

TALLINN UNIVERSITY OF TECHNOLOGY  
School of Information Technologies

Dermansah Hazar Ersoy Yalin 184583IVEM

# **Investigation of Model Predictive Control in Bioanalytical Devices**

Master's thesis

Supervisor: Yannick Le Moullec  
PhD

Kanwal Ashraf  
MSc

Tallinn 2021

TALLINNA TEHNIKAÜLIKOOL  
Infotehnoloogia teaduskond

Dermansah Hazar Ersoy Yalin 184583IVEM

***‘Model Predictive Control’i uurimine***  
**bioanalüütilistes seadmetes**

Magistritöö

Juhendaja: Yannick Le Moullec  
PhD

Kanwal Ashraf  
MSc

Tallinn 2021

## **Author's declaration of originality**

I hereby certify that I am the sole author of this thesis. All the used materials, references to the literature and the work of others have been referred to. This thesis has not been presented for examination anywhere else.

Author: Dermansah Hazar Ersoy Yalın

10.05.2021



## **Abstract**

Low-cost portable bioanalytical devices based on lab-on-a-chip/droplet microfluidics can be helpful to enable fast response to bacterial threats as well as for detecting infectious diseases by means of point-of-care testing. In turn, such devices can help reduce epidemic risks and save human lives.

Such bioanalytical devices include several sub-systems that require to be carefully controlled for proper operation. Among existing control methods, Model Predictive Control (MPC) has been used in industrial processes since the 1980s; more recently it has gained increased attention for the implementation of a wider range of applications. A salient feature of MPC is its ability to optimize the current timeslot while accounting for the next ones; the key mechanism is to consider a time-horizon that is finite but only implement the current timeslot and repeat the optimization. Additionally, it solves an optimization problem based on a plant model to predict the future plant outputs as close as to the desired output reference value.

However, the potential of MPC for resource-constrained platforms needed for realizing low-cost portable bioanalytical devices needs to be further investigated, which is the objective of this MSc thesis. The work presented in this thesis begins with an overview of the basic principles of two examples of bioanalytical devices based on lab-on-a-chip/droplet microfluidics, namely a flow cytometer and an isothermal DNA amplification system. Next, the work presents the main concepts of various control methods and special attention is given to MPC and the related embedded convex optimization. System design considerations are presented in terms of candidate resource-constrained platforms. Finally, simulation results illustrate the behavior of MPC depending on the tuning of its parameters, in particular for an example of temperature regulation using a heating PCB component.

This thesis is written in English language and is 66 pages long, including 5 chapters, 30 figures and 7 tables.

## **Annotatsioon**

### **‘Model Predictive Control’i uurimine bioanalüütilistes seadmetes**

Odavad kaasaskantavad bioanalüütilised seadmed, mis põhinevad labor-kiibil tehnoloogial / tilgapõhisel mikrofluidikal, saavad kanda olulist rolli bakteriaalsetele ohtudele reageerimisel ja hoolduskohas tehtavate testide abil nakkushaiguste avastamisel. Taolised seadmed võivad aidata vähendada epideemiariski ja päästa inimesi.

Sellised bioanalüütilised seadmed sisaldavad mitmeid alamsüsteeme, mida tuleb hoolikalt kontrollida nõuetekohase töö tagamiseks. Eksisteerivate kontrollimeetodite hulgas on mudeli-põhine ennustav juhtimine (Model Predictive Control, MPC) olnud kasutusel alates 1980ndatest, kuid viimasel ajal on see saanud suuremat tähelepanu seoses rakendamisega laiemal valikul seadmetel.

MPC silmapaistvaim funktsioon on selle võime optimeerida hetkelist ajavahemikku arvestades samal ajal järgmistega; selle põhiline mehhanism on arvestada prognoositava ajaga, mis on piiratud kuid rakendatakse ainult praegune ajavahemik ja korratakse optimeerimist. Lisaks lahendab see optimeerimise probleemi tuginedes tehasemudelil, prognoosib tulevasi tehase toodanguid nii täpselt, kui soovitud toodangu kontrollväärtus. Käesoleva magistritöö eesmärk on uurida edasi MPC potentsiaali piiratud ressurssidega platvormidel, mida on tarvis odavate ja kaasaskantavate bioanalüütiliste seadmete jaoks. See töö algab ülevaatega kahe bioanalüütilise seadme (voolutsütomeeter ja isotermilise DNA amplifikatsioonisüsteem) näitel, mis põhinevad labor-kiibil tehnoloogial / tilgapõhisel mikrofluidikal.

Järgmiseks tutvustatakse erinevate juhtimismeetodite põhimõisteid ning erilist tähelepanu pööratakse MPC-le ja sellega seotud sisseehitatud kumerale optimeerimisele. Süsteemi ülesehitus esitatakse seoses ressurssidega piiratud platvormide osas, mille kasutamist võiks kaaluda. Lõpetuseks simulatsioonitulemused illustreerivad MPC käitumist sõltuvalt selle parameetrite häälestamisest, eriti temperatuuri reguleerimisel kasutades PCB kuumutamist.

Lõputöö on kirjutatud inglise keeles ning sisaldab teksti 66 leheküljel, 5 peatükki, 30 joonist ja 7 tabelit.

## List of abbreviations and terms

CPU	Central processing unit
DMC	Dynamic matrix control
DNA	Deoxyribonucleic acid
DSP	Digital signal processor
FC	Flow cytometry
FFC	Fluorescence flow cytometry
FPGA	Field programmable gate arrays
GMO	Genetically modified organisms
GPU	Graphics processing units
IDE	Integrated development environment
LAMP	Loop-mediated isothermal amplification
MCU	Microcontroller unit
MPC	Model predictive control
MV	Manipulated Variable
NAAT	Nucleic acid amplification test
PC	Personal computer
PCB	Printed circuit board
PCR	Polymerase chain reaction
PID	Proportional-Integral-Derivative
PoC	Point of care
PWM	Pulse-width modulation
QP	Quadratic programming
SBC	Single board computer
SoC	System on chip
USB	Universal serial bus
WiFi	Wireless fidelity

# Contents

<b>Author’s declaration of originality .....</b>	<b>3</b>
<b>Abstract .....</b>	<b>4</b>
<b>Annotatsioon ‘Model Predictive Control’i uurimine bioanalüütilistes seadmetes... 5</b>	<b>5</b>
<b>List of abbreviations and terms.....</b>	<b>7</b>
<b>List of figures .....</b>	<b>10</b>
<b>List of tables .....</b>	<b>12</b>
<b>1 Introduction .....</b>	<b>13</b>
1.1 Concise Description of Flow Cytometry .....	13
1.2 Concise Description of Isothermal Nucleic Acid Amplification (NAAT) .....	15
1.3 Problem Statement and Steps of the Thesis.....	16
1.4 Organization of thesis .....	17
<b>2 Background .....</b>	<b>18</b>
2.1 Flow Cytometry .....	18
2.1.1 Main concepts.....	18
2.2 Isothermal Nucleic Acid Amplification (NAAT).....	21
2.3 Concepts of control.....	23
2.3.1 Control Algorithms.....	24
2.3.2 Feedback control .....	24
2.3.3 Proportional control.....	25
2.3.4 Proportional-Integral control .....	26
2.3.5 Proportional-Integral-Derivative control.....	27
2.3.6 Model Predictive Control .....	29
2.3.7 Control Signals .....	34
<b>3 System Design .....</b>	<b>37</b>
3.1 Hardware, Software and Communication Technologies.....	37
3.1.1 SBCs .....	37
3.1.2 Software.....	39
3.1.3 Communication Technologies.....	40
<b>4 Experimental Results and Discussion .....</b>	<b>46</b>

