energies

Article

The Architecture of a Real-Time Control System for Heating Energy Management in the Intelligent Building

Daniel Prusak [1,2], Grzegorz Karpiel [1,2] and Konrad Kułakowski [2,3,*]

[1] Department of Robotics and Mechatronics, AGH University of Science and Technology, 30-059 Kraków, Poland; daniel.prusak@agh.edu.pl (D.P.); gkarpiel@agh.edu.pl (G.K.)
[2] RIOT Sp. z o.o., 30-611 Kraków, Poland
[3] Department of Applied Computer Science, AGH University of Science and Technology, 30-059 Kraków, Poland
* Correspondence: kkulak@agh.edu.pl; Tel.: +48-126173408

Abstract: Very often, constructors and designers of intelligent building and building automation systems have a choice: to create a compact system with a limited configuration and modifying the system's behavior possibilities or provide a fully configurable solution at the expense of introducing a full SCADA system equipped with an additional knowledge database and inference system equipped with learning capabilities. In the presented work, we show that there is a third solution. Using a multilayer control system composed of programmable FPGAs, small PCs, and cloud computing resources, we can design and implement a fully configurable intelligent control system for the building's heating. Our solution combines the compactness of the structure and the ease of installation and assembly.

Keywords: intelligent building; intelligent heating control; FPGA; HVAC

Citation: Prusak, D.; Karpiel, G.; Kułakowski, K. The Architecture of a Real-Time Control System for Heating Energy Management in the Intelligent Building. *Energies* **2021**, *14*, 5402. https://doi.org/10.3390/en14175402

Academic Editor: George S. Stavrakakis

Received: 1 July 2021
Accepted: 24 August 2021
Published: 30 August 2021

1. Introduction

Intelligent building automation and control infrastructure are indispensable elements of many modern buildings scattered around the world. Some aspects of smart infrastructure are responsible for high-level functionalities provided to residents, such as entertainment, recreation, security (burglary detection), and network communication. However, other intelligent elements can support standard building functions such as heating, ventilation, lighting, etc. In the case of the former, customer satisfaction is a measure of their quality and effectiveness. The quality of systems ensures standard building functions; however, we must meet the requirements and standards [1–3]. So what can be further improved? One possible answer might be energy consumption. In the era of global warming, while still a large amount of energy is produced from fossil fuels, any attempt to decrease their use is worthwhile. It is crucial in Poland, where the energy allocated to hard coal and lignite combustion is estimated to 88% [4]. Coal combustion comes with many risks for the natural environment (carbon dioxide emission into the atmosphere, acid rains) and human health (particulate matter). Thus, reducing the amount of energy used is a concern for the climate, the natural environment, and human health. One of the main objectives of intelligent building automation and control systems is to increase energy efficiency while ensuring that all required standards and restrictions are met. We can realize increasing energy savings in many ways. One of them is improving the heating control system to predict heat demand and deliver optimal heat energy to the end-users. However, the construction of such a system is not easy. On the one hand, it must meet the requirements resulting from regulations and standards. On the other hand, it must use artificial intelligence algorithms to offer more advanced high-level control functions and forecast energy consumption. The need to minimize the cost makes the design and creation of the entire system even more difficult.

In the presented study, we propose an architecture of NOTOS™—a hybrid thermal energy management building system. The NOTOS system aims to control the amount of heat supplied to the building depending on the current demand, historical data, predicted consumption, and forecast. The presented architecture is complemented by the actual data of prototypes created.

2. Preliminaries

2.1. Intelligent Building Systems

To consider a subsystem as a part of an intelligent building, one must first answer what an intelligent building is. In the literature [5,6], there is no one correct answer but at least three different answerws used by the research centers dealing with this problem (In the case of artificial intelligence, the situation is quite similar. In their essential handbook, Russel and Norvig cite eight definitions trying to frame the phenomenon of AI [7] (p. 2)). The first definition focuses on performance. For example, according to the European Intelligent Building Group (EIBG), an intelligent building is designed to provide residents the most efficient environment and, at the same time, use resources efficiently, minimizing the costs of living.

A different approach is presented by the Japanese Intelligent Building Institute (JIB). According to JIBI, an intelligent building should provide to residents the following four services:

- receiving and sending information, supporting efficient management;
- ensuring satisfaction and comfort for residents;
- building management rationalization;
- adaptability to changes in the social environment and residents needs.

The third system-based definition recommended by the Chinese IB Design Standard assumes that intelligent buildings should have specific systems and technologies. In particular, such a building should provide building automation, communication network systems, and office automation. Moreover, an intelligent building should optimally integrate the structure, system, service, and management, providing high efficiency, comfort, and safety to habitants.

Regardless of the definition adopted, it seems that nowadays, an intelligent building is inextricably linked with a system that acquires and processes information and translates it into actions for the benefit of residents. This role is usually played by the IT system associated with the hardware layer responsible for collecting signals and performing actions. From the perspective of intelligent control, the effectiveness of the actions taken by the system is also important. Hence, the elements of the intelligent building system should improve its operation in a noticeable (measurable) way, e.g., reduce the price of use, increase comfort, improve safety, etc.

2.2. Hierarchical Real-Time Intelligent Control System

Usually, the concept of an intelligent building is not associated with the notion of a real-time system (RTS). This is probably because real-time (RT) systems are most often related to quick-change control of time-critical infrastructure or systems. For example, aircraft control (e.g., fly-by-wire system), advanced vehicle control systems such as ABS (anti-lock braking system), etc. However, the concept of an RTS is broader than the cases mentioned above. It is generally accepted that RT systems can be classified into two groups: hard and soft RT systems [8]. The first RTS type is when a system must respond over a specific, usually not too long, period. After that time, the computation result, even if it comes, does not matter. From the system point of view, it may then be too late for any action. For example, in such cases, the plane crashed, the device broke, the car skidded. The second kind of RT system has soft time constraints. Returning a delayed response does not make it useless but reduces its quality. Usually, the greater the delay, the greater the loss in response quality and the greater the loss (less benefit) for the overall system.

