.

By the mentioned advantages above, the classes has been defined as frame types. Therefore for the *user action load* the bellow classes are defined:

- **Point to Point**
- **Broadcasting**
- **Extended**

These classes are defined as closed classes; thus, it is possible to calculate the population of the each of them by knowing the commands or using the *WHAT-IF* and analyzing the increment in customer population. This is explained with more details latter.

But there are also the background loads, which should be defined separately. Before mentioning the classes for the background load some points should be highlighted.

1. In the background load there are not any *Point to Point* frame, because the background load (BL) are generated by the system which doesn't use the Point to Point frame in BL.
2. Sound Diffusion just generates the extended (EXT) frame as BL.
3. With the simplification has been done for the Thermoregulation BL, it has been estimated just *Brief Broadcasting* (BB) frame as BL for thermoregulation system.
4. No any BL has considered from alarm system in level 3.
5. The total BL is :

$$BL_{Total} = ThermoBL + SoundBL$$

In general, when there are both thermoregulation system and sound diffusion system two classes should be define as the following:

- **Thermoregulation Broadcasting BL**
- **Sound Diffusion Extended BL**

The BL classes have defined as Open classes with exponential distribution which let them to have the load during the simulation time. The distribution is explained in the next section.

Therefore as queuing network theorem there is a mixed model which is included five classes in total which depending to the configuration of the planet can be less or equal to 5. For example if in a planet there is not Thermoregulation system, the class Thermoregulation Broadcasting BL should be omitted or the load be equaled to zero.

4.4.2 Distribution of the Workloads

The defined distributions in the JMT tool have been explained in the Chapter3, and here it should be defined for each classes how to set a distribution. For arrival rates the distribution should be set just for open classes, because for the closed classes the population should be set.

There are two open classes and their distribution are as the following:

Thermo BL Distribution: by observing the system and analyzing the log files form the system and the theoretical definition of the system , the distribution of the background load (BL) of the thermoregulation

can be defined as *Exponential* distribution which depending to the configuration of the planet, and new λ should be calculated (by the software which has been developed and it is explained in chapter 6), and then run the simulator by new model to see the result. The definition of exponential distribution is[3] :

*"The exponential distribution is used to model **Poisson processes**. The time interval between two consecutive events generated by a Poisson process can be described by an exponential random variable with parameter λ . The parameter λ is a real number, the mean value is given by $\frac{1}{\lambda}$, the variance is given by $\frac{1}{\lambda^2}$.*

$$f(x) = \begin{cases} \lambda e^{-\lambda t} & x \geq 0 \\ 0 & x < 0 \end{cases}$$

By calculating the λ and choosing exponential distribution the arrival load has been defined.

Sound Diffusion BL Distribution : by the explanation in chapter 2 about the sound diffusion and in section 4.2.2 in this chapter, and also observation of the system, there are 3 extended (Ext) frames each second , therefore the BL load of this system has been defined as fixed exponential distribution where $\lambda = 3$.

4.5 Service Time in the BUS

In the chapter 3 some background information related to the queuing network model and D.G.Kendall notations has been mentioned. According to pervious section, the loads from the open classes are exponential and already in the defining the classes is mentioned why the frame types has been chosen to define classes. One of the main advantages which are mentioned is that by choosing the frame type to define classes, it is completely clear that the service time of the classes are Constant (Deterministic) values. Therefore the model is *M/D/1* queuing mode. The number of customers(frames) which can be in the queue are equal to number of devices in the system, because each devices can store a command and in case that the bus is occupied, after another time access to the bus , retries again to access to the bus and sends the frames. Because it is not know which command is stored and how many devices are in the system, therefore the queue is considered as infinite queue capacity.

4.6 How to Simulate the Time Access to the SCS BUS

Time access to the bus is a way to avoid collision in the SCS bus system and it is a special case of the CSMA/CA protocol to avoid collision in the LAN networks. According to the CSMA/CA protocol, each node in the network tries to access the media and first of all checks if the media is free, if it is then it can use it, otherwise for the random number of time it waits and then retries again.

In the SCS bus the same strategy has been used but with some changes. Instead of using a random number to waits, the address (or in case of command devices the ID of the command) is swapped and is multiplied by $0.104ms$ and then summed up by $5.2ms$ to make a kind of random time for waiting to access to the bus. As formula is:

$$\text{BUS Time Access} = 5.2ms + [0.104ms \cdot \text{swap}_{address}(\text{device address or ID})]$$

where $\text{Swap}_{Address}$ is a function which change the position of the Hex numbers, for example F2 becomes 2F.

The minimum time to access to the bus is for the device which has the address "00" and it is 5.2 milliseconds. Mostly the "00" address is for the Touch screen. For the other devices there are not any roles to put the addresses. The lesser the time access to the bus, the higher the priority of the device. The maximum time access to the bus is when the address of the device is "FF", therefore the access time is:

$$\text{Accesstime} = 5.2 + (0.104 * (255)) = 31.72ms$$

Where FF is in Hex and is equal to 255 in decimal. Hence in average the Time access to the Bus is: $31.72+5.2/2= 18.46ms$

In the modeling of the SCS bus the Uniform distribution for the time access to the bus which has been used in the delay center before the BUS and in its service time the following uniform distribution is used:

"The Uniform distribution, referred to as Rectangular distribution due the shape of its density function, describes a random variable that may assume all the values in the range (min,max) with the constant probability $\frac{1}{(max-min)}$. The probability is 0 outside the considered range. The user can either provide the pair (min,max) or the mean: $m = \frac{(max+min)}{2}$ and variance: $c = \frac{(max-min)^2}{12}$ "

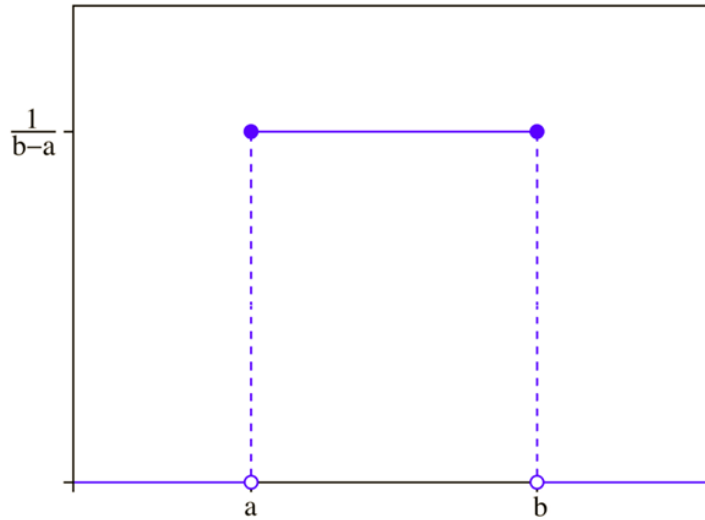


Figure 4.4: Uniform Distribution

4.6.1 The basic components and their parameters

Here we explain each component of the system and the parameters which has been defined for them in more details. these parameters has been defined either according to definitions in the protocols of BTicino or analyzing the real data from the system.

1. **Source of Background Load**
2. **Devices- Source of User Action Load :** This station is reference station of the closed classes and its parameters is defined as the following :

Strategy: *Load Independent*

Service Time Distribution: *Exponential with $\lambda = 1$*

Routing Section: *Random*

3. **Delay Station - Time to Access to the SCS BUS :** The delay time for accessing to the bus is done with putting this station where both loads reach here and then go through bus. The definition here for all the classes (open and closed) are defined the same and its parameters is defined as the following :

Strategy: *Load Independent*

Service Time Distribution: *Uniform with $min = 0.0052$ and $max = 0.03172$*

Routing Section: *Random*

4. **Service station - The SCS BUS:** The SCS bus gives service to the frames generated from the source and Device station. the service time for each type of the frame is calculated, considering the table 4.3. As have been mentioned before, the service time for each type of the frame is deterministic and we know exactly the time a frame occupies the bus, therefore we have defined the Service Time distribution as Constant. All parameters are defined in the table 4.4.

Queue section	<i>Capacity</i> : finite , max number of customer = 1. <i>Queue Policy</i> : FCFS <i>Drop Rule</i> : Waiting Queue (no drop) <i>Station Queue Policy</i> : non-preemptive scheduling(priority)
Service Time Distribution:	<i>Point to point</i> = Constant (0.02994) <i>Broadcasting</i> = Constant (0.02994) <i>Extended</i> = Constant (0.04374) <i>Thermo_{BB}</i> = Constant (0.02994) <i>Sound_{EXT}</i> = Constant (0.04374)
Strategy:	Load Independent
Routing Section:	Probability { <i>Point to point</i> = $P_{Device} = 1$, $P_{Sink} = 0$ <i>Broadcasting</i> = $P_{Device} = 1$, $P_{Sink} = 0$ <i>Extended</i> = $P_{Device} = 1$, $P_{Sink} = 0$ <i>Thermo_{BB}</i> = $P_{Device} = 0$, $P_{Sink} = 1$ <i>Sound_{EXT}</i> = $P_{Device} = 0$, $P_{Sink} = 1$ }

Table 4.4: defined parameters for the SCS bus

4.7 Results and Accuracy of the Model

In this section first we talk about the results we have got from simulation when the WHAT-IF mode when the papulation for closed classes increase

till the bus become completely occupied. Then we assume a typical configuration and without using WHAT-IF we discuss the result. In each part the accuracy has been discussed as well.

To understand the accuracy of the model, it might be difficult to check the accuracy when we have all the classes together, therefore the general model is divided to more significant model which can be easier to check and understand whether is close to reality or not. The model is changed to more simple models when in each of them we have one closed class and all the other background loads (open classes). the reason to doing this is when we define a planet we already can calculate the Background load in the system, because they are generated periodically and they are not dependent to user action. but knowing how much loads come from the user and is added to the utilization from the BL to fully occupy the BUS. for doing this we study each closed model one by one when all other open classes are kept in the model. this could be done without any open classes and just with a closed class but it was more than enough simple, thus we used the former way.

we discuss a model with the Extended class ,when it has still the other open classes. Then the same model but with the Broadcasting closed class instead of extended one.

Note:the below model has been run with What-IF function when the number of customer increases from 1 to 40.

4.7.1 Throughput

In this section we talk about the throughput of the system. According to chapter 2 and basic quantities, throughput of system is defined in as below :

$$X \equiv \frac{C}{T}$$

where C the number of request completions we observed and T is the length of time we observed the system.

Extended

In the light load when the number of customer for Extended frame is equal to 1, the throughput should be 1 as well, because we defined the simulator gathers the samples in each second. This is defined when we have set the service time of the device station as exponential distribution with $\lambda = 1$. Another point is that for open model we defined different load , for example for Sound Diffusion we have defined three Extended frame for RDS in each

second by using exponential distribution with $\lambda = 3$.