4.1 Experiments for a heating pad .....	46
4.1.1 Tuning the sample time for the heating pad .....	49
4.1.2 Tuning the prediction horizon for the heating pad .....	53
4.2 Experiments with a PCB heater and thermistor.....	55
4.2.1 Tuning the sample time for the PCB heater and thermistor .....	59
4.2.2 Tuning the prediction horizon for the PCB heater and thermistor .....	60
<b>5 Conclusion .....</b>	<b>65</b>
5.1 Summary.....	65
5.2 Perspectives .....	66
<b>References.....</b>	<b>67</b>
<b>Appendix 1 – Non-exclusive licence for reproduction and publication of a graduation thesis.....</b>	<b>69</b>

## List of figures

Figure 1: Example of an FC system [1] composed of five major elements 1) the fluidics system, 2) the laser source, 3) the mirror system, 4) the detectors and 5) the computer system. ....	19
Figure 2: Illustration of the DaimonDNA biosensor. Part a) shows the principle of operation for the realtime monitoring of the biosensor. Part b) shows the various components used for the construction of biosensor, including the electronic PCB and heater [7].....	21
Figure 3: Illustration of the workflow of the DaimonDNA biosensor[7].....	22
Figure 4: Illustration of the temperature curve of the DaimonDNA biosensor controlled by a PID algorithm. The target temperature is 65°C with an accuracy of $\pm 0.1$ °C [7]..	23
Figure 5: Generalized block diagram of feedback control system. ....	25
Figure 6: The working principle of a proportional controller .....	26
Figure 7: The working principle of a proportional-integral controller .....	27
Figure 8: A PID controller block diagram in a feedback loop .....	28
Figure 9: Characterization of the Model Predictive Control [32] .....	29
Figure 10: Dynamic Matrix Control working principle scheme .....	32
Figure 11: Predictive Functional Control working principle scheme.....	34
Figure 12: Function of the general solver and a code generator for embedded convex optimization .....	34
Figure 13: Obtained curve for heating pad in Matlab Curve Fitting .....	46
Figure 14: Single input single out MPC Algorithm in Matlab Simulink .....	48
Figure 15: Graph of change of temperature in time, based on MPC with 1 second of sample time, prediction horizon with 50 moves, control horizon with 2 steps. ....	50
Figure 16: Graph of change of temperature in time, based on MPC with 0.8 second of sample time, prediction horizon with 50 moves, control horizon with 2 steps. ....	51
Figure 17: Graph of change of temperature in time, based on MPC with 0.5 second of sample time, prediction horizon with 50 moves, control horizon with 2 steps. ....	52
Figure 18: Graph of change of temperature in time, based on MPC with 0.2 second of sample time, prediction horizon with 50 moves, control horizon with 2 steps. ....	52

Figure 19: Graph of change of temperature in time, based on MPC with 0.2 second of sample time, prediction horizon with 30 moves, control horizon with 2 steps. ....	53
Figure 20: Graph of change of temperature in time, based on MPC with 0.2 second of sample time, prediction horizon with 40 moves, control horizon with 2 steps. ....	54
Figure 21: Graph of change of temperature in time, based on MPC with 0.2 second of sample time, prediction horizon with 60 moves, control horizon with 2 steps. ....	55
Figure 22: Image of NCP21XV103J03RA PCB heater and thermistor .....	56
Figure 23: Measuring the output resistance of NCP21XV103J03RA.....	56
Figure 24: Obtained curve of temperature ( $^{\circ}\text{C}$ ) as a function of resistance ( $\Omega$ ) for NCP21XV103J03RA in Matlab Curve Fitting.....	57
Figure 25: Graph of the temperature change of the PCB heater and NCP21XV103J03RA thermistor in time, based on MPC with 0.5 second of sample time, prediction horizon with 50 moves, control horizon with 2 steps. ....	59
Figure 26: Graph of the temperature change of PCB heater and NCP21XV103J03RA thermistor in time, based on MPC with 0.2 second of sample time, prediction horizon with 50 moves, control horizon with 2 steps. ....	60
Figure 27: Graph of the temperature change of PCB heater and NCP21XV103J03RA thermistor in time, based on MPC with 0.2 second of sample time, prediction horizon with 40 moves, control horizon with 2 steps. ....	61
Figure 28: Graph of the temperature change of PCB heater and NCP21XV103J03RA thermistor in time, based on MPC with 0.2 second of sample time, prediction horizon with 30 moves, control horizon with 2 steps. ....	62
Figure 29: Graph of the temperature change of PCB heater and NCP21XV103J03RA thermistor in time, based on MPC with 0.2 second of sample time, prediction horizon with 60 moves, control horizon with 2 steps. ....	63
Figure 30: Graph of the temperature change of PCB heater and NCP21XV103J03RA thermistor in time, based on MPC with 0.2 second of sample time, prediction horizon with 70 moves, control horizon with 2 steps. ....	64

## **List of tables**

Table 1: Effects of increasing a PID parameter independently[13] .....	28
Table 2: Examples of signal types used for various applications.....	34
Table 3: Comparison of SBC and CPUs .....	37
Table 4: Comparison of short-range communication technologies.....	41
Table 5: Comparison of long-range communication technologies.....	44
Table 6: Generated values based on heating pad temperature Equation 16 .....	47
Table 7: Measurement results of PCB heater and NCP21XV103J03RA thermistor .....	57

# **1 Introduction**

This MSc thesis is aligned with research activities related to bioanalytical devices based on lab-on-a-chip/droplet microfluidics at Tallinn University of Technology. These include the research project “CogniFlow-Cyte: Cognitronic Lab-on-a-Chip System for Highly-Automated Flow Cytometry” and the development of an isothermal DNA amplification system. The above activities take as a starting point the fact that bacterial threats and infectious diseases are a modern major challenge for society. Fast response to such threats and diseases, by means of in-the-field testing (including point of care (PoC)) can help saving human lives and reduce epidemic risks. In the above research and development activities, the goal is to create a proof of concepts for low-cost portable bioanalytical devices such as a flow cytometer and an isothermal DNA amplification system based on lab-on-a-chip/droplet microfluidics. However, making such low-cost portable bioanalytical devices user-friendly and fully automated requires, among other things, that their various elements be controlled effectively, while the implementation of such control mechanisms be efficient in terms of resources usage to minimize the cost of the overall system. In what follows, we first give a concise description of flow cytometry (FC) and isothermal DNA amplification and related implementation challenges, and we proceed with the problem statement for this MSc thesis.