A typical example of such a system is a mobile phone with an application for playing streaming media.

The question arises, however, as to how artificial intelligence fits into this division. Again, there is no one correct answer, similarly, as there is no one good definition of Artificial Intelligence. Russel and Norvig mention at least eight definitions of this concept [7] (p. 2). For this article, we choose one of them coming from Kurzweil. According to this definition, AI is "the art of creating machines that perform functions that require intelligence when performed by people" [7,9]. Indeed, more sophisticated control that goes beyond simply reacting to events, a rule linked to planning and forecasting the future, requires intelligence on the human being. It suggests that in the case of RT systems, AI may be associated with a sufficiently large time horizon. This observation underlies the concept of a hierarchical real-time control system proposed by Albus, Quintero, Barbera, and others [10,11]. According to this approach, the behavior of a system determines the hierarchy. Every activity of a system is defined by decomposition into levels of resolution. The higher the level in the hierarchy, the longer the planning horizon and the lower the perceptual resolution of observed patterns. The authors propose that individual levels of the control hierarchy differ in resolution (accuracy) by order of magnitude. In their works, the authors consider seven levels of time decomposition of tasks in an intelligent control system: servo, primitive, move, individual, group 1, group 2, and world/group 3. The highest "world" level covers one day (Table 1).

Table 1. The hierarchical decomposition of activities in the Intelligent Real-Time Control System [10].

Level	Name	Event Detection Interval	Command Update Interval	Short Term Memory	Planning Horizon
1	servo	3 ms	3 ms	30 ms	30 ms
2	primitive	30 ms	30 ms	300 ms	300 ms
3	move	300 ms	300 ms	3 s	3 s
4	individual	3 s	3 s	30 s	30 s
5	group 1	30 s	30 s	5 min	5 min
6	group 2	5 min	5 min	1 h	1 h
7	world	1 h	1 h	events of the day	plan for the day

The inseparable elements of artificial intelligence in the hierarchical control model are actuators, sensors, sensory processing, world model, value judgment, and behavior generation (Figure 1) [10]. Sensors and actuators are equivalent to the input and output to the system. Sensors provide information about the current state of the world (system environment), while actuators can change the observable world. Sensory processing stands for the system module, which is responsible for translating sensoric values into the expected state of the world expressed in the form of knowledge. Another AI system element is the world model, often linked to a knowledge base [12]. The world model is the best representation of the external world's state of an intelligent control system. Very often, the module implementing the world model can carry out simulations and future state predictions. It can answer the questions about past, current, and future events, so it maintains a database of facts that happened in the past. The world model is kept up to date by appropriately processed sensory information. The value judgment module evaluates the current state of the world against its future states based on the knowledge kept in the world model. This module generates opinions indicating what is and is not beneficial to the system. These opinions are transferred to the module responsible for the generation of behavior. Here, the best plan for the system is selected and implemented. The immediate result of the decision to implement the plan is the activation of appropriate actuators.

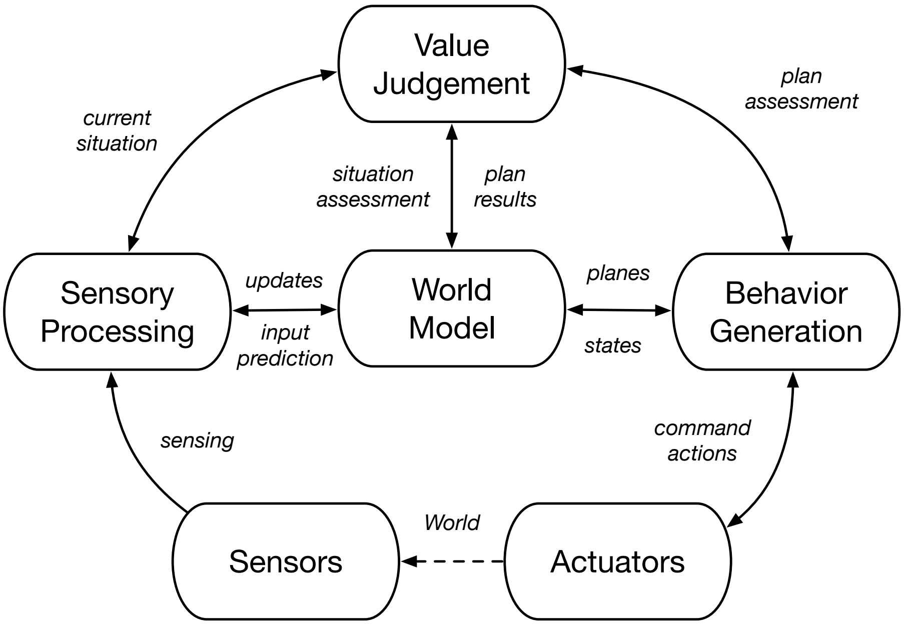

Figure 1. Intelligent control modules [10].

The model of the world takes center stage in the system developed by researchers at NIST. All the system components, except sensors and actuators, may cooperate with the knowledge stored in the world model. The higher the intelligence of the system, the more intense this collaboration can be. However, having a world model in the scheme seems like a good idea; processing the accumulated knowledge may be difficult and time-consuming. For this, the lower the level in the hierarchy (Table 1), the smaller and simpler it uses the world model. Finally, it is worth noting that in the context of control systems, the MPC (Model Predictive Control) architecture is being developed [13]. The model in this approach most often has a relatively simple functional form (e.g., linear functions), allowing for the prediction of future states of the system. Therefore, MPC systems are usually not considered intelligent.