By above explanation now we can understand the results from the system. In the figure 4.5 we can see throughput of the model when it is shown the total throughput of the model in (a), throughput of open class in (b) and throughput of the closed model in (c). While there is one customer obviously X is equal to one, but the maximum throughput of the system is 23. to understand this number of throughput is correct or not, we calculated the maximum throughput we can have in a system with Extended frames only. Let's calculate it ,step by step, according to the theory and definition of the system if it is correct :

1. 1 second = 1000 millisecond
2. Each extended frame has service time equal to 43.74 ms.
3. maximum throughput can be obtained is $\frac{1000ms}{43.74} = 22.8ms \simeq 23$
4. Each second there are 3 EXT frames from SOUND Diffusion as background load.
5. The maximum throughput of extended frames from user action (when utilization is maximum)is equal to $23 - 3 = 20ms$

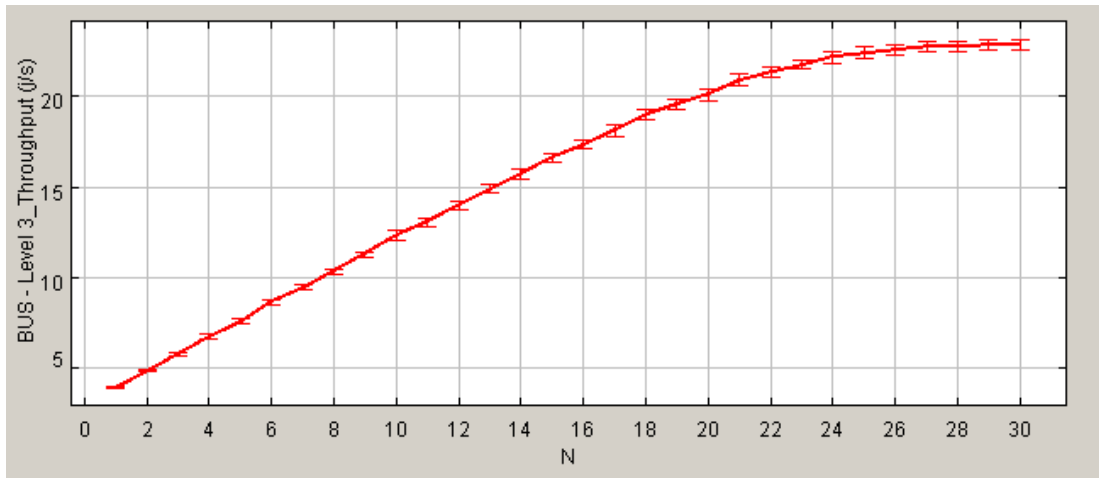
As it can be seen, the results are reasonable for throughput of the system. we have done the same evaluation for the Broadcasting frame in next subsection.

Broadcasting

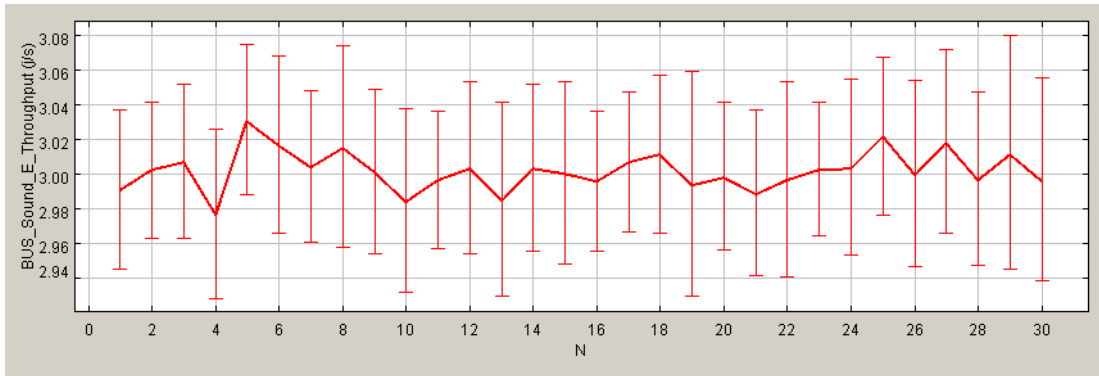
In this section we investigate throughput of the system when there is one closed class, Broadcasting class, and one open class, Sound Diffusion class. As we know the service time of Broadcasting is less than extended frame which we have seen pervious section. therefore there is an increment in number of the throughput. What we have done to evaluate the accuracy of the model is that, first we calculated how much the bus become occupied by Background load, then for each second (1000 ms) we eliminated the occupancy by the BL and the remaining time is divided by service time of a Broadcasting frame to find out the maximum number of frame can be archived from the model.

In the figure 4.6 the result from the model with broadcasting frame is shown. The maximum number of throughput is calculated as below :

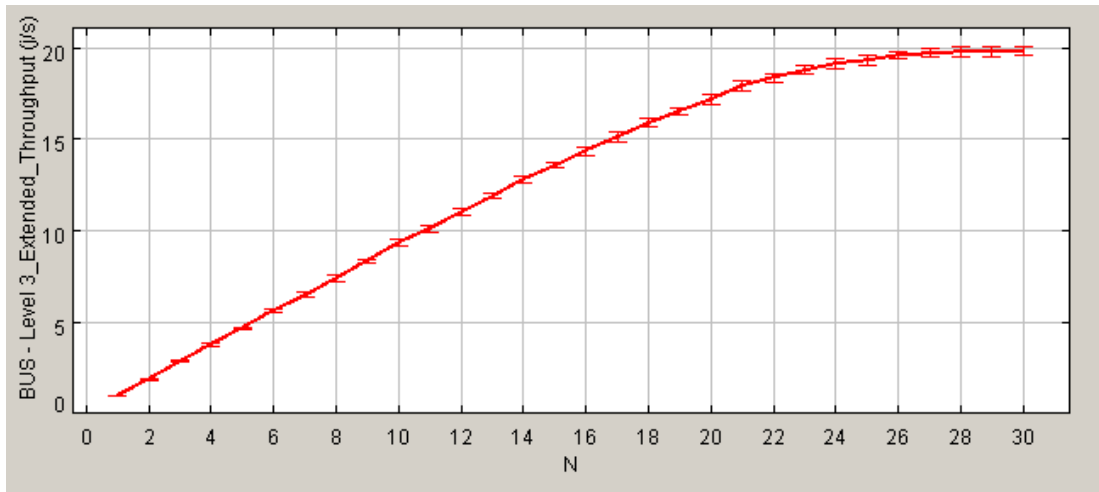
- The utilization of the system from background load is equal to 0.135, therefore we can calculate the maximum throughput with remaining capacity.



(a) Total Throughput of the system



(b) Throughput of the Sound-EXT class



(c) Throughput of the closed class(User Action load)

Figure 4.5: Throughput Result- simplified model to Extended class

- $X = \frac{U}{S} = \frac{1-0.135}{0.02994} = 28.89 \simeq 29$
- The Total throughput is the sum of the BL and UA which is :
 $29 + 3 = 32$,while we can see the same in the simulation results.

4.7.2 Utilization

the Utilization or BUS occupancy in each fraction of time is the sum of utilization from each class on that moment. according to utilization low we have:

$$U \equiv XS$$

$$U_c \equiv X_c S_c$$

$$U = \sum_c U_c$$

For the utilization, here we explain just about the model with extended frame as closed class and because of the similarity in the method of calculation and to not repeat everything again and explaining briefly the models for point to point and Broadcasting are not mentioned.

4.7.3 Response Time

To calculate Response time, we should sum up the response time of each station k, therefore :

$$R \equiv \sum_k R_k \equiv R_{BUS} + R_{Devices} + R_{Delay}$$

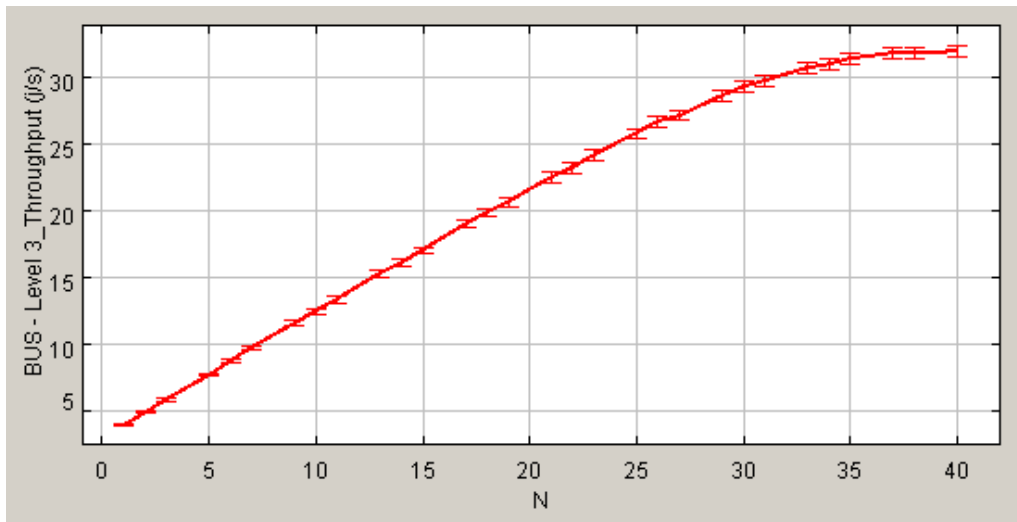
Because we have defined in the model to take the samples each second, therefore the reasonable response time for one broadcasting frame should be summing up of the response time of BUS and response time of Delay station.

$$R \equiv R_{BUS} + R_{Delay}$$

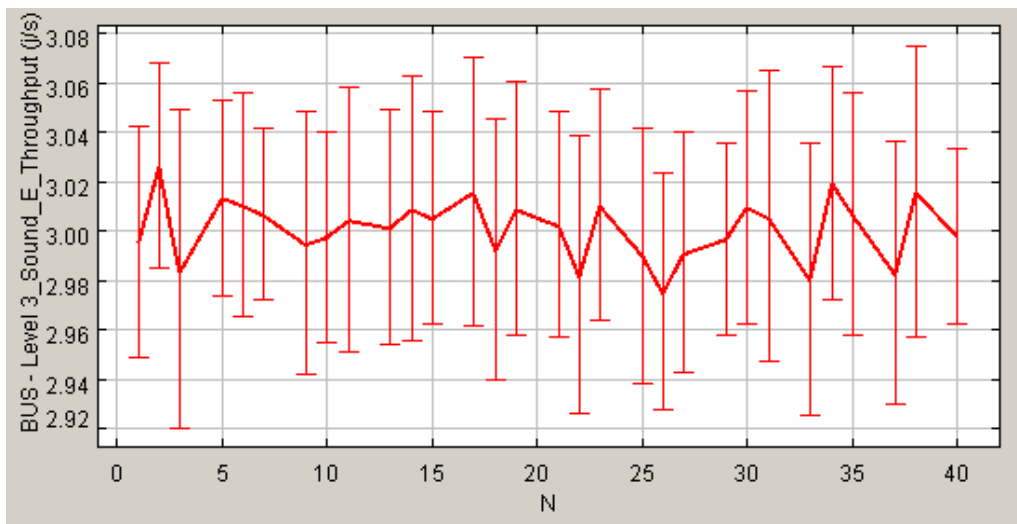
According to [17], the response of queuing center can be calculated from below formula :

$$R_k \equiv \frac{D_k}{1 - \sum_{k=1}^K U_k}$$

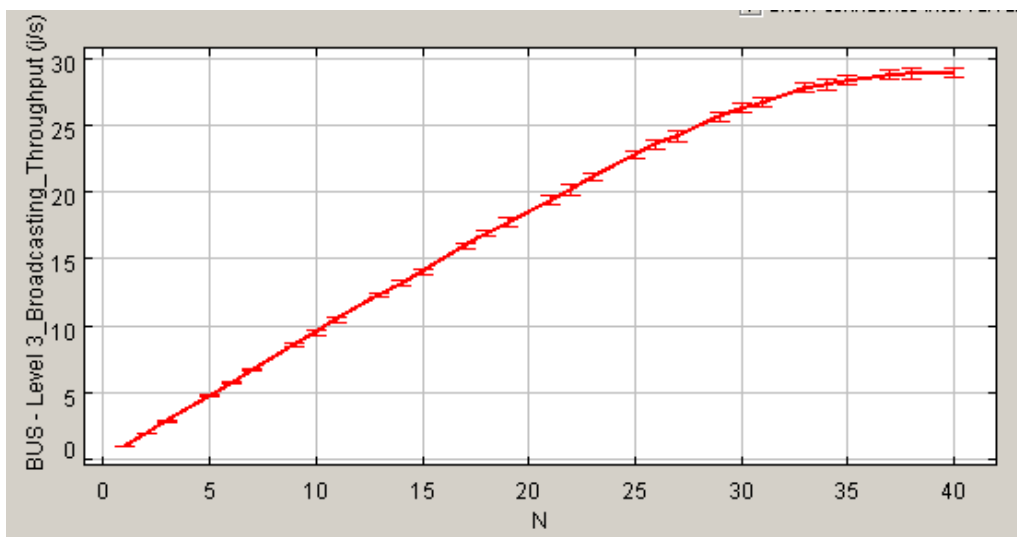
But for the delay center is equal to the service time is set in the delay center. Here we can make an example to calculate the response time of an extended frame in the BUS station and then see the result from the simulator.