## **1.1 Concise Description of Flow Cytometry**

Flow cytometry is used for the measurement of characteristics of cell flows by the derivation of dyes or antibodies which may be located either on the surface or inside the cells [1]. To put it in other words, FC is the method for quantifying cells stored in a fluid phase. In fluorescence flow cytometry (FFC), the fluidic chamber maintains that the cells pass a single file through a light source that stimulates fluorochrome. The method helps researchers to compare and quantify particular types of cells within this heterogeneous population, making it possible to study limited numbers of cells or rare subgroups in a mixed population. As of today, there are three varieties of flow cytometers on the market today:

- Fluidic-based FC
- Acoustic-based FC
- Mass cytometers

Given these three cytometer types, fluidic-based FC is most commonly used; to detect protein expression on cells, cells must be labelled with antibodies linked either fluorochromes or, in recent advances, heavy metal tags, regardless of the type of flow cytometer used.

In recent years, parallel to the technologic advancements especially in the field of biology and chemistry regarding to development in FC, the focus of the research on the development of embedded/portable microfluidics measurement devices is gaining traction. Microfluidic FC, in which microfluidics and flow cytometry are combined to realize novel functionalities on a microchip, is a powerful device for measuring several biological sample characteristics. The main idea behind the implementation of an embedded/portable device is to keep it low-cost, modular and compact flow cytometer in order to make this technology more accessible in need of necessarily scientific areas especially in the underserved areas where it is difficult to reach to health care technologies quickly and economically.

Existing FCs are mostly based on desktop computer CPUs, Field Programmable Gate Arrays (FPGA), or even Graphics Processing Units (GPU) for example imaging FCs, to execute the necessary processing and control tasks. Typically, communication is done over Ethernet or USB cables and FC devices are not connected to the internet.

The fluorescence emissions are measured electronically in the form of discrete events; in imaging FCs, the time needed to process the images is a possible bottleneck; this is an essential issue to consider when targeting the embedded and portable implementation of an FC device.

Today, various processing units such as system-on-chip (SoC) with an on-board microcontroller unit (MCU) and digital signal processor (DSP) can potentially implement the required processing and control operations for FC, while being compact, energy-efficient, cheap, and commonly available on the market. Modern SoC and single board

computers (SBC) also provide wireless communication and built-in battery management. That said, there remain various open challenges: the computational power and memory size of the processing unit must be sufficient to properly execute the processing and control operations.

## **1.2 Concise Description of Isothermal Nucleic Acid Amplification (NAAT)**

Note: this section builds upon the introduction to NAAT given in previous work conducted at Tallinn University of Technology [2].

Nucleic acid amplification aims at studying and/or detecting either individual genes or DNA regions of interest. For practical purposes, several copies of a target are generated in order to have a sufficiently large quantity of nucleic acid for study purpose and/or ease the detection process. This can be done by means of in-vitro amplification methods that amplify nucleic acids selectively. If the target to be detected is present in the sample, it will be amplified; no amplification occurs otherwise. Thermal cycles are key to nucleic acid amplification by means of polymerase chain reaction (PCR) in order to create the reactions that depend on the temperature. In such thermal cycles, reactants are subject to repeated heating and cooling cycles. Primers reactants are short, single strand DNA fragments of which the DNA sequence is complementary to that that of the target DNA. On the other hand, synthesis of new DNA involves polymerase (which is an enzyme).

In PCR, DNA melting occurs at high temperature and separates the DNA strands. This is followed by lowering the temperature in such a way that the primers can bind to the complementary sequences of DNA. Finally, a new DNA strand is assembled thanks to the polymerase. The thermal cycles create an exponential amplification, i.e. after being generated, the initial DNA template is used for replication, and the cycles repeat.

Isothermal NAAT methods are based on other reagents than that of PCR. A typical case is the loop-mediated isothermal amplification (LAMP) that makes use of two sets of purpose-made primers and a DNA polymerase, enabling strand displacement. A major advantage of this approach is that there is no need for high temperatures as is the case for DNA melting; moreover, the procedure does not need thermal cycles, i.e. only one temperature is needed.

Isothermal amplification methods have been shown to have comparable performance than PCR in terms of sensitivity, selectivity, and throughput. Thanks to their simplified concept and requirements, such methods are desirable for the design and implementation of low-cost portable devices. Yet, a critical point is precise temperature control over specific time periods. Various isothermal NAAT protocols have different requirements. A few examples: rolling circle amplification should have a temperature ranging between 30–65°C for 60 minutes; exponential amplification reaction should have an exact temperature of 55°C for 10-20 minutes; and cross-priming amplification 65°C for 60 minutes. See [3] for a more complete list of protocols' values. Although the range of temperatures for some of those protocols are given as several °C, the practice often requires a tolerance of  $\pm 1$  °C [3].

A common point with the FC presented in the previous subsection is that the various operations needed for NAAT, in particular the control operations, can be targeted for implementation on modern SoC/SBCs with wireless communication and built-in battery management; likewise for FC they are also subject to the same challenges, including sufficient computational power and memory size.

### **1.3 Problem Statement and Steps of the Thesis**

As mentioned above, the problem that is investigated in this thesis is that there are several challenges when bioanalytical devices such as FC and isothermal DNA amplification should be implemented in portable and embedded (resource-constrained) structures.

Among other things, bioanalytical devices are built around several sub-systems and they must be carefully controlled for the analysis procedures to be successful. Among various existing control methods, Model Predictive Control (MPC) has been used in industrial processes since the 1980s. It is worth noting that over the last years, it has gained increased attention and can be considered for the implementation of a wider range of applications.

A specificity of MPC is that it can optimize the current timeslot and at the same time to take the next ones into account. Its key mechanism is to work with a time-horizon that is finite but to only implement the current timeslot and repeat the optimization. Moreover,

MPC solves an optimization problem; by means of a plant model it can predict the future plant outputs as close as to the desired output reference value.

**The research statement for this thesis is formulated as follows:**

Considering the advantages of MPC, but given the limited research efforts exploiting it in bioanalytical devices, this thesis seeks to investigate the potential of MPC for resource-constrained platforms in view of realizing low-cost portable bioanalytical devices.

In order to reach the expected outcomes, the steps which are given below are followed in this MSc thesis:

- i. Gain some understanding of the main principle of bioanalytical device examples. In the thesis, FC and isothermal DNA amplification are considered.
- ii. Explore and describe various control methods with a focus on MPC.
- iii. Research and compare different candidate SoC/SBC boards that could be considered for implementing the various operations (including control) needed for low-cost portable bioanalytical devices.
- iv. Perform various experiments with MPC in order to explore and understand its behavior depending on the tuning of its parameters.

