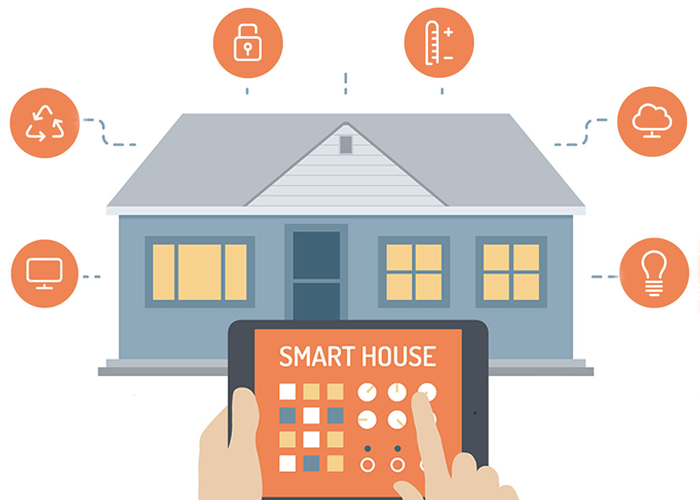
Home Domotica Project

Design Information



Wim Jacops

TAble of  
Contents

Home Domotica Project Introduction 1

Existing System 2

Bill Of Material 2

Concept Diagram 3

Design Critera 3

Hardware specification 3

Software specifications 5

Non-functional requirements 6

# Home Domotica Project Introduction

The purpose of this document is to described the system design information for a Home Domotica Project. The design covers both the hardware and software parts of the new system

A self-build home domotica system was implemented when the house was build back in 1999. This system is used to control lights and power outlets in different rooms of the house.

The goal of this project is to replace an existing system by a new one. Although the system has been running for more than 16 years without any problems, the main reasons that this system needs replacement are:

* Expansion capabilities limited, mainly due to limited memory in the central controller.
* Software that controls the existing system is a self-developed C program. Updates to this C program need to be done using an old laptop with a serial port connection. Although this laptop (still running on Windows 98) is operational, the risk of this device failing is high. Should this happen, there are no available alternatives for maintaining the system.
* No working spare device is available. In case of a hardware failure, light and power outlets connected to this system would no longer be available.

## Existing System

The existing design is implemented using a star-based wiring concept, where all wires that connect light switches end-up at a single location in the house.

A central control unit, consisting of one Z-World micro-controller and 6 expansion boards (3 boards with 32 input and 3 boards with 32 output connections), is installed at the location where all wires are centrally coming together. The central control unit runs on 24Vdc, which also powers the expansion boards.

Since the output boards run on a 24Vdc current, they can’t be directly connected to the lights or power outlets that are powered by 220 Vac.

As a solution for this, the light and power outlets are directly controlled via relays. The relays are controlled by/connected to the digital outputs on the output expansion board.

The input boards operate at 5Vdc and are directly connected to the light switches. This 5Vdc is provided by the input boards. Pressing a light switch pulls the input to the ground.

The existing system is programmed in C. This allowed it to be very fast and compact, the entire program is for example able to run in less than 128k of memory. The drawback of using this program language is that it requires a specific compiler, which is only available via a dedicated (payable) software package. The program is specifically written for the Z-World controller and is not portable to other devices.

### Bill Of Material

Central Control Unit:

* 1 x Z-World PK2200 Micro Controller
* 3 x XP8110 expansion board => 32 digital inputs
* 3 x XP8120 expansion board => 32 digital outputs

Relays:

* 50 x Releco C7-A10/24Vdc
* 20 x Vynckier Pulsar 24V – 25A/250Vac

Note: not all relays are actually used, some have been installed as spares

Transformer:

* 1 x Siemens 4AV2400-2AB

Compiler:

* Dynamic C

### Concept Diagram

<TO BE INCLUDED>

## Design Critera

### Hardware specification

Although the existing system is getting older and needs replacement, not all components should (or even can) be replaced/changed.