(a) Total Throughput of the system

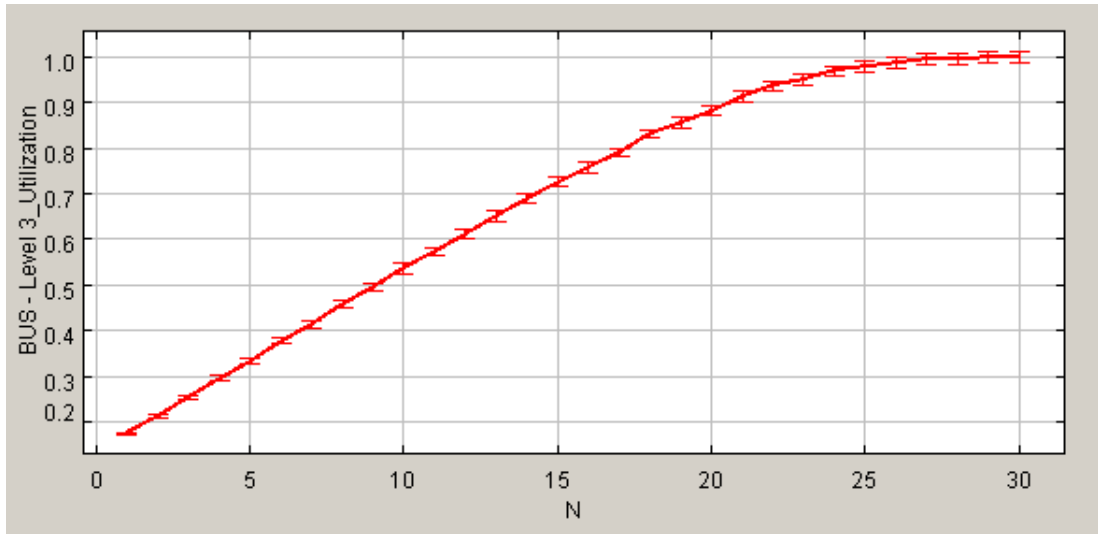


(b) Throughput of the Sound-EXT class

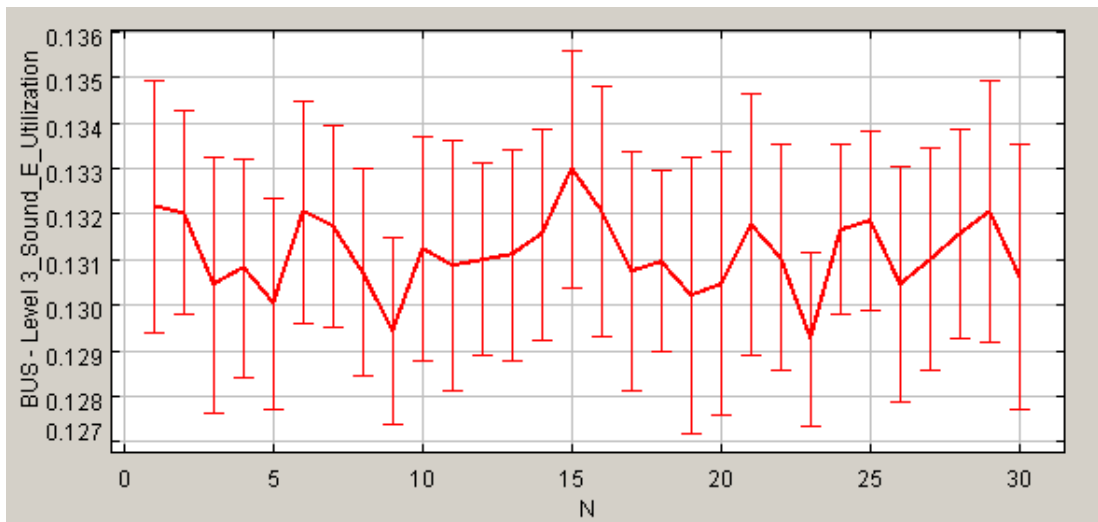


(c) Throughput of the closed class(User Action load)

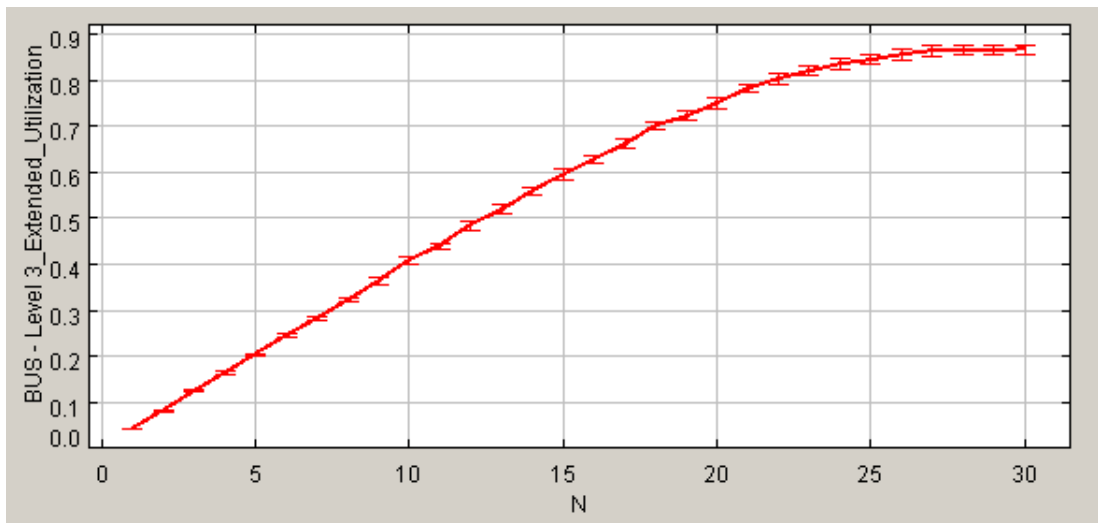
Figure 4.6: Throughput Result- simplified model to Extended class



(a) Total Utilization of the system



(b) Utilization of the Sound-EXT class



(c) Utilization of the closed class(User Action load)

Figure 4.7: Utilization Result - simplified model to Extended class

4.7.4 System Response Time

system response time is the trimming to carry out all requested frames. An example of a system with just extended class and sound diffusion class is shown in figure 4.8.

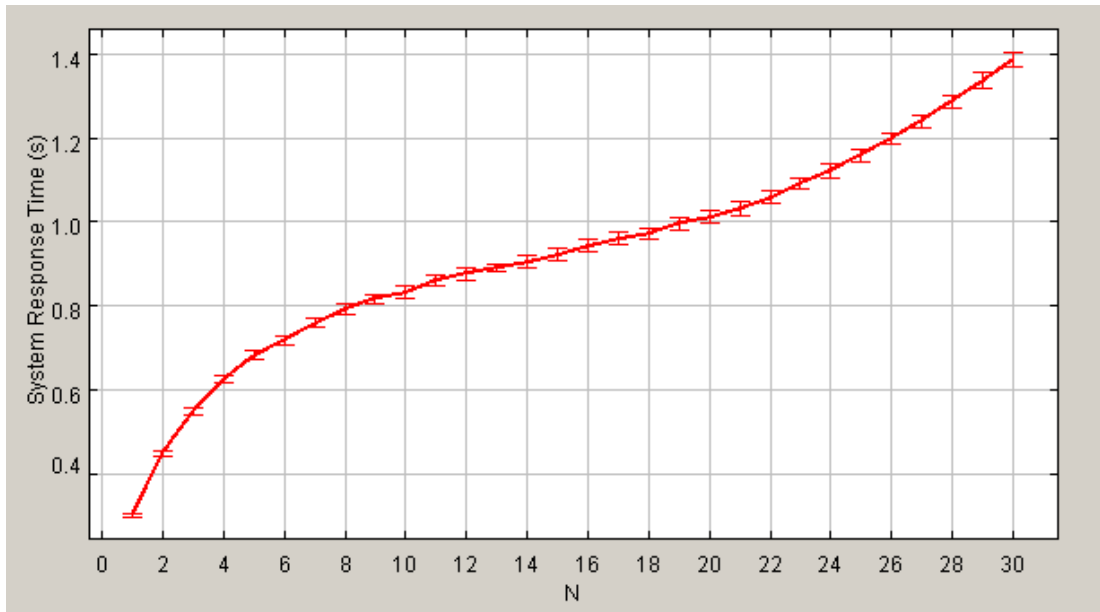


Figure 4.8: System response time - Extended class and Sound class

4.8 Mean Value Analysis - using JMVA

In this section, we discuss about the modeling the system with *MVA* solution (Mean value Analysis) [25]. To do this we used *JMVA* (Java Mean Value Analysis) tool [4] and we compared the results with the result form the *JSIMgraph*. This could help us to decide how to model the system analytically. Thus, if the error between the results of two tools was small we could use *MVA* solution.

According to [3] :

JMVA is the GUI front-end to JMT analytic engine. It allows the solution of open, closed or mixed queuing networks with the exact or approximate MVA algorithm with single or multiple customer classes. Service centers can be of three types:

- *Load Independent*

- *Load Dependent (currently supported only in single-class closed models)*
- *Delay*

JMVA allows the automated execution of What-if analysis i.e. different models are evaluated by changing a defined control parameter. Model definition is guided with a wizard interface. Optionally, created model can be solved with simulator by importing it into JSIM.

To use MVA, the service times of each station (delay stations and service center stations) should be defined as exponential distribution. This means that instead of Uniform distribution for the Delay station and Constant value in the Bus station we should use Exponential distribution.

The same way we used to test the model in JSIMgraph, which we splitted the model to more significant model and compared the result with real data and definition of the system, we do here the same to compare the result of JMVA and JSIMgraph. in the following result discussion, a simplified model with one closed class (*Extended*) and one open class (*Sound_{ext}*) has been considered for evaluation.

4.8.1 Throughput

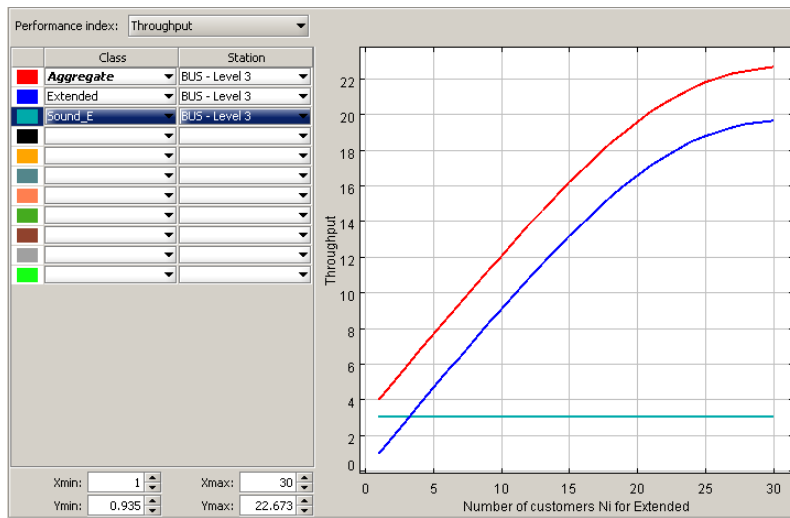
In the figure 4.9 the textual result of the JMVA is shown and later in figure 4.10 comparison between results of JSIMgraph and JMVA are shown.

*	Aggregate	Extended	Sound_E
Aggregate	3.935010	0.935010	3.000000
BUS - Le...	3.935010	0.935010	3.000000
Devices	0.935010	0.935010	0.000000
Delay 0	3.935010	0.935010	3.000000

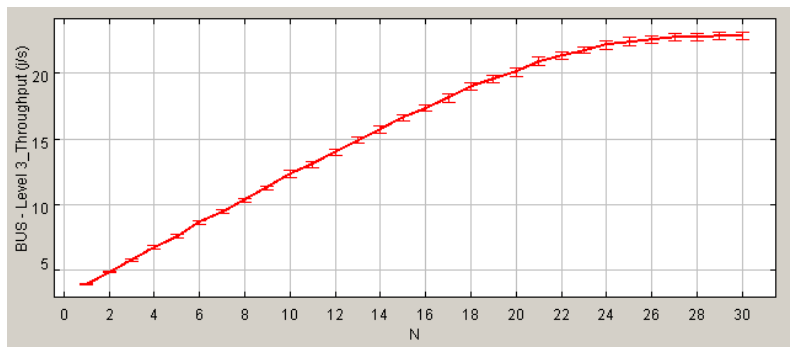
Figure 4.9: JMVA Throughput result - textual

In section 4.7.1 is described how we can measure if the results are closed to reality or not and the accuracy of the model is discussed, therefore here we just make an comparison with the former results and new taken results from JMVA.