## **1.4 Organization of thesis**

The remainder of this MSc thesis is organized as follows. Chapter 2 presents the main concepts of flow cytometry, isothermal DNA amplification, and various control methods, including MPC. Next, Chapter 3 discusses design considerations in the form of an overview of candidate boards that could be used as part of the implementation of bioanalytical devices. Subsequently, Chapter 4 describes the experiments and results related to the investigation of MPC's behaviour depending on how its parameters are tuned; as examples, temperature regulation using a heating pad, and a heating PCB component are considered. Finally, the last chapter concludes the thesis and suggests directions for future work.

## **2 Background**

This chapter consists of four main sections based on the research which has been done based on several sources regarding the existing projects on bioanalytical devices, control methods, and control signals. In the first section, the main concepts of FC are presented. In the second section, the main concepts of isothermal DNA amplification are introduced. Control methods, including MPC, are described in the third section, and examples of control signals are given in the fourth section.

### **2.1 Flow Cytometry**

This section is based on [1] and introduces the main concepts of FC.

#### **2.1.1 Main concepts**

The term cytometry means that the physical and chemical properties of cells (i.e. the structural, functional, biological brick of organisms) are analysed. FC refers to an analytical system that generates a certain amount of information from a special liquid about each of the various groups of cells contained together. The objective is to detect cells in the same group among cells in different groups by means of device-generated data. Such an FC device, which Andrew Moldavan developed in 1934 [4] as the first known example, was more like a microscope. With the technology evolution, FC has become prominent today because of its further growth in the past years and its importance and use areas. FC, which was initially used in hematology laboratories, is nowadays used for diagnosis in clinical laboratories such as immunology, oncology, organ transplantation units, pathology, biochemistry, determination of cell subgroups and cell cycle process. Specific information that distinguishes the approximate size of the cells, their granularity (the density of the organelles in the particle or the organelles in the cell) and sub-cell groups can be obtained using the information produced by the FC system [1].

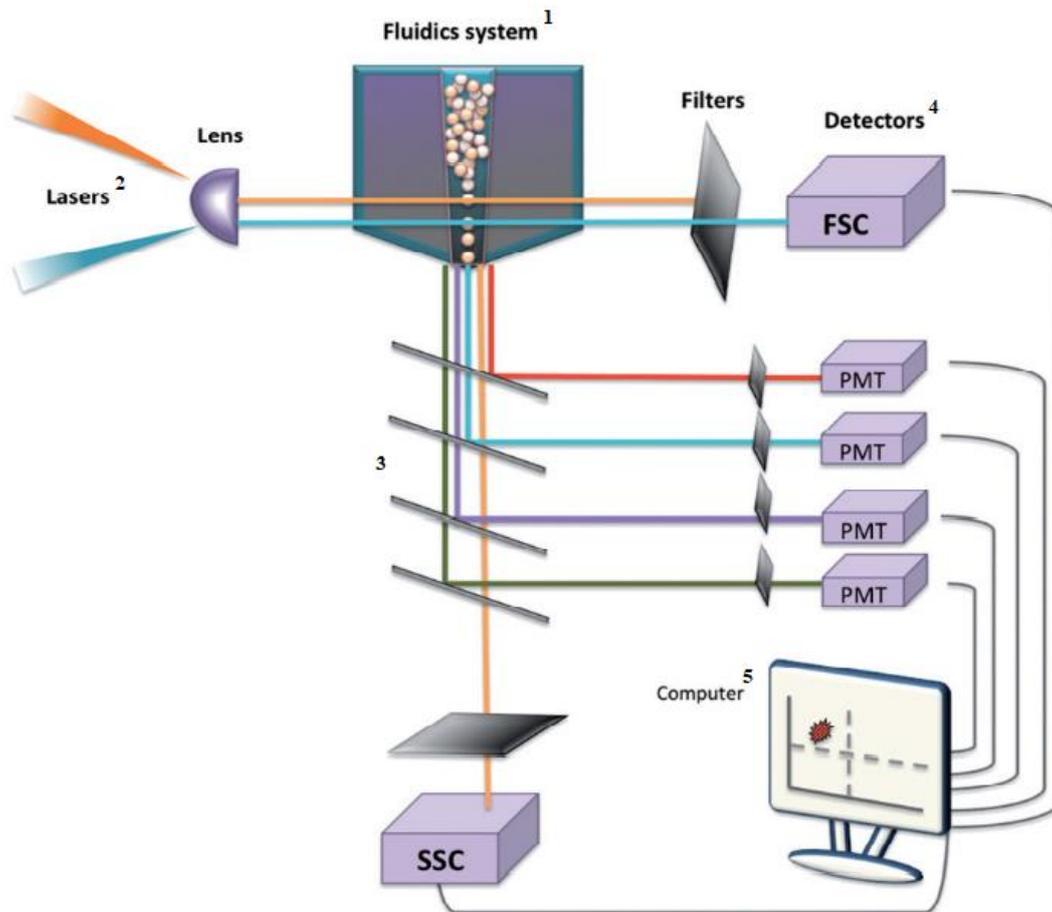


Figure 1: Example of an FC system [1] composed of five major elements 1) the fluidics system, 2) the laser source, 3) the mirror system, 4) the detectors and 5) the computer system.

There are five major elements (denoted with labels 1-5 in Figure 1 in FC: 1) the fluidics system, 2) the laser source, 3) the mirror system, 4) the detectors and 5) the computer system.

The FC method begins with the suspension of the cells in a specific laminar solution fluid (the state of the cells being in the liquid without melting). The cells are grouped into a single row here. Later, one by one, the cells in the liquid go out of the canal and are exposed to the laser beam. At various angles, a cell that has been exposed to the laser beam reflects the light. With the aid of mirrors, the reflected rays are transmitted to the detectors. These rays are transformed by detectors into electrical signals. Finally, these signals are translated by means of the computer system into numerical data. For all cells, these operations are completed. The most significant point in the process is that the light is mirrored at various angles by cells exposed to the laser beam. In order to represent its

specific characteristics, each type of cell reflects light and tries to differentiate cell groups from the data achieved as a result of these reflections [1]

As a result of the process, the properties acquired can be grouped into three specific classes called forward scatter, side scatter and fluorescence characteristics.

#### **2.1.1.1 Forward scatter**

The cell reflects the light on it in 2 directions, i.e. forward and sideways. Scattering that propagates forward in a same position as the direction of arrival of the laser light is called forward scattering. This encompasses the released rays at an angle of around  $20^\circ$ . Forward scattering relates to the cell's approximate size. This function is used to differentiate them, because living and dead cells have different sizes [5].

#### **2.1.1.2 Side scatter**

Side scattering is the scattering at an angle of around  $90^\circ$  caused by the reflection of the light source. Side dispersion contains details about the cell's granularity, that is, its internal structure. Distinctive characteristics unique to cells are both front and side scattering features [5].