The most obvious part is the cabling structure. No new wiring is planned to be installed and the existing cables should be reused as much as possible.

The same applies to the relays. They are still functioning and are expected to keep operational for quite some time. The Releco relays have a mechanical life time of more than 10 million switching cycles. Replacing them would not be necessary, as they are expected to last many more years, and create unnecessary costs.

It would also allow to keep the transformer and no investment is needed to replace this component.

Keeping the 24Vdc relays has as consequence that the hardware of the new solution also needs to operate on 24Vdc.

The input boards should use the same mechanism as in the current system. This means that, pressing a light switch will pull the digital input to the ground. The operating power can be freely chosen, but the system should be able to provide the required operating current (preferably also at 5Vdc as is the case in the existing system)

Accessing the system using a serial connection should be avoided as this type of connection is no longer standard available on computers/laptops. Access via an existing (wireless) network in the house is preferred. Alternative methods for connecting could be USB or Bluetooth.

The new system should comply with following hardware specification:

|  |  |  |
| --- | --- | --- |
|  | Value | Justification |
| Operating power | 24Vdc or 220Vac | The available electrical power supplies are 24Vdc provided by the transformer or 220Vdc from the electrical power grid. |
| Inputs | Minimum 96 digital inputs, operating at 0-5Vdc | There are 96 light switches installed in the house |
| Outputs | Minimum 70 digital outputs, operating at 0-24Vdc | There are in total 70 relays installed in the house |
| Network | Wireless and/or wired ethernet | Access to the system for programing and/or maintenance activities. |

### Software specifications

Although Home Domotica systems are available on the market the goal is to develop my own software. The programming language is not necessarily fixed to known languages (like C or C++), but the learning curve for a new language should be limited.

Investments in programming software should be avoided, so opensource or free compilers/programing languages are mandatory.

The new system should comply with following software requirements:

|  |  |  |
| --- | --- | --- |
|  | Requirement | Justification |
| Portability | Re-usability of software developed for the home domotica platform. | Hardware systems evolve fast and the same systems might not be available after a few years. The software should be portable to new (versions of the ) hardware should this be required, with limited effort and/or recoding |
| Programming software | Free/Open source development environment | Avoid investments in Software packages |
| Program language | Known or easy to learn mainstream programing language | Avoid steep learning curve to start development activities.  Program language should be widely used by the programming community. This will allow libraries and documentation to be available. |

### Non-functional requirements

Next to the Hardware and Software requirements, the new systems should also be fit-for-use and fit-for-purpose. In order to achieve this, several non-functional requirements should be taken into account, not only looking at the short term but also on long term.

The existing system has turned out to be very reliable. It is however not easy/possible to add new, more modern functionality; for example integration with other devices like smartphones.

The new home domotica platform (both hard- and/or software) should comply with following non-functional requirements:

|  |  |  |
| --- | --- | --- |
|  | Requirement | Justification |
| Reliability | HW and program needs to run with limited to no manual maintenance activities or intervention. | “Turn-on and forget” concept should apply. Not all members of the family are technically skilled. Therefore the system needs to be able to recover from (most of the) errors without human intervention |
| Security | Unauthorized access to the home domotica system is not allowed.  System patching is possible | Initially the system will be used to control lights and power outlets via physical switches in the house. This is expected to change in the future where the system could also be used for other functionality like camera protection or be operated via a smartphone. This should only be possible by authorized people/devices. |
| Scalability | Home domotica system should be scalable for both hardware and/or software components. | The home domotica platform should be able to grow and support new features in the future. |
| Monitorability | Usage on both system and application level should be monitored and/or logged. | Identify issues in a proactive manner.  Perform trend analysis for potential cost optimization. |
| Availability | System needs to be available 24/7. | When the system is down, no lights can be turned on/off. |
| Supportability | The hard- and software used in the home domotica platform should be supportable on both short and long term. | As hardware evolves, the environment needs to remain supportable. Spare or replacement equipment will be required in case of a hardware failure.  The same applies for the software. Upgrades or patches will be required over time. |