The reverse approach to the design of intelligent real-time systems comes from Rodney Brooks [14]. His subsumption architecture control system is composed of many different behaviors, of which the more complex, higher-ranking ones may supersede the simpler ones with the lower priority. In their view, "the world is its own best model". Thus, systems based on the subsumption architecture are usually purely reactive. In practice, intelligent RT control systems are a combination of hierarchical and reactive approaches. The higher the level in the control hierarchy, the less time-critical it is responding to stimuli, and the more important planning is. Conversely, low levels in the control hierarchy (servo, primitive) can be purely reactive, while high levels intensively use the knowledge accumulated in the world model.

2.3. Related Works

Although originally hierarchical real-time systems were developed in the context of autonomous vehicles [11,15], they soon became used in many other fields of applications. Lefort et al. present an example of the use of a hierarchical real-time control system [16]. They use a model predictive control (MPC) approach. In this approach, the world model is used to predict the state of a process iteratively. Such prediction supports the control module in selecting optimal parameters. The proposed model [16] comprises the higher scheduling level and the short-term piloting level.

Another example of a two-layer hierarchical control system comes from Dounis et al. [17]. The system built based on the agent paradigm consists of a coordinator and five fuzzy logic controllers. A further example of fuzzy control in intelligent building systems can be found in [18]. Dounis and Caraiscos [2] took up the problem of determining the thermal comfort of inhabitants. Castilla et al. [3] collected several definitions and indices relating to various aspects of the comfort of building occupants. A substantial part of the book is devoted to the hierarchical architecture of building control systems. MPC approach dating back to ideas from the sixties [13] will provide endless inspiration for researchers and practitioners. Hazyuk [19] proposed a control system that allows one to determine the optimal heating temperature of the building based on the MPC approach. Ma et al. [20] modeled an MPC of a chilled water tank, helping minimize energy consumption while preserving operational constraints and cooling demand. Yuan and Perez propose using the MPC strategy to control multiple-zone ventilation and temperature [21]. Despite the popularity of the MPC solutions, more advanced systems using knowledge processing and artificial intelligence mechanisms can also be noticed [22]. When creating intelligent building systems, the designer must take into account the cooperation with external systems, e.g., the intelligent power grid [23,24]. The presence of more and more complex control systems implementing more and more advanced heterogeneous strategies will probably be the direction of development in the coming years [25,26].

More information about different types of intelligent building control systems can be found in the review [27].

Of course, building automation is not only controlled by complex and intelligent systems. Peeters et al. [28] focus on determining the efficiency of a simple control model of a heating system consisting of a gas boiler and radiators. Very often, the boiler and thermostat are controlled on an on/off basis. Their work indicates that the simple heating control systems are not adequately adapted to the actual operating conditions, which, in turn, may result in oversizing. Of course, PID regulators could not be missing among building control systems. According to Wang [6], the percentage of PID algorithms used in closed control loops for buildings can be more significant than 80%.

A complete overview of different control techniques used in buildings can be found in the [3] (pp. 152–153).

2.4. Embedded Controllers

Effective control of heating in an intelligent building cannot be done without hardware components. They are often implemented as a SoC (System on a chip) or a SoPC (System on a programmable chip).

In the SoC (System-on-a-chip) approach, the user receives a ready-made microprocessor system developed by the manufacturer. Such a system is implemented in one integrated circuit. It has a microprocessor, memory, input/output circuits, and other dedicated function blocks such as counters, pulse-width modulation (PWM) generators, etc. SoC architecture is often used in embedded PCs, industrial computers, IoT devices (Internet of Things), and PLC controllers (programmable logic controllers). The advantages of SoC are low energy consumption, a broad spectrum of implemented microprocessors, easily accessible, ready-to-use libraries, and lower programming requirements. The obvious drawback of this approach is the loss of flexibility.

The SoPC approach is a natural extension of the FPGA (field-programmable gate array) concept. It extends the SoC architecture, increasing its flexibility and ease of adaptation to specific applications. SoPC uses programmable blocks containing logic elements LE (Logic Elements) that can create any functional structure. We can use logical parts to build: microprocessors, input-output circuits, and, above all, blocks dedicated to the project underway. These blocks can implement completely new functionalities not provided for by the FPGA manufacturer. The essential advantages of SoPC include:

- Ease of adding more cores;
- Structural flexibility, e.g., the system may contain dependent or independent subsystems;
- Including customized functional blocks.

The SoPC system has one drawback. It is limited by the size of the FPGA used. Hence, we should plan the resources of the FPGA with great care and engineering knowledge.

FPGA functional blocks are called IP-Core (Intellectual Property Core). They are, in fact, computing subunits that perform a specifically described functionality, which, in itself, may constitute intellectual property. IP-Core blocks are described using hardware description languages, such as Verilog or VHDL (Very High-Speed Integrated Circuits Hardware Description Language).

3. NOTOS System

NOTOS is a proprietary system for controlling a heating substation co-designed by the RIOT company. It comprises several components, including a real-time controller, communication layer, and cloud computing module. The purpose of the components is to ensure the right amount of heat in the heating system of the building, meeting the standards and minimizing energy consumption.

3.1. System Assumptions

The NOTOS system is designed to control the heat in individual buildings. Each building has at least one heating substation. A heating substation's minimal and maximal power is estimated to be 50 kW and 200 kW, respectively. The input and output temperatures of the heating medium are approximately 70 and 50 degrees Celsius, respectively.