#### **2.1.1.3 Fluorescence characteristics**

Acquisition of data starts with the collection of fluorescently tagged antibodies that can provide distinctive information to the desired cells. In other words, by converting them into different colours, fluorescence is characterized as the reflection of rays falling on a surface to rays of various wavelengths. On the other hand, antibodies may be regarded as compounds that bind sufficiently to their properties to the cell surface and are used to differentiate cells. For instance, antibody A attaches to the surface of the X cell but does not adhere to the surface of the Y cell. In this way, an X cell can be considered the cell holding the A antibody. For these fluorescent inhibitors, antibodies are mixed / labelled. Fluorescent agents that represent distinct wavelengths are mixed with multiple forms of antibodies. Thus, though different types of different fluorescent antibodies reflect different colours, the same colour represents a same type of fluorescently labelled antibodies [6].

## 2.2 Isothermal Nucleic Acid Amplification (NAAT)

The fundamental principles of NAAT have been introduced in Chapter 1 of this thesis. As was highlighted, a common NAAT method is the loop-mediated isothermal amplification (LAMP) in which two sets of primers and a DNA polymerase are used, and in which only one temperature is needed. What follows presents an example of isothermal NAAT systems for better understanding of the principle of operation of the aspects related to the electronics and control parts.

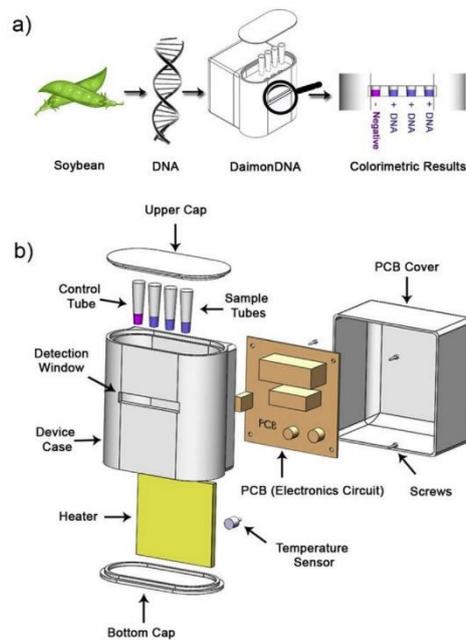


Figure 2: Illustration of the DaimonDNA biosensor. Part a) shows the principle of operation for the realtime monitoring of the biosensor. Part b) shows the various components used for the construction of biosensor, including the electronic PCB and heater [7].

In [7] the authors tackle the issue of the need for portable devices to perform LAMP reactions and present a platform named DaimonDNA. Their system performs LAMP-based detection and visualization using a microfabricated platform. They use off-the-shelf electronic components, as well as physical parts that are 3D-printed. Their application is the detection of soybean species control and genetically modified organisms (GMO). The assembly of their DaimonDNA is shown in Figure 2[7]. It is composed of electronic and physical components, sample tubes that can be removed, a heater based on the Peltier effect, a temperature sensor, a PCB, as well as a detection window for monitoring changes in color of amplicons during the LAMP reaction.

Figure 3 illustrates the workflow of the DaimonDNA biosensor [7]. A microcontroller (Arduino Nano ATmega328) controls the electronics circuit used to switch and amplify the signals to control the Peltier element whose temperature (heat) depends on the voltage applied to. The temperature is measured by a thermocouple inside the DaimonDNA and then converted to the digital domain and passed on to the microcontroller. The system requires the temperature to be carefully controlled so that it remains stable at 65 °C. A detection process takes 30 min and the temperature is controlled by a heat control PID algorithm to provide stable temperature with  $\pm 0.1$  °C accuracy, as shown in Figure 4 [7].

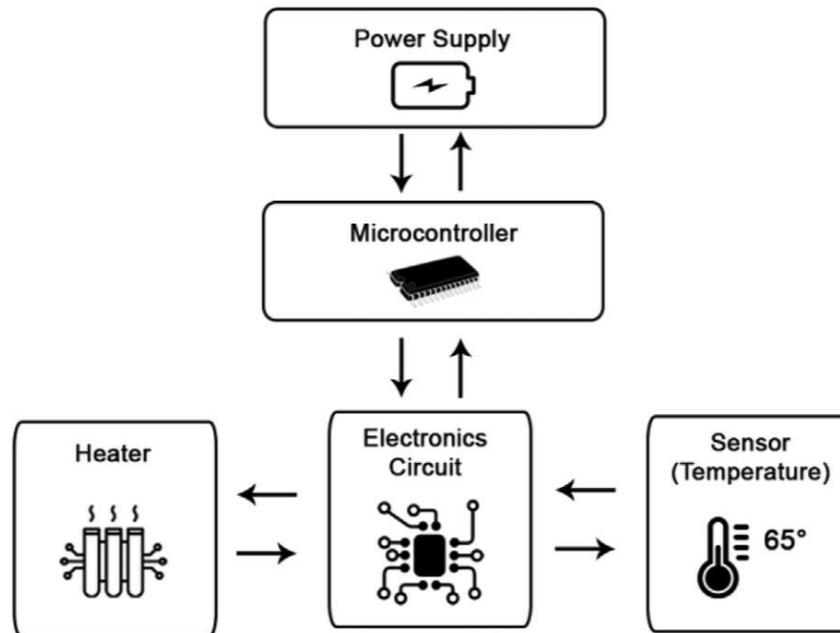


Figure 3: Illustration of the workflow of the DaimonDNA biosensor [7]

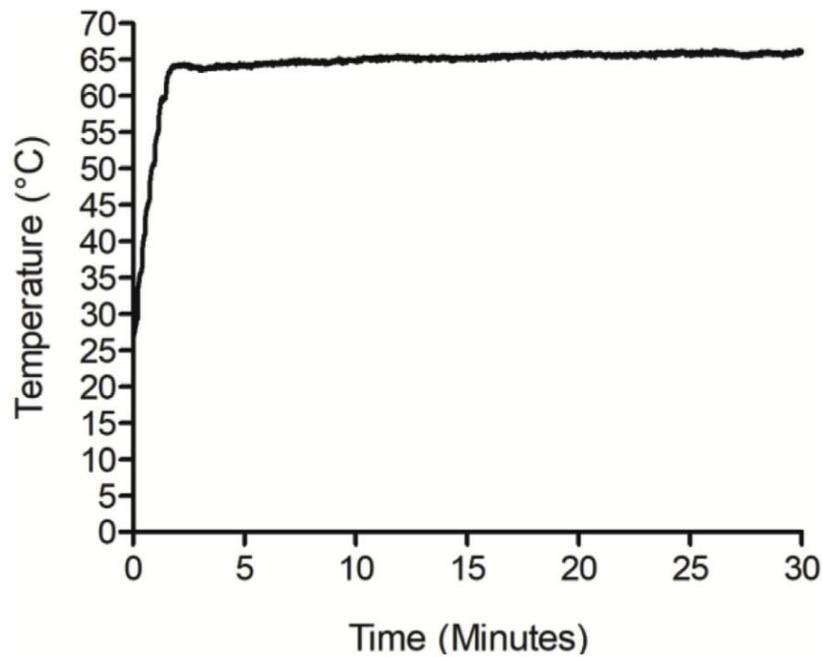


Figure 4: Illustration of the temperature curve of the DaimonDNA biosensor controlled by a PID algorithm. The target temperature is 65°C with an accuracy of  $\pm 0.1$  °C [7].