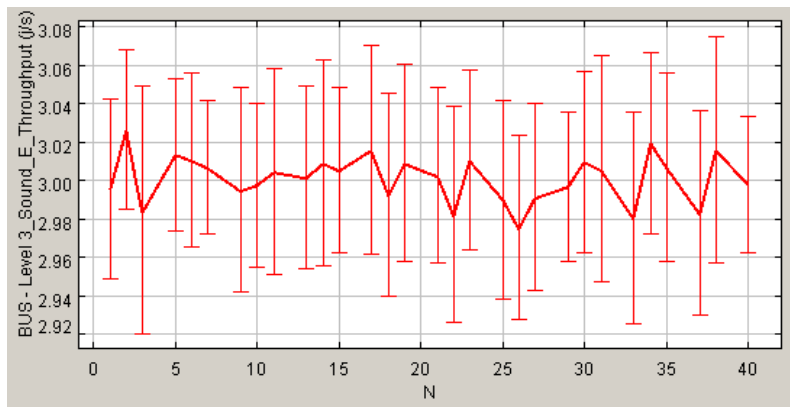
As we can see in figure 4.9 and 4.7, the throughput in both results are the same and an accuracy has been obtained from both model. In the textual result of figure 4.9 , we can see that for Sound diffusion we obtained three extended frames per second which it is what it has been defined as BL of



(a) Throughput result of JMVA



(b) Throughput result of JSIMgraph



(c) Throughput of the Sound-EXT class -JSIMgraph

Figure 4.10: Throughput Results Comparison between JMVA and JSIM - simplified model to Extended class

sound diffusion system, and when we have just one customer (population equals to 1) in total we should have four extended frame. The maximum throughput can be compared in figure 4.10. According to these result the error is very low and the throughput from both solution are almost the same.

4.8.2 Utilization

In section 4.7.2 we have described how the utilization is calculated. According the MVA algorithm for mixed model in [17] (chapter 7), which is explained in chapter 3.

in the figure 4.10 the utilization of the JSIM and JMVA are compared. In the results from JSIM the utilization equals to one when the number of customer is 27, while for the JMVA the number of customer is 31. In both cases the maximum throughput of he system are the same, hence. we can see the same as the throughput the error in results are that much low that we can consider both solutions are equal.

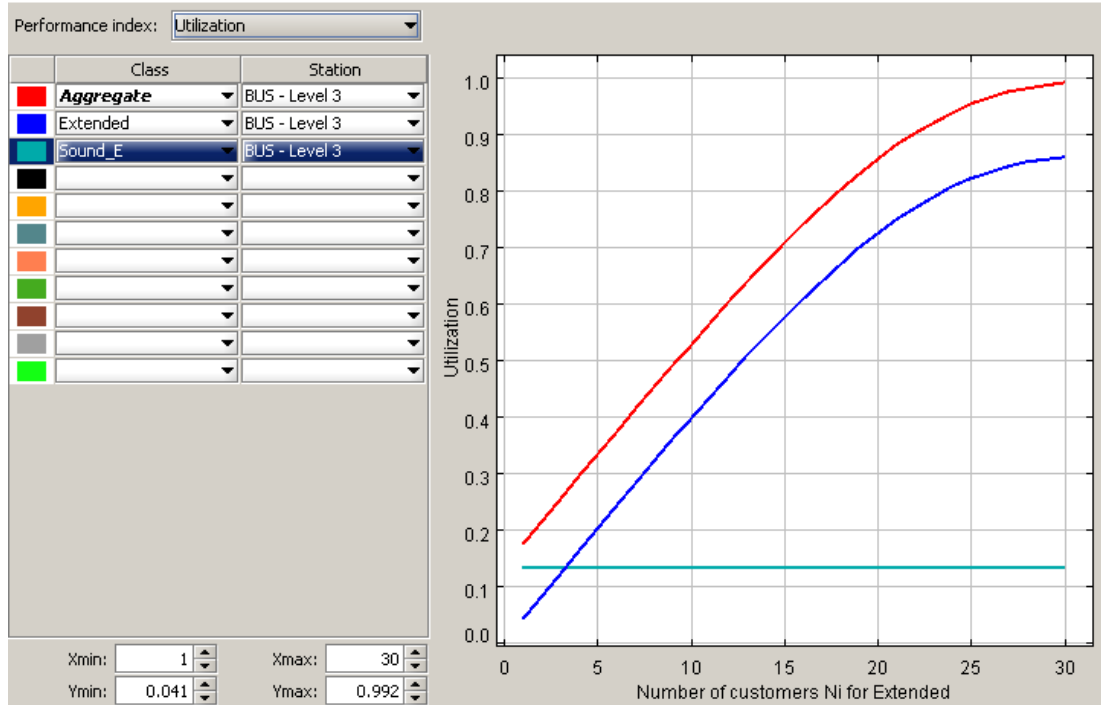
4.8.3 System Response Time

By looking at the result form both tools, it is obvious that the System Response time are completely equal when the number of customer is minimum and when is equals to 30. When th number of customers(frames) are 30, we have the response time of the system is 1.4s. In the figure 4.12 the results are shown.

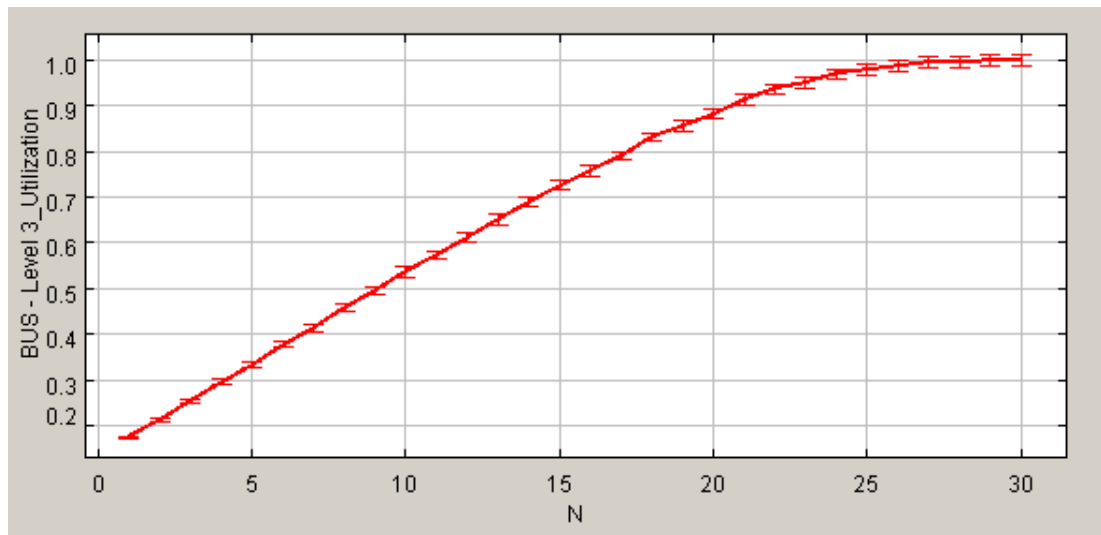
4.8.4 Residence Time

In our assumption, which has been defined in simulation, each frame is a job and number of visit is equal to 1, thus, $S = D$, and from this we can result $Response\ Time = Residence\ Time$. From the figure 4.13 we can see the Residence time of both models are the same.

According to the test we have done, the error in result is very low and we can solve the model with MVA.

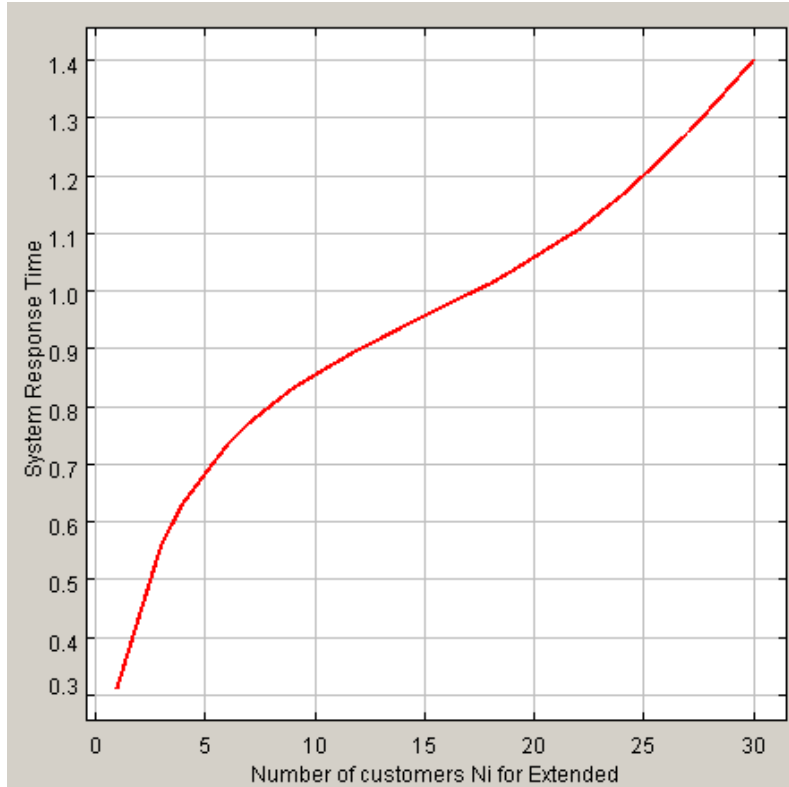


(a) Utilization result of JMVA

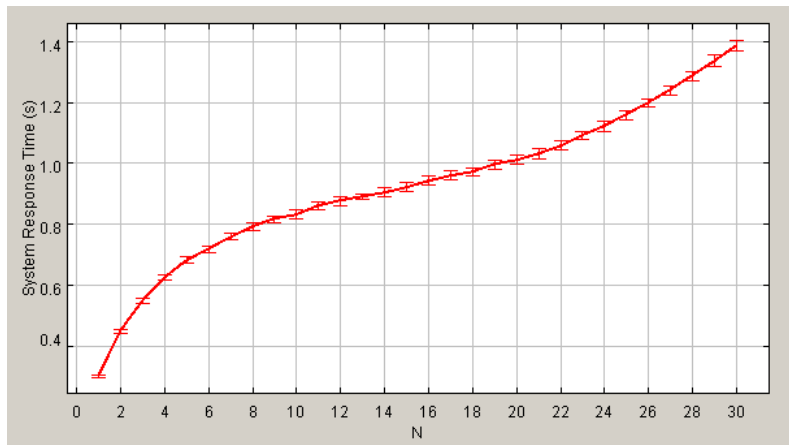


(b) Utilization result of JSIMgraph

Figure 4.11: Utilization Results Comparison between JMVA and JSIM - simplified model to Extended class