The purpose of the NOTOS is to control one or more heating substations whose goal is to transfer the heat supplied by the heating network through the heating plant to the individual apartments, and further, to radiators. The creation of such a system poses several challenges for designers. In particular, it should provide

- fault tolerance and intrusion prevention;
- access control and preventing against unauthorized access;
- a mechanism for updating and tuning the system during its operation;
- monitoring the correctness of the performed operations;
- access to historical data;
- system design open for further development;
- real-time control capability;
- autonomous work within defined boundaries.

Different parts of the system meet these assumptions. For example, a real-time response time to incoming control signals is provided by the hardware part of the system. At the same time, the security of information processing must be implemented appropriately by each component of the created solution.

All modules work together to control a heating substation in a multi-apartment residential building. The amount of heat consumed in the apartments determines the amount of heat the substation needs to provide heat. It is assumed that each apartment has a thermostat that allows the resident to set the desired temperature of the heating medium. Interestingly, even though the possibility of this regulation, residents often treat thermostats as on/off appliances [29], i.e., they set them either "fully off" or "fully on". Hence, one of the most straightforward operating modes of the system is to react simply to switching thermostats from "on" to "off" or vice versa.

3.2. System Components

The NOTOS system consists of three modules (Figure 2):

- NOTOS embedded—a hardware part of the system responsible for the actual control of heating substation;
- NOTOS communication—a communication module responsible for the safe data transfer between the hardware layer and the cloud;
- NOTOS cloud—a cloud application responsible for long-term planning, tuning, data collection, and analysis.

Of course, communication between these modules requires intermediary hardware and software, which, although necessary in practice, have been omitted in this paper.

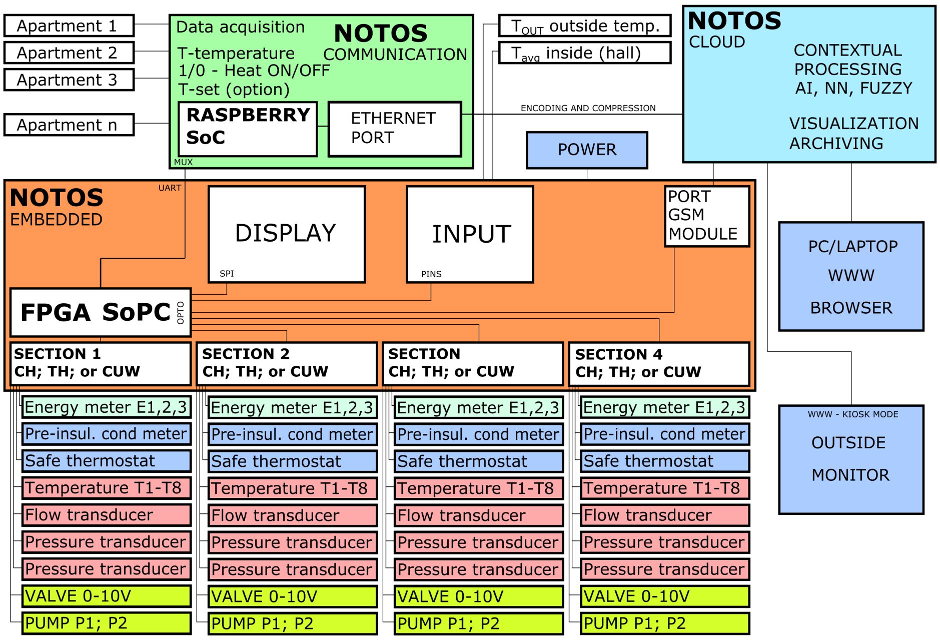

Figure 2. Components of the NOTOS™ system.

3.2.1. Hardware Controller

NOTOS embedded is a critical element of the system. On the one hand, it must ensure correct and standard-compliant control of the substation. On the other hand, it must be open to intelligent long-term control and scheduling carried out by cloud applications. The hardware unit should also be easily accessible on the market and offer significant configuration capabilities. Following these premises for implementing NOTOS embedded, SoPC Intel/Altera FPGA Family Max10 was selected. Based on this FPGA chip, the IPCore NIOS™ II processor module [30] and the NIOS II processor have been used as the execution environment for the heating control algorithms. Algorithms have been implemented in C++.

NOTOS embedded works in the following three modes: autonomous, network, and intermittent (mixed mode). Autonomous mode means that the device works independently. In this mode, it does not have access to the Internet, without communicating with the cloud environment, and without access to external data, also defined by artificial intelligence algorithms (AI cloud processing).

The network mode determines the operation of the system with full network access. In this mode, the device can communicate with the cloud environment and access external data, including the results of intelligent processing in the cloud.

An intermittent mode alternately defines intermittent operation. Such a situation may occur in the absence of communication. For example, it may happen due to a system failure or the modernization of the building installation with which the NOTOS controller cooperates. Intermittent operation determines the alternating and random shortages of external communication signals with a cloud-based environment or in the field of a network of intelligent thermostat sensors located in the rooms.

The NOTOS embedded hardware controller is based on the PID algorithm. For the NOTOS hardware module simulation, we assumed that the control algorithm would be a discretized model of the universal MATLAB™/Simulink™ PID block. In Figure 3, which shows the simulation model, there is an additional block “unit delay”. Its purpose is to remember the value for the next simulation step.