The introduction chapter of this thesis and the above sections have mentioned the need for control methods in bioanalytical devices. The next section gives an overview of concepts of control methods, including MPC which is to be investigated in this thesis.

### 2.3 Concepts of control

Control is a set of activities, each phase of which contains various cycles, and these interconnected cycles are in stages that are influenced by one another, for the execution and/or monitoring of a task that should be performed. Various internal or outer elements influence the efficiency of any control cycle. These factors can be characterized as control variables[8] When creating a control system, characteristics such as system speed, response time, sensitivity, and/or tolerance are examined, and a decision is made according to these values. When creating a control system, characteristics such as system speed, response time, sensitivity, and/or tolerance are examined, and a decision is made according to these values.

### **2.3.1 Control Algorithms**

Algorithms of control systems can be examined under two main headings: control algorithms for 1) open-loop systems and 2) closed-loop systems. There are fundamental differences between open-loop and closed-loop control algorithms. In the open circuit control system, adjustment is made regarding the controlled parameter, but no measurement, evaluation or correction is made for this adjusted value. In other words, in the event of a possible error in the output, the input variable does not have any feedback based corrective or compensating function. Due to this detrimental situation, open-loop algorithms are generally not preferred in industrial control systems.

On the other hand, in closed-loop control algorithms, continuous measurements of the set value are made to control the desired output value and to remain constant, that is, to compensate for the error in a short time. Closed-loop control algorithms have wide usage areas in industrial studies due to these positive and corrective characteristics. In control/feedback systems, 3 different types of basic controllers are widely used, i.e. proportional (P), integral (I) and derivative (D) [9].

#### **2.3.1.1 On-off control**

The setting variable above or below the set value of the device is completely turned on or off. The variable that is kept under control in this control is in a state of continuous oscillation. It is opened for the desired values and closing for the values above [10].

### **2.3.2 Feedback control**

In a closed-loop control system, the output variable is measured by the measuring element and compared with a reference value after the measurement variable is fed back to the input. As a result of this comparison, the error signal is obtained, and a control signal suitable for the nature of the error signal and the controlled output variable is generated. This closed system is called feedback control system [10]

Examples of feedback control systems include but are not limited to sensor based lighting systems, pumping systems dependent on pressure level, air conditioning systems or watering systems based on the humidity level of plants' soil.

Figure 5 shows a generalized example of a feedback control system.

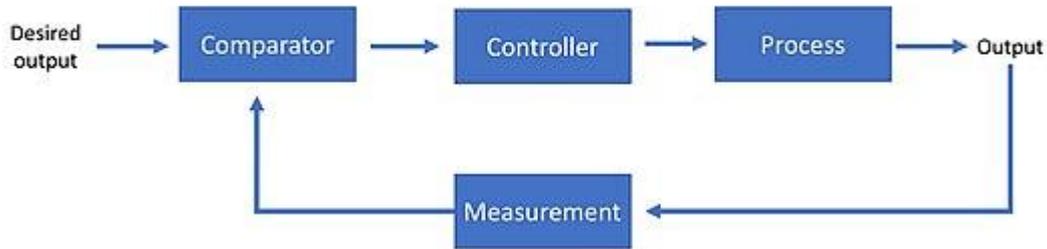


Figure 5: Generalized block diagram of feedback control system.

The drawback of this type of control is that the control step does not start until there is a deviation from the reference value. However, as soon as a deviation is detected, the control device acts to eliminate it. This can cause major damage to some systems that cannot tolerate even the smallest changes, and intervention may be too late. This delay, which will pass from the moment the load variable occurs until the effect of the load variables is detected and a correction command is issued by the controller, is called "dead time". Since this control method works by trial-and-error method, positive or negative departures are seen from the balance position. Feedback control typically has a fluctuating tendency [11].

### 2.3.3 Proportional control

Depending on the error amount and the proportional gain ( $K_p$ ) coefficient, proportional control generates the controller output. It increases the static accuracy and dynamic response of the system. It is a direct function of the error and proportional control coefficient. The controller output is calculated by multiplying the current error value by a gain value.

Proportional control, which shows a lower control effect as the error decreases, ensures that the output approaches the reference closely, so that the oscillation effect that occurs in on/off control is not observed. However, in this case, since the control effect is very small, the effect of the drive system to which the control signal is connected is also very

small and the system output will never reach the reference fully. This difference is also called offset. The offset can be reduced by decreasing the proportional band, but this causes the control system to act as an on / off control after some point [12].

A proportional controller can be expressed as per Equation (1) and is illustrated in Figure 6.

$$P_{out} = K_p e(t) + p_0 \quad (1)$$

where,

$P_{out}$  : Output of the proportional controller

$K_p$  : Proportional gain

$p_0$  : Controller output with zero error.

$e(t)$  : Instantaneous process error at time t.  $e(t) = SP - PV$

$t$  : The present time

$SP$  : Set point

$PV$  : Process variable

A constant controller output is generated for a constant value of the error. The control output does not change if the error persists.

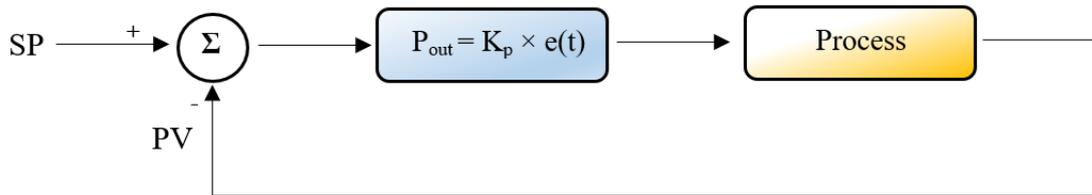


Figure 6: The working principle of a proportional controller

### 2.3.4 Proportional-Integral control

In order to reduce or eliminate the deviation in the above-mentioned proportional control, a control device integrator circuit is introduced. The difference signal between the measured value and the set value is integrated over time. This integral value is added up with the difference value and the proportional band is shifted.