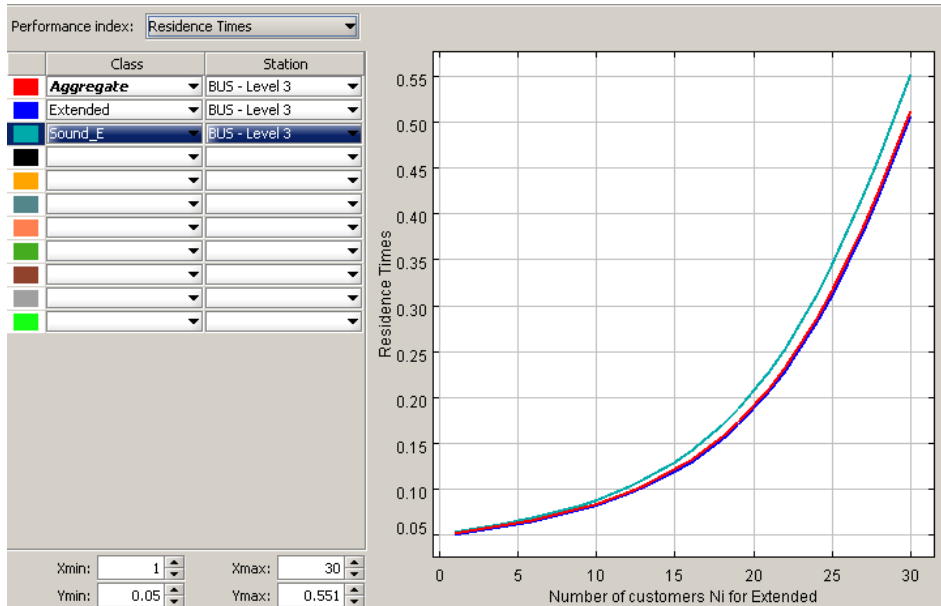


(a) System Response Time result of JMVA

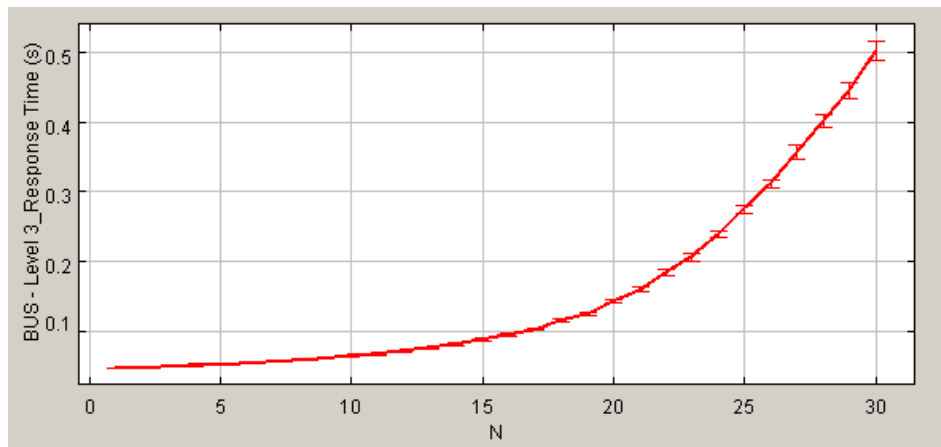


(b) System Response Time result of JSIMgraph

Figure 4.12: System Response Time Results Comparison between JMVA and JSIM - simplified model to Extended class



(a) Residence Time result of JMVA



(b) Residence Time result of JSIMgraph

Figure 4.13: Residence Time Results Comparison between JMVA and JSIM - simplified model to Extended class

4.9 Tree SCS BUS network

Till now we have evaluated a single shared SCS BUS in level 3 where most of the system and devices are exists, and therefore we were more interested to no BUS occupancy and performance indicates in this level of network. But in bigger and more complex installation, the number of SCS BUS increases and this creates a tree shape SCS BUS network. In the figure 4.14 the idea of tree network is shown. In this section we discuss about two type of models, what creates tree network. The first one is when there is one or more level 4 in the palnet which for each of them we have one SCS bus. As it has been explained in chapter 2, there is a device called *Interface* which behaves like a Router in network and it connects two bus together. Where "IL" is the

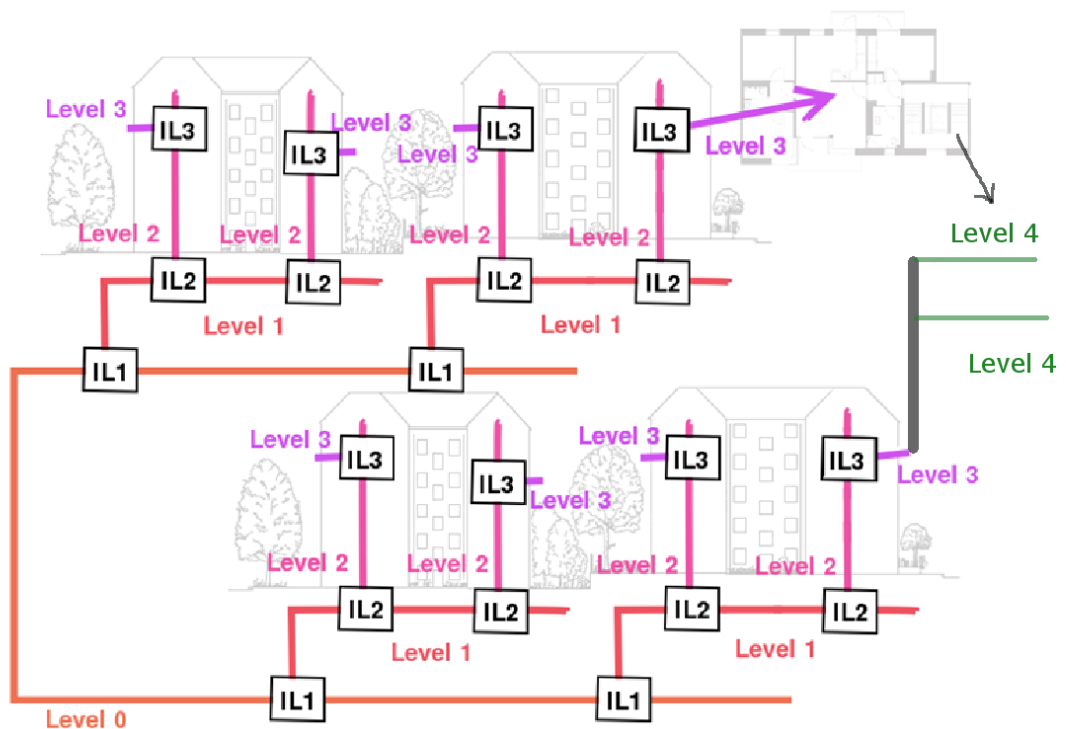


Figure 4.14: Tree SCS BUS Network

Interface between each two BUS.

4.9.1 QN Model for Level 3 and 4

In this section we describe how the model is extended when the level 4 is added to the plant. In this case for example for an apartment, another Bus is added (e.x a room) while it is connected to level 3 SCS BUS. The QN model when level 4 SCS bus is included in model, is shown in figure 4.14. It should be mentioned that all the loads from all the Buses which are in L4 should be summed up and be entered in the model. For example if there are two L4, the population and arrival rate in model is the sum of all the population and arrival rate of the same class of both L4s.

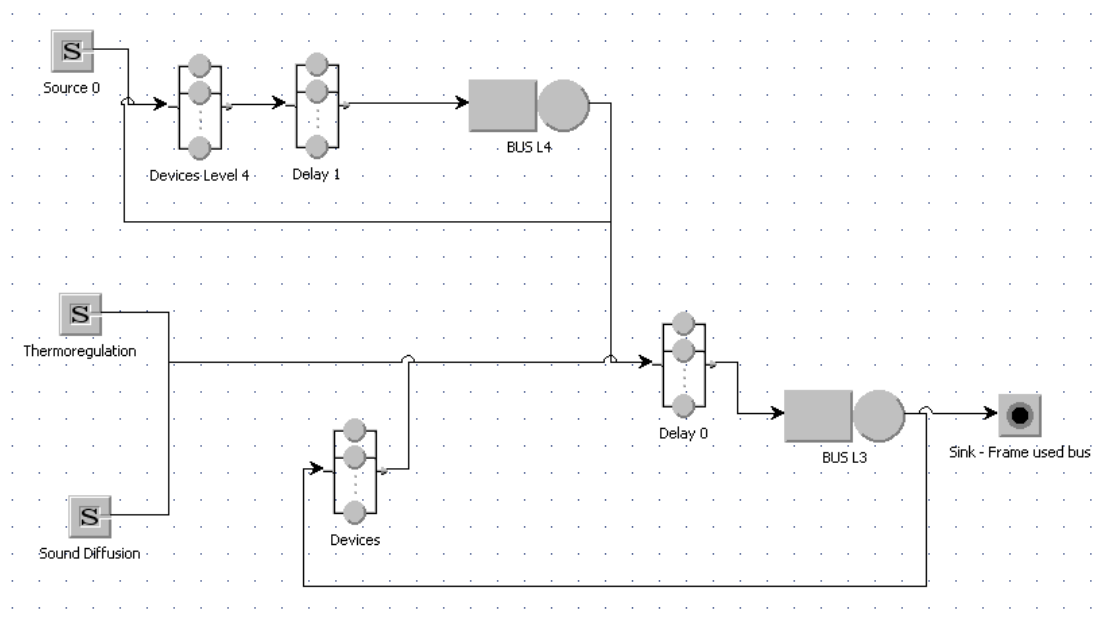


Figure 4.15: QN model for Level 3,4

The structure of the sub model for level 4 is the same with some changes. It has been mentioned in chapter 2 that all the Point to Point frames in level 4, stay in the SCS BUS network of Level 4 and they don't pass through Level 3, while all the other frames pass through level 3.

Another point is that the source of User Action load from Level 3 and 4 should be different because might two different user used the system from different levels of network. Therefore, for level 4. three classes should be defined as User Action Load, while for Point to Point frame it is closed class, because it remains to its BUS, and for the others(BB and EXT) open classes are defined.

In addition, the behaviour of the system regarding to Broadcasting frame of

Automation system is different to others, because the Broadcasting frame from Automation system becomes an Extended when it exists from Level 4 and wants to use the SCS BUS level 3. The reason is that the address of *Interface* of Level 4 is added to the frame. Thus, we define another class for this type of frame as *Automation Broadcasting Frame for level 4*.

In general, there are 9 classes, while the first 5 classes have been defined before (Level 3- UA and BL) and the last 4 classes are added for the Level 4 as the following:

1. **Point To Point:**Closed Class;
2. **Broadcasting:**Closed Class;
3. **Extended:**Closed Class;
4. **Thermo-BB:**Open Class;
5. **Sound-Diffusion-EXT:** Open Class;
6. **PtP-L4:** Closed Class; Point to Point frame for level 4.
7. **BB-L4:** Open Class; Broadcasting frame for level 4.
8. **Auto-BB-L4:** Open Class; Broadcasting frame for level 4.
9. **EXT-L4:** Open Class; Extended frame for level 4.

4.9.2 QN Model for Level 3, 4 and Burglar Alarm

In this section we describe more extended model which includes Burglar Alarm system as well. This model can be simplified to former model, because all the Background Load of Burglar Alarm remain in the SCS BUS of Burglar Alarm and all the commands as User Action load from Burglar Alarm system pass through Level 3 SCS BUS. A question might come to mind that why Burglar Alarm has different Bus if all the UA load comes through level 3 ?; the answer is that because of Power consumption, we need to add another power supply therefore it is needed to add another Bus, and the benefit of this is that the background load doesn't occupy the SCS bus of level 3. According to all explanations of above, we can keep the former Model as a acceptable model of the system and we add the UA load of Burglar Alarm the same as the other systems, supposing the devices are in level 3. This model can be fine if we are interested to evaluate and get the results of Burglar Alarm bus. The extended model is shown in figure 4.15.