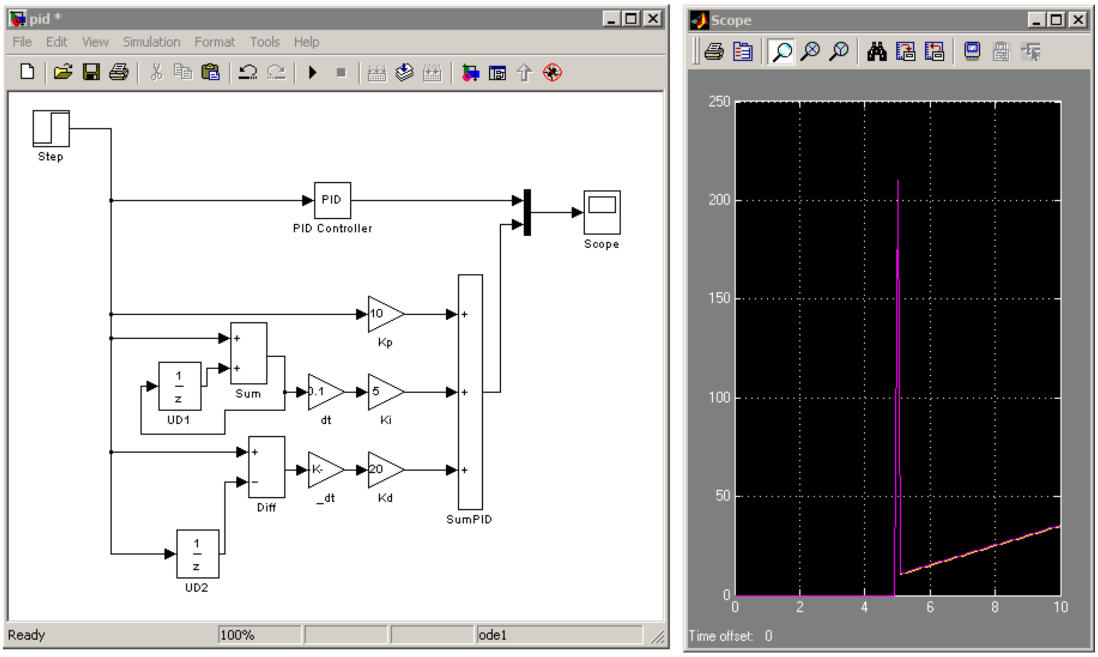

Figure 3. NOTOS™ Hardware module, based on the PID algorithm simulation.

The created simulation version served two purposes. The first was to conduct simulation tests and verify the correct implementation of the base algorithm (yellow is the waveform from the original PID block, and purple is the result from the discrete version—see Figure 3). The second was to prepare the hardware implementation. As a result of the tests carried out, we decided to use a 32-bit processor NIOS II (as IPCore implemented in FPGA) with an arithmetic co-processor and a software implementation of the system. (Although the NIOS II requires almost three times as many logical elements as the direct implementation of the PID algorithm, it offers the flexibility needed to implement functions related to interaction with the cloud environment.)

Currently, the PID algorithm has been implemented using Intel™ MAX 10 Eval Kit board (Figure 4).

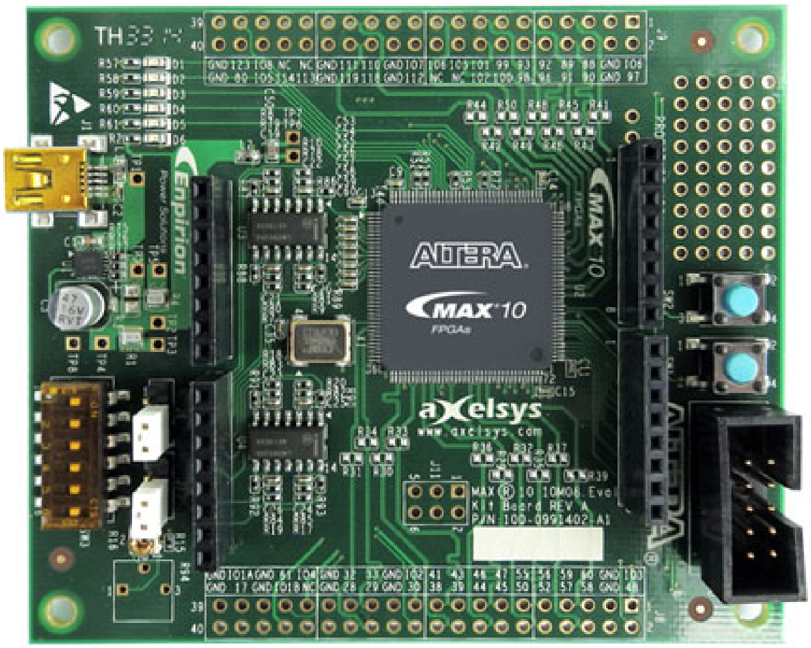

Figure 4. The board used for PID implementation.

We implemented the algorithm according to the following scheme (Figure 5), where the meaning of the blocks is as follows:

- AltPll0 block (library) converts input frequency 50 MHz to 10 MHz;
- Counter block (implemented) generates frequency for 100 Hz PID;
- UnitDelay block (implemented) latches the input data using trigger frequency (i.e., 100 Hz);
- ADD_FP and SUB_FP (library) are floating-point blocks responsible for adding and subtracting. They work in single-precision arithmetic with one cycle latency, 10 MHz frequency (42 MHz is the highest available frequency), 938 LUT;
- MK_FP (library) is a single-precision multiplication floating point block with the two cycles latency, maximal frequency 46 MHz, and LUT 288.

The above diagram (Figure 5) translates to 3565 logic elements (44% of all available), 300 registers, using 65 pins (101 possible). Finally, the PID algorithm works at 100 Hz with the possibility of increasing the speed to 1MHz. The output control is calculated, on average, in 6 steps. Thus, for 10MHz, the complete calculation can take 600 ns. As the maximum clock speed of the system is 40 MHz, we can reduce the computation time to 150 ns, where PID works with 20 MHz integration and differentiation.