In this way, the energy supplied to the system is automatically increased or decreased, as a result, the process variable is set to the set value [13]

A proportional-integral controller can be expressed as per Equation (2) and is illustrated in Figure 7.

$$P_{out} = K_p e(t) + K_i \int_0^t e(\tau) d\tau \quad (2)$$

where,

$P_{out}$  : Output of the proportional controller

$K_p$  : Proportional gain

$K_i$  : Integral gain

$e(t)$  : Instantaneous process error at time t.  $e(t) = SP - PV$

$t$  : The present time

$\tau$  : Integration variable

$SP$  : Set point

$PV$  : Process variable

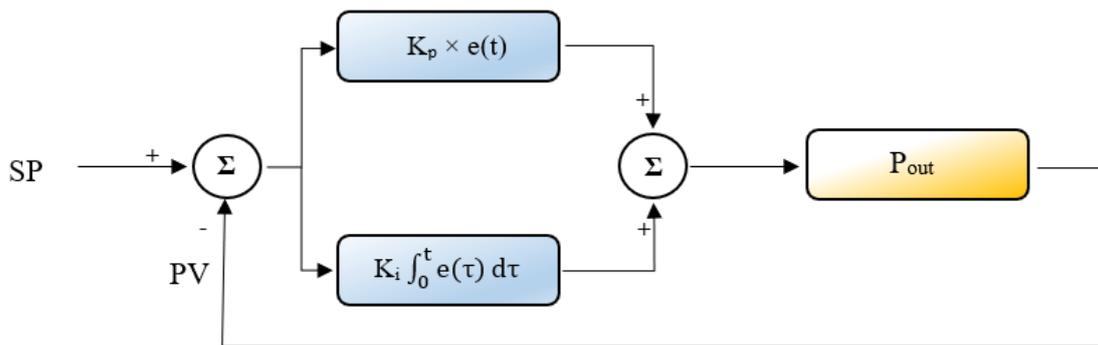


Figure 7: The working principle of a proportional-integral controller

### 2.3.5 Proportional-Integral-Derivative control

Proportional-Integral-Derivative (PID) control systems come into existence in complex processes where Proportional (P) or Proportional Integral (PI) controllers are not sufficient. In proportional control, if the desired value is exceeded or below the desired value with the offset integral algorithm, it is eliminated with the derivative algorithm [13].

The PID control output is the sum of proportional, integral, and derivative terms. If we say that the set variable (MV) to be the sum of these variables, and  $u(t)$  to be the controller output, the final form of the PID algorithm is as in Equation (3) and illustrated in Figure 8.

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau_i) d\tau_i + K_d \frac{de(t)}{dt} \quad (3)$$

where,

- $K_p$  : Proportional gain
- $K_i$  : Integral gain
- $K_d$  : Derivative gain
- $e(t)$  : Instantaneous process error at time t.  $e(t) = SP - PV$
- $t$  : The present time
- $\tau$  : Integration variable
- $SP$  : Set point
- $PV$  : Process variable

Table 1 summarizes the effects of increasing a PID parameter independently.

Table 1: Effects of increasing a PID parameter independently [13]

Parameter	Rise time	Overshoot	Settling time	Steady-state Error	Stability
$K_p$	Decrease	Increase	Minor change	Decrease	Degrade
$K_i$	Decrease	Increase	Increase	Eliminate	Degrade
$K_d$	Minor Change	Decrease	Decrease	No effect in theory	Improve if $K_d$ is small

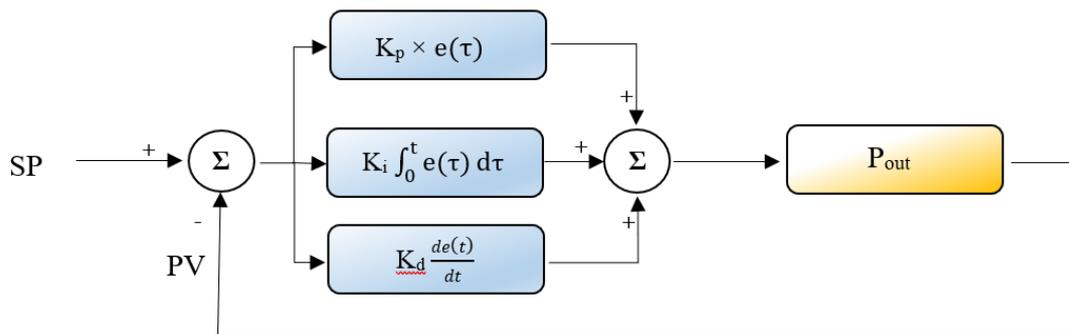


Figure 8: A PID controller block diagram in a feedback loop

### 2.3.6 Model Predictive Control

Model Predictive Control (MPC) uses a variety of control methods, allowing the use of an explicit dynamic plant model to estimate the effect of future outcomes of the manipulated variables on performance and control signals obtained by minimizing the computational cost. The efficiency of the controller depends on how well the device dynamics are captured by the input-output model used for the controller design [14]

Model Predictive Control, MPC, typically involves the following three concepts.

- Explicit use of the model to forecast the performance of the process over the future time horizon.
- Calculation of the control series to maximize the output index.
- An accelerating horizon strategy, so that at each moment the horizon is shifted towards the future, which requires the application of a first control signal of the series at each phase.

Characterization of the Model Predictive Control is given in Figure 9.

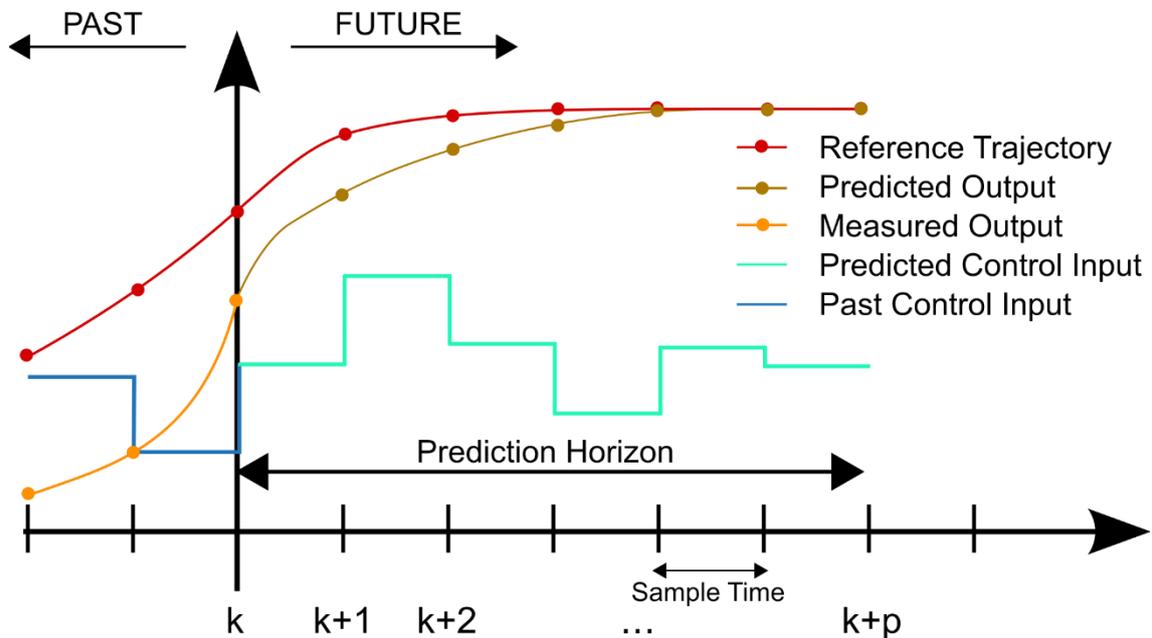


Figure 9: Characterization of the Model Predictive Control [32]

- For the prediction horizon at each sample time, the model computes the predicted future output. These are dependent on the known values at time k (past input and measured output) including the initial condition and the predicted input.
- To optimize the performance criterion, the sequence of future inputs is calculated. The control effort is usually included in the criterion.
- Only the current control signal is sent to the process. All sampling instants are measured, and the data are updated at each sampling instant.