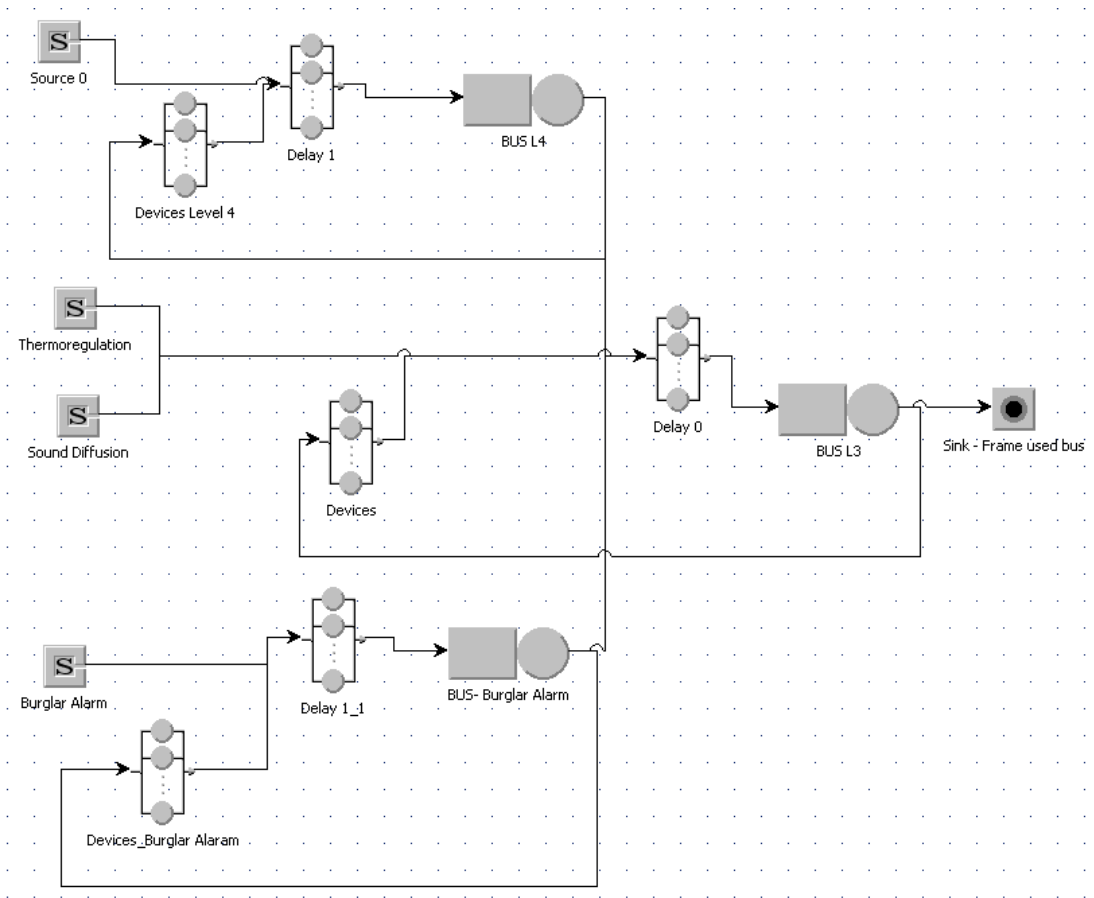


Figure 4.16: QN model for Level 3,4 and Burglar Alarm

4.10 Real Data Analysis

During this thesis, many systems has been tested and has been analyzed. The log files of sample defined systems has been evaluated to underspend the behavior of the system and accuracy of the model and the gathered results from JMT.

In this section the procedure of analyzing data is explained.

Frame meaning According to chapter 3, SCS viewer, the logger of SCS BUS, can log just the time and frames passed through the bus. therefore for any analysis in real data (log file) understanding the frame is very important.

What we did is that we imported the log file in Microsoft Excel and by defining the roles and definitions of the system, each part of frame was separated to understand the meaning of frame such as type of frame, type of system and etc. After this, it is easier to do analysis. Next step is to calculate the Utilization of each frame, which in this case is equal to Service time of each frame depending to its type.

Because of Confidential data, figures of this part is not included in this document.

Inter-Arrival time , Utilization, Frames Count To understand the distribution of load in system, interval-time has to be defined. We consider 1 second as observation time and for each second we calculated the Utilization of system and Number of frames in each second. This has been done by importing the d EXCEL file into Microsoft access and using SQL.

Distribution of date It was necessary to find the distribution of the data. To do that, we should use Goodness of Fit(GOF) of data to understand which distribution is fit to our data [19]. Kolmogorov-Smirnov and Anderson-Darling tests provide Goodness of fit of the date. To find distribution of the date, Matlab code and EasyFit tool have been used. EasyFit tools is a software for Goodness of Fit tests over data [20].

All these steps can be illustrated by an example in the following.

4.10.1 Real data Analysis - Thermoregulation Background Load

Suppose we have a MHS system that has just Thermoregulation system and we are interested in to analyze the Background Load of this system. The

specification of the system is as below:

- Number of Central Units = 4
- Number of Probes = 16
- Number of Actuators = 16
- Number of Pompes = 0.

According to chapter 2 we know that each centerl unit has its own zone and also can control 3 more zones. Therefore we have 16 zones.

Also we know each zones needs at least one Probe(master), therefore in each Zone we have one master probe and one Actuator.

In this specific installation there is one Central Unit which sends "3 BB" frame and the rest of Central units send "2 BB". According to formula 4.1 we have :

$$N_{CU_{Type1}} \cdot 3 + N_{CU_{Type2}} \cdot 2 + 2N_{Act} = 24 + 12 + 32 = 68 \quad \text{BB frames} \quad (4.4)$$

As it has shown in figure 4.17, the number of frames in real data are equal to calculated result from formula.

1	Time	utilization	Count - Frames
2	13:30:39	0.05988	2
3	13:30:40	0.23952	8
4	13:30:41	0.41916	14
5	13:30:42	0.5988	20
6	13:30:43	0.17964	6
7	13:30:44	0.05988	2
8	13:30:45	0.11976	4
9	13:30:47	0.05988	2
10	13:30:49	0.11976	4
11	13:30:50	0.05988	2
12	13:30:51	0.05988	2
13	13:30:52	0.05988	2

SUM Frame :
68

Figure 4.17: Real data - Thermoregulation

The graph of Utilization is shown in figure 4.18. The graph of Utilization for just one period of 15 minutes is shown in figure 4.19. By using EasyFit, the distribution of data according to log file is exponential as it shown in figure 4.20.

According to formula 4.4 there are 68 frames and by using formlula 4.2 for calculating λ we have :

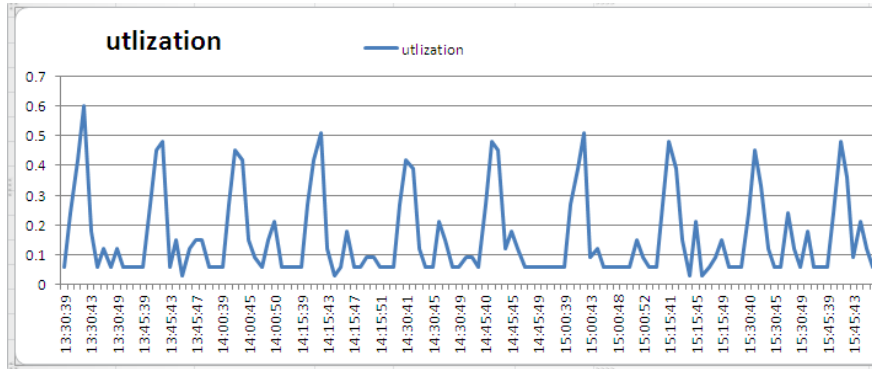


Figure 4.18: Real data - Thermoregulation , Utilization

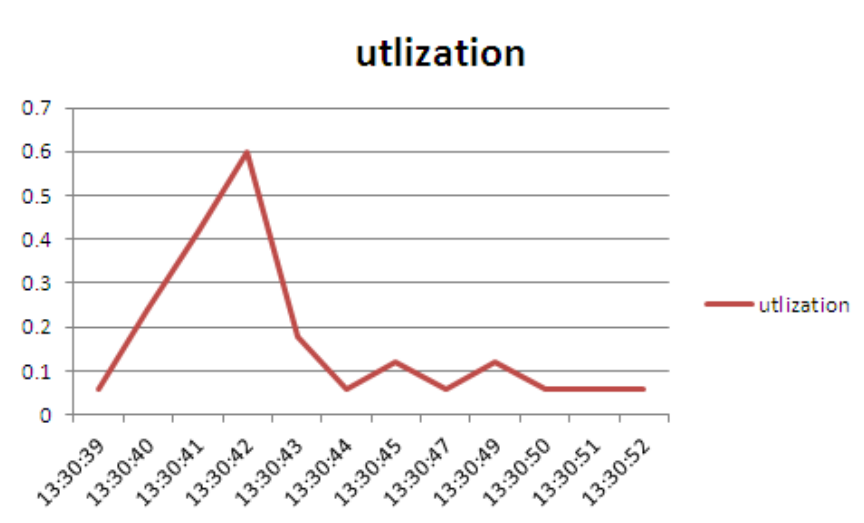


Figure 4.19: Real data - Thermoregulation , Utilization of a period 15 minutes

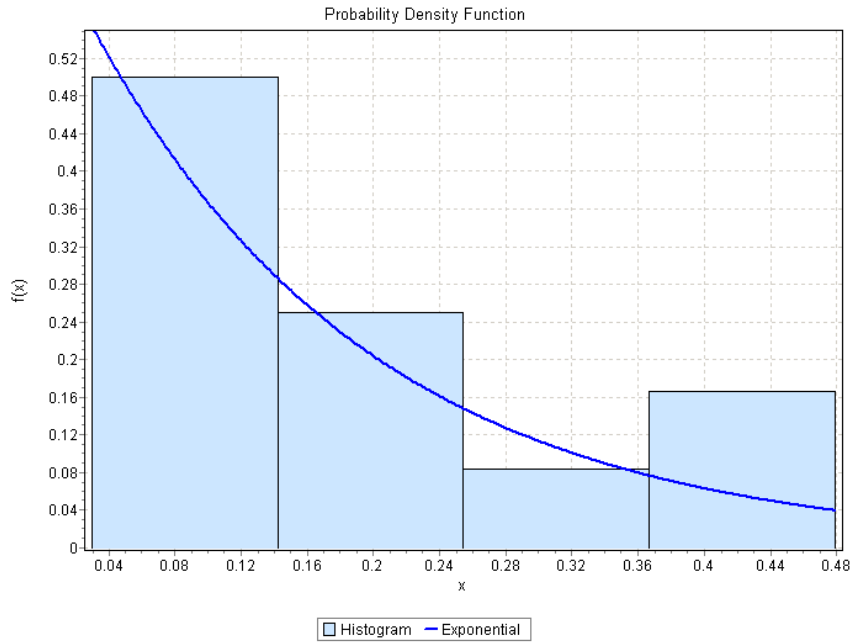


Figure 4.20: Real data - Distribution of data in a period of 15 minutes

$$\lambda = \frac{68}{13} = 5.23$$

This is the λ we define in model for simulation.

The QN model for this example is an open model with one class as BL of Thermoregulation. This sub model of General model which is defined perviously, can help us to do comparison with real data. The model is in figure 4.21 and the results are in figure 4.22.