3.2.2. Communication Module

The NOTOS communication module aims to mediate between the hardware controller and the intelligent processing layer in the cloud. The module construction is based on the Raspberry PI computer operating under the control of the Linux operating system. It is placed close (in the same housing) to the hardware module. Thanks to this, we can implement communication between them based on a simple unsecured protocol. It allows us to transfer communication reliability and safety problems to Raspberry Pi and relieve the NIOS II processor. In particular, this module is responsible for handling an internet connection and encryption of input/output information.

3.2.3. Cloud Subsystem

The NOTOS cloud is part of the system placed in a cloud environment. The main aims of this module are the storage of measurement data, analysis of collected information, visualization, and modifying the operating parameters of the hardware controller. Data analysis will use artificial intelligence mechanisms, including machine learning. A cloud subsystem will also be responsible for maintaining the system model. This model will allow the NOTOS cloud to interpret and adequately react to incoming hardware information. In particular, the NOTOS cloud will avoid the usage of any vendor-specific solutions or services. Currently, two proprietary cloud solutions are considered: Amazon (AWS—Amazon Web Services) and Google. To avoid vendor lock problems, we proposed an implementation quite loosely related to the cloud platform in which it operates.

3.3. NOTOS as an Intelligent Control Hierarchy

The NOTOS system consisting of three main modules can also be divided into three layers (Figure 6). The lowest reactive level is responsible for the immediate processing of incoming signals and a fast response generation. In the case of NOTOS, hardware input signals indicate thermostats equipped with the ability to set the desired temperature and measure the current temperature in the individual apartments, outside temperature, and the temperature of the heating water entering and coming out of the building. Behavior generation is limited to determining the appropriate valve settings on the pipe with the heating water. The PID algorithm is responsible for the correctness of the controller's operation. In the context of the Albus model (Section 2.2), NOTOS hardware is a complete intelligent control system operating, but in real-time rigor. The PID algorithm represents the role of the "world model" and the "value judgment".

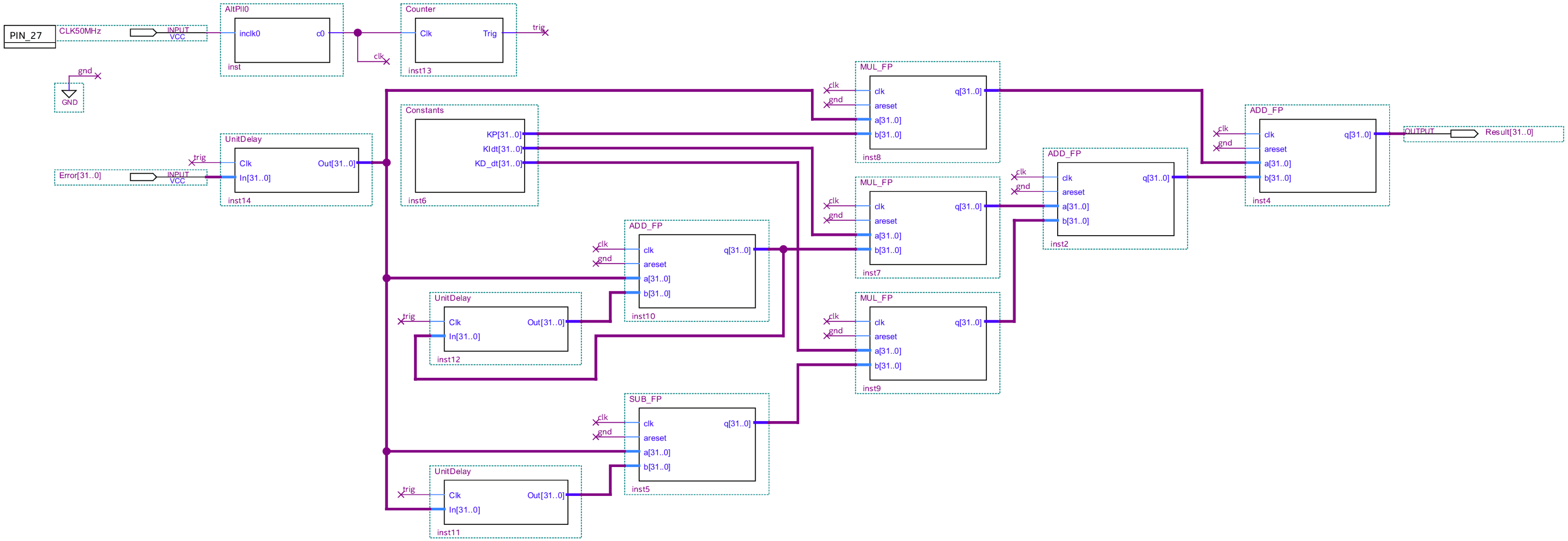

Figure 5. The PID scheme—hardware implementation.

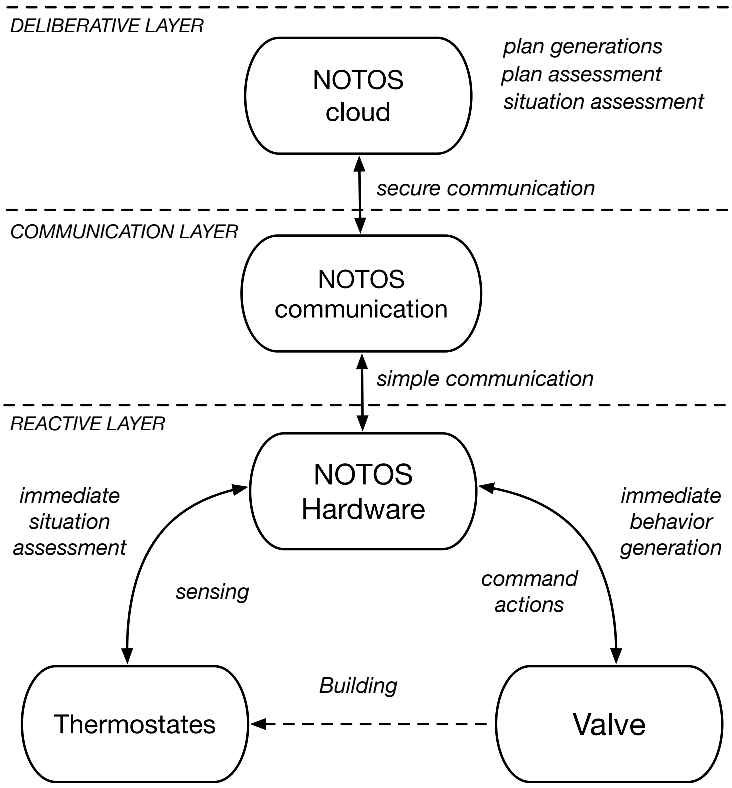

Figure 6. NOTOS™ control hierarchy.

The cloud part of the system is responsible for long-term intelligent planning, data analysis, and strategic decisions. The PID algorithm affects the incoming information from the cloud. There are two types of such decisions. They can ultimately interfere with the PID algorithm (e.g., a decision to enable or disable heating) or improve its operation. The latter can take place by modifying the parameters of the algorithm. Furthermore, on this intelligent (deliberative) level, we deal with the world model. It is present here in the form of several parameters stored in the cloud together with their change history. A value judgment is the result of AI used for data analysis and processing. Deliberative layer algorithms do not have to work immediately. Instead, they work slower, analyzing more considerable amounts of data. However, as in our model, the cloud application has priority over the hardware control, then it can change both the PID operating parameters and the valve control signals. Hence, unlike the Albus model, the responsibilities between the various control levels are not strictly delimited.

The last communication level has an auxiliary purpose. It complements the functionality of the hardware module, allowing for safe (including wireless) communication between the hardware module and the cloud application. It does not have any role in the processing of signals or generation of system behavior.

The presented system is well suited for the implementation of the MPC approach. First, at the cloud level, we can implement the model and perform the control optimization. In this case, each algorithm loop requires a two-way exchange of messages (measurements and control signals) between the cloud and the hardware controller. The goal of the latter will be two-fold. First, it will act as a proxy between the cloud and sensors-effectors layer. Second, however, in an emergency, e.g., when the connection between the cloud and the controller is lost, it will be able to take control of the heating substation.

The designed system can also support MPC locally at the hardware driver level. As this driver uses an FPGA processor, it is possible to implement the entire control loop as the SoPC directly. Of course, this requires a sufficiently large FPGA chip, but it reduces the network communication and, hence, reduces the dangers of losing contact with the cloud. The role of the cloud application can be reduced to supervising the operation of the local controller.

In the presented approach, we chose an intermediate solution. Even though the local driver is quite simple, it can run on its own. This secures the system against losing contact with the cloud application. It can also run in proxy mode depending on the instructions received from the cloud application. This aspect of the presented control structure is similar to Brooks' subsumption architecture [14].

3.4. Self-Adaptation

One of the features of intelligent control systems is the far-reaching ability to adapt control algorithms to changing operating conditions. In buildings, such a change may include thermal modernization (replacement of windows, doors, adding wall insulation, heat pumps etc.). Thanks to the data analysis, the NOTOS cloud can change the operating parameters of the heating substation and thus adapt the system to work in a significantly changed environment in a way that is invisible to the user. We cannot overestimate this possibility. The research carried out by the project partner during the process of energy efficiency certificates (aka the white certificates) obtaining procedure shows that energy losses resulting from the failure to adapt the control of the heating substation to the changed energy conditions of the building may reach up to 30% of the energy consumed.

4. Summary

The presented article shows the architecture of a modern hierarchical control system of a heating substation. Component layouts and system layers are based on the hierarchical control structure with the world model. The presented approach is open to various AI algorithms. The use of a computing cloud ensures scalability and access to high computing power. On the other hand, due to the relatively small complexity of the hardware module, the installation process and system maintenance cost are relatively small. Combining these two features makes the presented solution interesting for both practitioners and designers of such systems. Practitioners can see the simplicity and reliability of the concept of the hardware component. Designers will appreciate the flexibility of architecture and ample possibilities hiding in the use of cloud technology.

Author Contributions: Conceptualization, hardware and software design, testing and investigation, D.P., G.K.; solution architecture, project administration and supervision, D.P.; Conceptualization, methodology, literature review and article preparation, K.K. All authors have read and agreed to the published version of the manuscript.

Funding: This material is based upon work supported by the National Centre for Research and Development (NCBR) within the project POIR.01.02.00-00-0308/17.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Restrictions apply to the availability of these data. The collected data is the result of research conducted by RIOT sp z o.o. and Ekoprojekt sp. z .o.o.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Wilkins, C.K.; Doeffinger, R.C.; Rutkowski, H.T. *Peak Cooling and Heating Load Calculations in Buildings Except Low-Rise Residential Buildings*; Technical Report; American Society of Heating, Refrigerating and Air-Conditioning Engineers: Atlanta, GA, USA, 2020.
2. Dounis, A.I.; Caraiscos, C. Advanced control systems engineering for energy and comfort management in a building environment—A review. *Renew. Sustain. Energy Rev.* **2009**, *13*, 1246–1261. [CrossRef]
3. Castilla, M.; Domingo Álvarez, J.; Rodríguez, F.; Berenguel, M. *Comfort Control in Buildings*; Springer: London, UK, 2014.
4. Marks-Bielska, R.; Bielski, S.; Pik, K.; Kurowska, K. The importance of renewable energy sources in Poland's energy mix. *Energies* **2020**, *13*, 4624. [CrossRef]
5. Omar, O. Intelligent building, definitions, factors and evaluation criteria of selection. *Alex. Eng. J.* **2018**, *57*, 2903–2910. [CrossRef]
6. Wang, S. *Intelligent Buildings and Building Automation*; Spon Press: New York, NY, USA, 2010; p. 260.