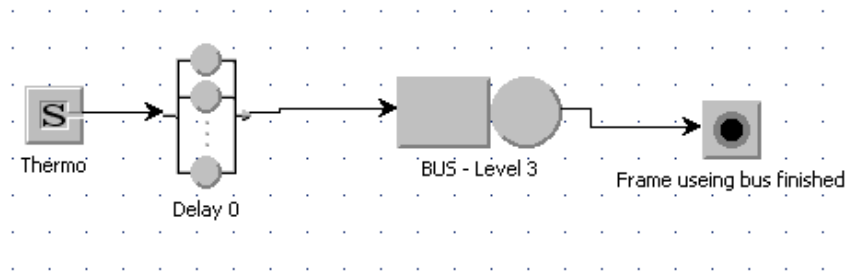
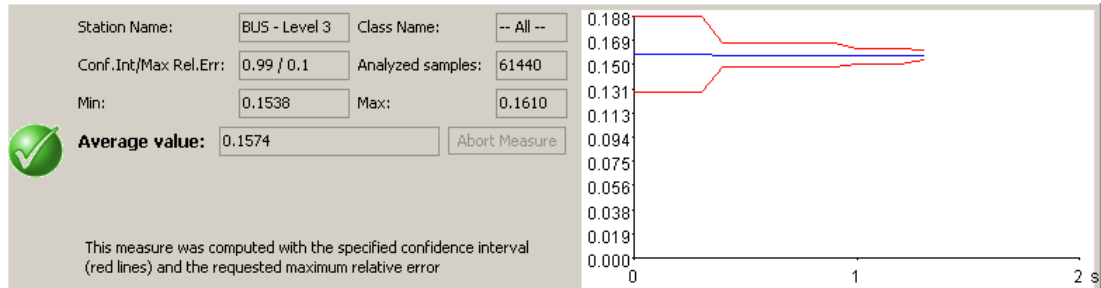
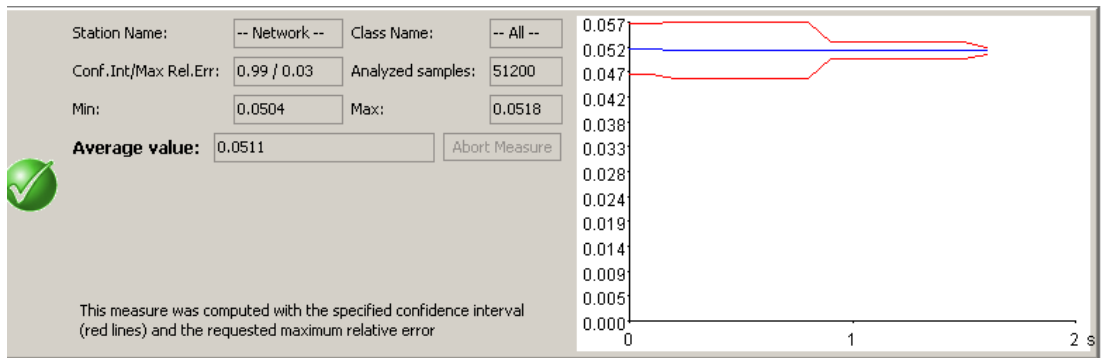


Figure 4.21: QN model -Thermoregulation BL

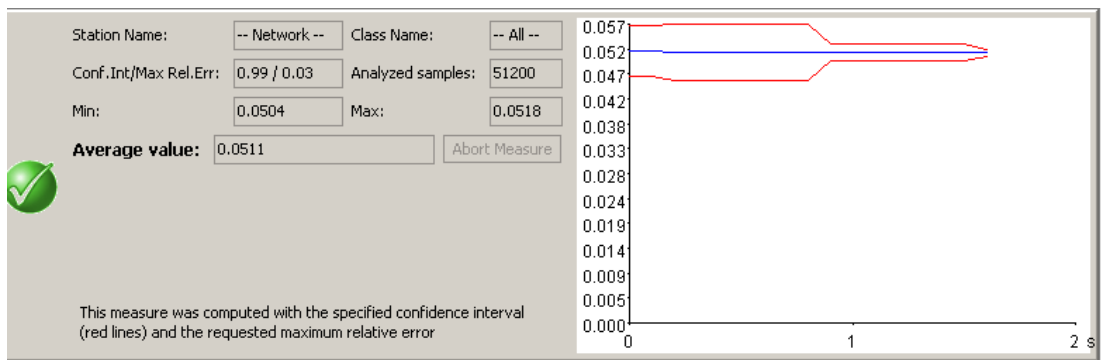
The analytical part is not inserted in this document because we didn't do test and comparison with the simulator and real data, so we avoid to put



(a) JSIMgraph Utilization - Thermoregulation BL



(b) JSIMgraph Response Time - Thermoregulation BL



(c) JSIMgraph Throughput - Thermoregulation BL

Figure 4.22: JSIM Results - Thermoregulation Background Load

it here but the way it should be done is explained briefly in the following and it will be completed in the next step of the BTicino STRATUS project. We made an analytical model, according to MVA algorithm for mixed model in [17]. The basic approach to solve a mixed multiclass model is to solve their submodels independently, but with adjusted service demands to represent the impact on each other[13]. Because the device utilizations do not depend on the closed submodel, the open submodel is solved first and the device utilizations due to the open classes is obtained. The effect of contention at a server due to customers of different submodels is represented by elongating the service demands for the closed submodel. To elongate the service demands appropriately, they are multiplied by a factor that is proportional to the inverse of the idle time remaining from the open submodel [17]. The closed submodel with the elongated service demands is then solved using the MVA algorithm. To solve the closed model, an approximation algorithm can be used which can be iterative or non-iterative [18] [14]. Response times of the open submodels are affected by the closed submodel. To account for this interference, the average queue lengths of closed customers is used in the calculation of response times of open customers (i.e., is included in the number of customers seen by an arriving open customer)[13].

Chapter 5

BTicino QN Model Generator (BMG)

“We are here to change. We are here to grow, develop and unfold. We are progressive beings that have infinite capacity!”

Michael Beckwith

5.1 Introduction

In this Chapter we explain about the software which has been developed which helps to have a model for specific configuration of *MyHome* system. As has been mentioned in previous chapter, in *My Home* system there are different types of the control systems which the owner by considering of his needs can have them. By defining the needs, the configuration of the system can change and therefore the occupancy of the SCS bus will change as well. We will have similar model to basic model with changing to some parameters of model such as Population, arrival rate, number of classes and etc. to be clear we make an example. suppose a customer needs just Automation system for lighting without having other systems. in this case there is no background load and all the load is generated by *user action*. Hence the model should be changed. even in the mentioned example, with having different number of activators in different room or groups, the loads can change and new model should be simulated.

5.2 JMT XML file

According to [2] the QN models of the JMT are saved in XML format. the *JSIMG* file is XML file of the *JSIMgraph* and *JSIMwiz*. In the figure 6.1 the JMT Architecture is shown.

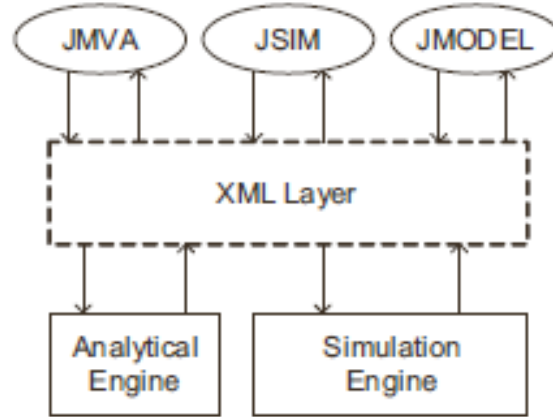


Figure 5.1: JMT Architecture

The XML File parameter is a well formed XML File according to *JMT-model.xsd* schema described in the section 2.2 and figure 2.2 of JMT System manual [2].

Therefore by knowing and using the XML Architecture and having the configuration of the system, we can find the the parameters of the model which should be change, add or delete from the XML file and provide new model and the idea behind the application is exactly this.

5.3 BMG application

the motivation to develop this application has been explained in pervious section. BMG is developed in JAVA, the reason is that JMT is an open source application written in Java and to use the *Simulation engine* the easiest way was using Java.

The flowchart of the application is in the figure 6.2.

In the figure 6.3 the interface of BMG is shown.

there are three steps to generate new model with BMG :

1. Selecting the systems in the MyHome system
2. Getting Configuration

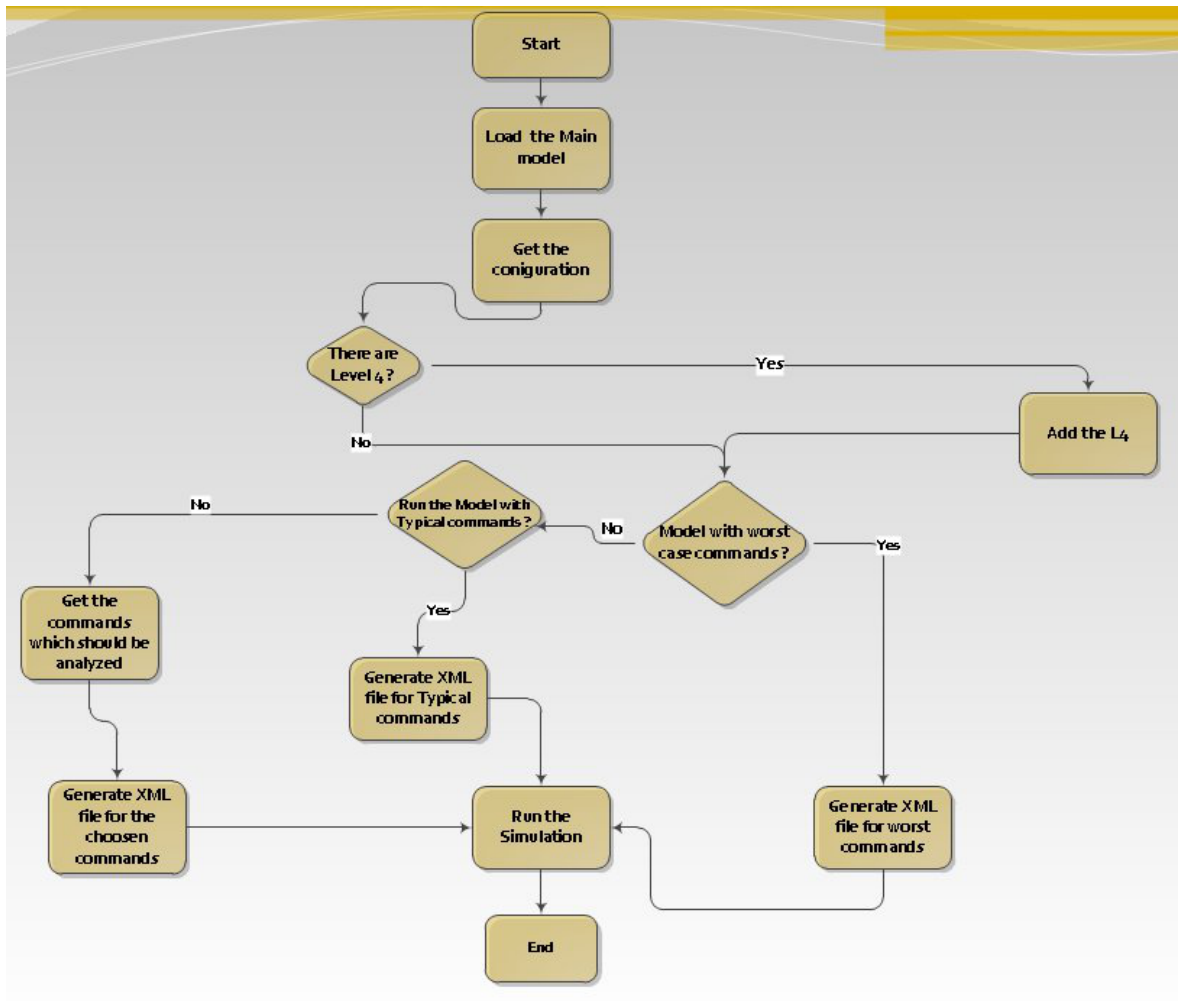


Figure 5.2: Flowchart of BMG

3. choosing the type of commands (*User Action*) for simulation

the steps are explained in the following.

5.3.1 Selecting the systems in the MyHome system

In this part, system analyst(or Installer) selects the systems of the MyHome to be configured in the next step. The figure 6.3 shows the first step.