7. Russell, S.; Norvig, P. *Artificial Intelligence: A Modern Approach*, 3rd ed.; Prentice Hall: Hoboken, NJ, USA, 2009.
8. Gupta, A.; Chandra, A.K.; Luksch, P. *Distributed Real-Time Systems*; CRC Press: Boca Raton, FL, USA, 2020; pp. 46–63.
9. Kurzweil, R. *The Age of Intelligent Machines*; MIT Press: Cambridge, MA, USA, 1990.
10. Albus, J.S. Outline for a Theory of Intelligence. *IEEE Trans. Syst. Man Cybern.* **1991**, *21*, 473–509. [CrossRef]
11. Quintero, R.; Barbera, A. *A Real-Time Control System Methodology for Developing Intelligent Control Systems*; NIST: Gaitheresburg, MD, USA, 1992
12. Kułakowski, K.; Wąs, J. World model for autonomous mobile robot—Formal approach. In Proceedings of the Intelligent Information Systems, Siedlce, Poland, 8–10 June 2010; pp. 37–45.
13. Morari, M.; Lee, J.H. Model predictive control: Past, present and future. *Comput. Chem. Eng.* **1999**, *23*, 667–682. [CrossRef]
14. Brooks, R. Planning Is Just a Way of Avoiding Figuring out What to Do Next; Technical Report; MIT Artificial Intelligence Laboratory: Cambridge, MA, USA, 1987.
15. Albus, J.; Huang, H.M.; Lacaze, A.; Schneier, M.; Juberts, M.; Scott, H.; Balakirsky, S.; Shackleford, P.W.; Hong, T.; Murphy, K.; et al. 4D/RCS: A Reference Model Architecture for Unmanned Vehicle Systems Version 2.0; Technical Report; NIST: Gaitheresburg, USA: 2002. [CrossRef]
16. Lefort, A.; Bourdais, R.; Ansanay-Alex, G.; Guéguen, H. Hierarchical control method applied to energy management of a residential house. *Energy Build.* **2013**, *64*, 53–61. [CrossRef]
17. Dounis, A.I.; Tiropanis, P.; Argiriou, A.; Diamantis, A. Intelligent control system for reconciliation of the energy savings with comfort in buildings using soft computing techniques. *Energy Build.* **2011**, *43*, 66–74. [CrossRef]
18. Papantoniou, S.; Mangili, S.; Mangialenti, I. Using Intelligent Building Energy Management System for the Integration of Several Systems to one Overall Monitoring and Management System. *Energy Procedia* **2017**, *111*, 639–647. [CrossRef]
19. Hazyuk, I.; Ghiaus, C.; Penhouet, D. Optimal temperature control of intermittently heated buildings using Model Predictive Control: Part II - Control algorithm. *Build. Environ.* **2012**, *51*, 388–394. [CrossRef]
20. Ma, Y.; Borrelli, F.; Hencey, B.; Packard, A.; Bortoff, S. Model Predictive Control of Thermal Energy Storage in Building Cooling Systems. In Proceedings ofthe 48th IEEE Conference on Decision and Control, Shanghai, China, 15–18 December 2009; pp. 392–397.
21. Yuan, S.; Perez, R. Multiple-zone ventilation and temperature control of a single-duct VAV system using model predictive strategy. *Energy Build.* **2006**, *38*, 1248–1261. [CrossRef]
22. Chen, Y. Building Control Knowledge Information Modeling and Control Self-Configuration. Ph.D. Thesis, The Pennsylvania State University, State College, PA, USA, 2015.
23. Babar, M.; Grela, J.; Ożadowicz, A.; Nguyen, P.H.; Hanzelka, Z.; Kamphuis, I.G. Energy Flexometer: Transactive Energy-Based Internet of Things Technology. *Energies* **2018**, *11*, 568. [CrossRef]
24. Ożadowicz, A.; Grela, J. An event-driven building energy management system enabling active demand side management. In Proceedings of the 2016 Second International Conference on Event-based Control, Communication, and Signal Processing (EBCCSP), Kraków, Poland, 13–15 June 2016; pp. 1–8. [CrossRef]
25. Lilis, G.; Conus, G.; Asadi, N.; Kayal, M. Towards the next generation of intelligent building: An assessment study of current automation and future IoT based systems with a proposal for transitional design. *Sustain. Cities Soc.* **2017**, *28*, 473–481. [CrossRef]
26. Treado, S.; Chen, Y. Saving building energy through advanced control strategies. *Energies* **2013**, *6*, 4769–4785. [CrossRef]
27. Dounis, A.; Caraiscos, C. Fuzzy comfort and its use in the design of an intelligent coordinator of fuzzy controller-agents for environmental conditions control in buildings. *J. Uncertain Syst.* **2008**, *2*, 101–112.
28. Peeters, L.; Veken, J.V.D.; Hens, H.; Helsen, L.; D'haeseleer, W. Control of heating systems in residential buildings : Current practice. *Energy Build.* **2008**, *40*, 1446–1455. [CrossRef]
29. Meier, A.; Aragon, C.; Hurwitz, B.; Peffer, P.; Pritoni, M. How People Actually Use Thermostats. In Proceedings of the 2010 ACEEE Summer Study on Energy Efficiency in Buildings, Pacific Grove, CA, USA, 15–20 August 2010; pp. 193–206.
30. Intell. *Nios® II Processor Reference Guide*; Intel: Santa Clara, CA, USA, 2020.