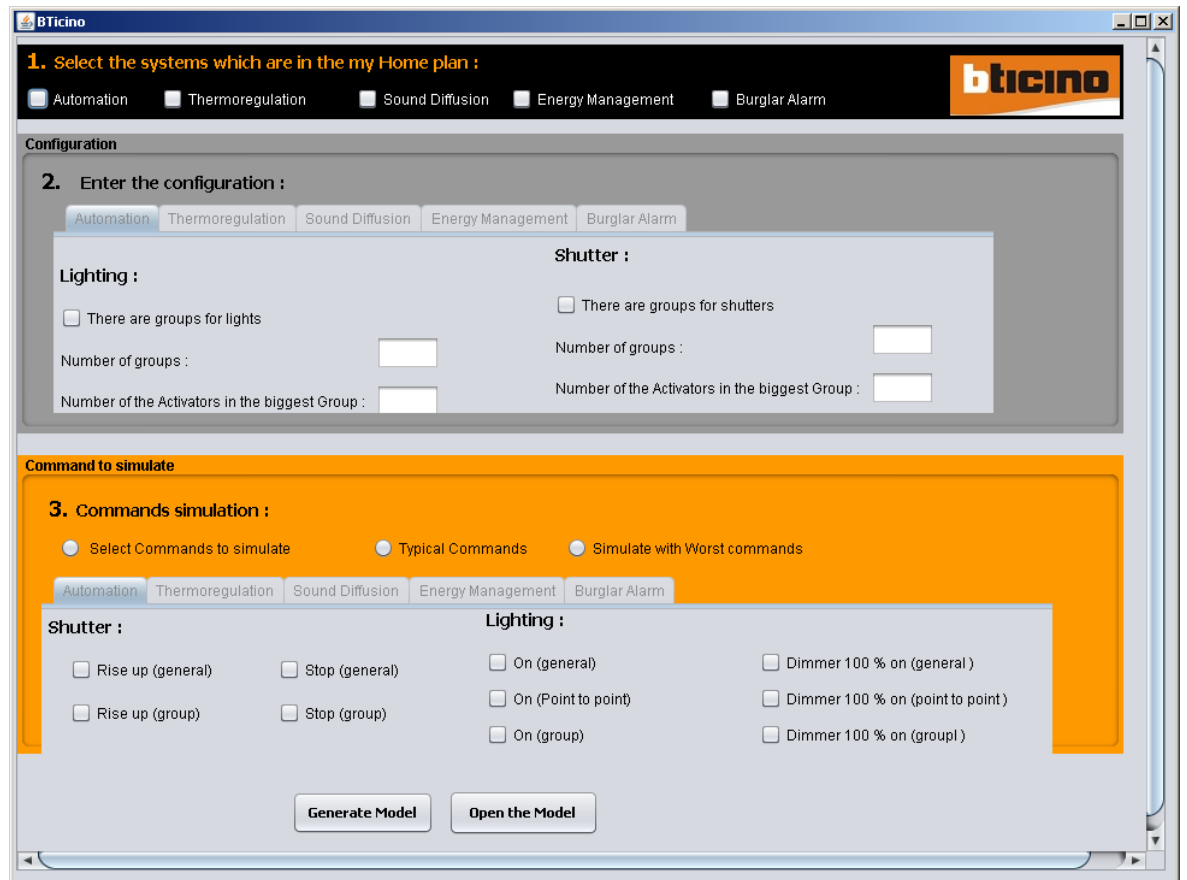


Figure 5.3: BMG Interface

5.3.2 Getting Configuration

In this part, for each selected system in pervious step, the configuration should be defined. The requested information is those which can effect on the model. For example, for Automation system, to simulate the worst case, we need the number of activators in the biggest zone, which makes the biggest load among group commands. The figure 6.4 shows the second step.

For *Energy Management* and *Burglar Alarm* system no information related to their configuration is asked because in their configurations don't change the model.

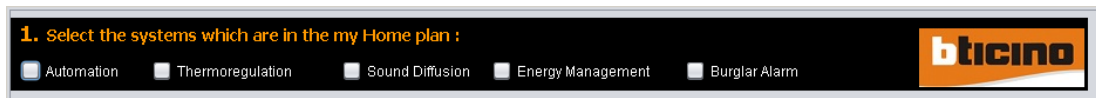


Figure 5.4: First step: Selecting the systems in the MyHome system

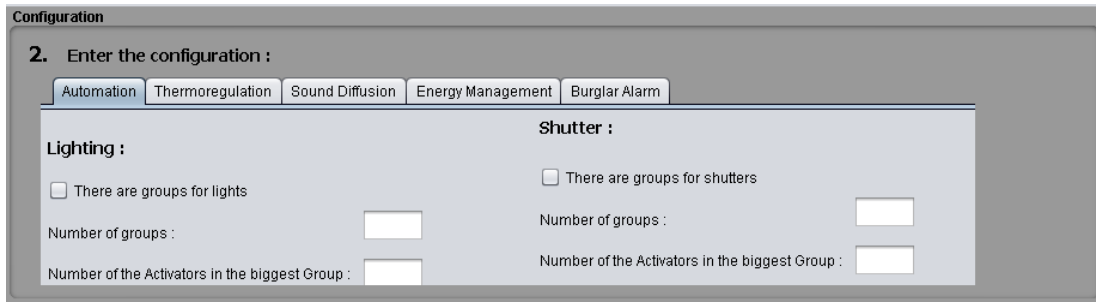


Figure 5.5: Second step: Getting Configuration

5.3.3 Choosing the type of commands

Simulating the user action is very important and define a general distribution for user behavior is difficult job to do, hence to simulate user behavior, in this part we ask to do some test in case there are some users and they push some commands simultaneously and generate a load.

The figure 6.5 shows the third step.

Depending the several options to choose some commands and push simultaneously, we can have three type of loads :

- Worst commands
- Typical commands
- Desired commands

Worst Commands are those commands in each system which send many frames especially Extended that is longer than other types of frame. For example in *Automation system* the *Group* commands are the worst commands(if groups exist).

Typical Commands are the most probable commands user could push, they might even not generate many frames but in each system there are some commands that are used more than the others. For example , in *Lighting system* ON and OFF commands are most used commands.

If we want to choose commands we want, we should select them as *Desired*

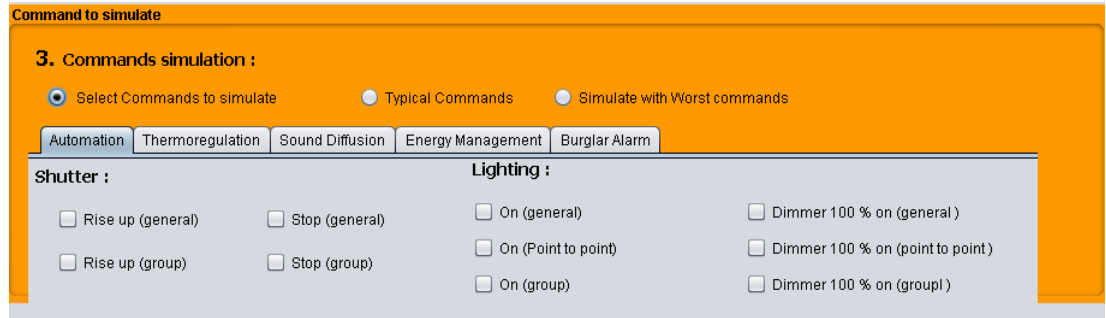


Figure 5.6: Third step:choosing the type of commands

Commands which we want to simulate from the listed commands of each system.

In any case, after choosing the type of User Action Load, the formula to calculate the population of the closed classes is as below :

$$Population_C = \sum_i Command_i \text{ number of frames Type C} \quad (5.1)$$

where C is the closed classes and i is the commands of user action.

In this application background load of the thermoregulation is calculated by using formula 4.1 and BL of Sound Diffusion is calculated using formula 4.2.

Finally by pushing the *Generate Model* button the new model is generated and can be saved and then it will be automatically open in the *JSIMwis* where user can start the simulation and see the result. If a saved file is needed to be simulated again *Open Model* button should be pushed to select file.

To open a saved model and used the simulator again , *Open Model* button can be used.

Chapter 6

Conclusion and future works

“Intellectual growth should commence at birth and cease only at death.”

Albert Einstein

In this thesis, a generic Queuing network models that can be used to simulate BTicino SCS BUS network has been presented. The QN models has been simulated by Java Modeling Tool(JMT) and the results has been compared by real data and definitions of the My Home system. The whole model has been divided to simple sub-models to find out the accuracy of the models and results of simulation. Generally, the results from the presented models are very close to the real data and the models have reasonable accuracy.

As a proof-of-concept, for thermoregulation system which has complex mechanism, the background load has been tested for its effect on the network. However, different plants have different configurations, therefore the model should be defined for each specific installation when all the information of system such as number of devices, number of actuators and etc are provided. For this reason, depending to the information of the plant, the parameters of the model have to be changed to simulate that specific system. To reach this goal, a framework has been developed to generate models according to the specification of the plant as input for JMT.

Furthermore, the tree networks of SCS bus when there are level 4 and the network of Burglar alarm has been modeled by extending the main model what was defined for Level 3. Not limited to these investigations, the models are analyzed by MVA techniques and JMVA tool and the results compared with former results from JSIM. This could help us to decide how to write Analytical model of the system. We found out that error between both tools are very small and we can use Approximate MVA algorithms.

6.1 Future works

The future work considers the following aspects:

Runtime simulator: A simulator for BTicino network can be developed which can work also runtime. The main point of this aspect is that user behavior can be simulated easier and the behavior of the system depending to current action can be seen at the moment.

Scenario validator: User can define several Scenario in the system. A validator can analyze the defined actions and validate it to understand if it is acceptable or not.

Recommendation system: This system can help installer to decide how to install the system in MHS, and do some suggestions such as separating the systems and etc.

Command Logger: The features of SCS viewer can be improved by a new logger which can log utilization, Command of user and etc. This can help to understand user behavior easier and also to find out faults of the system.

Research to other domotic systems: studying other domotic system such as KNX to illustrate the differences and compare their key performance indicators.

Bibliography

- [1] Bticino internal document, "technical specification of the scs bus bti-
cino". 1999.
- [2] *Java Modelling Tools - System manual, v.0.1, 9 pp, June, 2006.* 2006.
- [3] *Java Modelling Tools- Users manual, v.0.8.0, 173 pp, Jun, 2010.* 2010.
- [4] Marco Bertoli, Giuliano Casale, and Giuseppe Serazzi. Jmt: performance engineering tools for system modeling. *SIGMETRICS Perform. Eval. Rev.*, 36(4):10–15, 2009.
- [5] BTicino S.p.A. *BTicino Internal Document, "Limits of My Home System"*.
- [6] BTicino S.p.A. *BTicino Internal Document, "SCS-Collision "*.
- [7] BTicino S.p.A. *BTicino Internal Document, "SCS HANDBOOK-Automation System"*.
- [8] BTicino S.p.A. *BTicino Internal Document, "SCS HANDBOOK-Burglar Alarm System"*.
- [9] BTicino S.p.A. *BTicino Internal Document, "SCS HANDBOOK-Energy Management System"*.
- [10] BTicino S.p.A. *BTicino Internal Document, "SCS HANDBOOK-Sound Diffusion System"*.
- [11] BTicino S.p.A. *BTicino Internal Document, "SCS HANDBOOK-Thermoregulation System"*.
- [12] BTicino S.p.A. *"BTicino TECHNICAL GUIDE" , version 08.*
- [13] Lawrence W. Dowdy By Daniel A. Menascé, Virgilio A.F. Almeida. *Performance by Design - Computer Capacity Planning by Example.* Pearson Education, Inc., 2004.

- [14] Chandrakanth Chereddi. Mean value analysis for closed, separable, multi class queueing networks with single server and delay queues.
- [15] R. Muntz. R. Palacios E Baskett, K.M. Chandy. Open, closed and mixed networks of queues with different classes of customers. April.
- [16] G. Pujolle E. Glenbe and J. C. C. Nelson. *Introduction to Queueing Networks*. John Wiley and Sons, 1987.
- [17] John Zahorjan G. Scott Graham Kenneth C. Sevcik Edward D. Lazowska. *Quantitative System Performance : Computer System Analysis Using Queueing Network Models*. Prentice-Hall, Inc., 1984.
- [18] C. T. Hsieh and S. S. Lam. Pam - a non-iterative approximate solution method for closed multichain queueing networks. May 1986.
- [19] <http://www.itl.nist.gov/div898/handbook/>. *Engineering Statistics Handbook*.
- [20] <http://www.mathwave.com/products/easyfit.html>. *EasyFit website*.
- [21] jmt.sourceforge.net/. *JMT website*.
- [22] Simon S. Lam. A study of the csma protocol in local networks.
- [23] Giuseppe Serazzi Marco Bertoli, Giuliano Casale. User-friendly approach to capacity planning studies with java modelling tools.
- [24] A. Sterrett. Mathematical techniques for the performance oriented design (pod) tool. August 1990.
- [25] Hai Wang. Approximate mva algorithms for solving queueing network models. Master's thesis, University of Toronto, 1997.