In MPC, process models may be either linear or nonlinear and used to classify trigger mechanisms for system behaviour. The manipulated variables move into the desired direction by minimizing the error or objective function.

There are several different techniques of MPC. What follows presents two commonly used examples, which are Dynamic Matrix Control and Predictive Functional Control [14].

### 2.3.6.1 Dynamic Matrix Control

Dynamic Matrix Control algorithm works by using a step response to simulate input and output. Its ability to control complex high dimensional multivariable systems and handle constraints is what makes its industrial success strong [15].

According to Equation (4), the step response of the system can be described.

$$y(t) = \sum_{i=1}^{+\infty} g_i \Delta u(t - i) \quad (4)$$

where,

$y(t)$  : Model output  
 $g_i$  : Coefficients of the step response  
 $\Delta u$  : Increment of the command

Therefore, the prediction of the output at (t+k) can be given as in Equation (5).

$$\hat{y}(t + k/t) = \sum_{i=1}^{+\infty} g_i \Delta u(t + k - i) + \hat{n}(t + k/t) \quad (5)$$

where,

$\hat{n}(t + k/t)$  : Predicted disturbance in the time t+k

It is expected that disturbance is constant along the prediction as it is given in Equation (6).

$$\hat{n}(t + k/t) = \hat{n}(t + t) = y_m(t) - y(t) \quad (6)$$

where,

$y(t)$  : Model output  
 $y_m(t)$  : Measured output

Following to that, the predicted output can be given as Equation (7).

$$\begin{aligned} \hat{y}(t + k/t) &= \sum_{i=1}^k g_i \Delta u(t + k - i) + \sum_{i=k+1}^{+\infty} g_i \Delta u(t + k - i) + y_m(t) \\ &\quad - \sum_{i=1}^{+\infty} g_i \Delta u(t - i) \\ &= \sum_{i=1}^k g_i \Delta u(t + k - i) + f(t + k) \end{aligned} \quad (7)$$

where,

$f(t + k)$  : Free response of the system

Thus, the free response system is calculated as in Equation (8)

$$f(t + k) = y_m(t) + \sum_{i=1}^{+\infty} (g_{k+i} - g_i) \Delta u(t - i) \quad (8)$$

If the system is asymptotically stable, the  $g_i$  coefficients of the step response will converge to a constant number at the end of  $N$  sampling periods. Therefore, it is possible to consider  $g_{k+i} - g_i \approx 0$  after  $i > N$  and calculation of the free response is calculated as per Equation (9).

$$f(t + k) = y_m(t) + \sum_{i=1}^N (g_{k+i} - g_i) \Delta u(t - i) \quad (9)$$

And if it is written in a matrix form, it is formulated as in Equation (10) and Equation (11)

$$\hat{y} = G \Delta u + f \quad (10)$$

$$G = \begin{bmatrix} g_1 & 0 & 0 & \dots & 0 \\ g_2 & g_1 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ g_m & g_{m-1} & g_{m-2} & \dots & g_1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ g_p & g_{p-1} & g_{p-2} & \dots & g_{p-m+1} \end{bmatrix} \quad (11)$$

As a result, the  $m$ (control horizon size)  $\times$   $p$ (prediction horizon size) sized matrix  $G$  is obtained[15].

The general structure and working principle of dynamic matrix control is given in Figure 10.

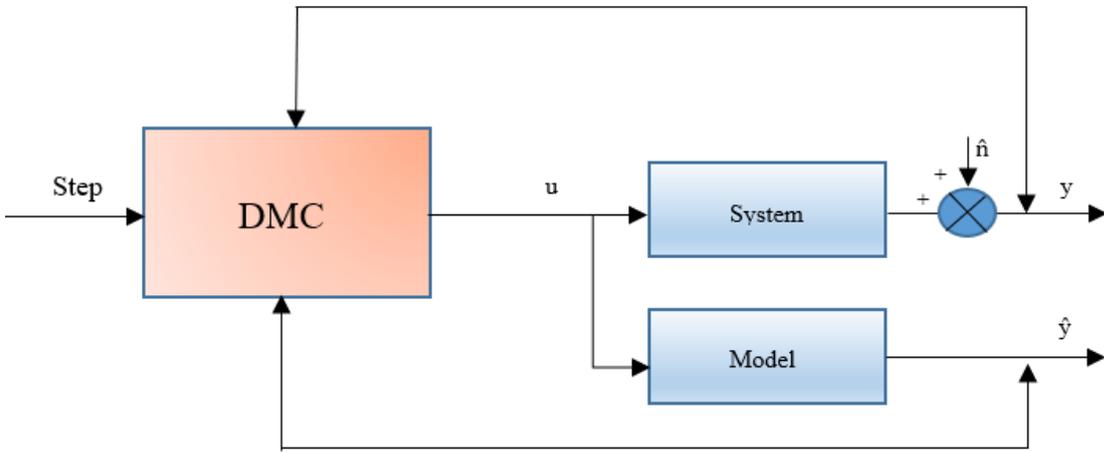


Figure 10: Dynamic Matrix Control working principle scheme

### 2.3.6.2 Predictive Functional Control

The main idea of the predictive functional control is that the controlled variable uses one change or a minimum number of changes possible in the manipulated variable to reach the reference trajectory at the target point. The expected change in the controlled variable during the forecast horizon is computed based on the desired change in the reference trajectory and the expected change in the model output. The manipulated variable can be easily determined from the change in the reference trajectory and the predicted change in the output of the model during the prediction horizon. The expected changes in the controlled variable during the prediction stage can be specified, assuming that the controlled variable matches the reference direction at the target point. This is given in Equation 12 [16].

$$\hat{y}(k + n_p/k) - y(k) = e(k) - \hat{e}(k + n_p/k) \quad (12)$$

Where  $(k) = y_r - y(k)$

$y_r$  : Assumed constant reference signal

$y$  : Controlled variable

$n_p$  : Prediction horizon

$k$  : Actual time

To maintain simplicity, the reference trajectory is selected to be an exponential function. The tracking error then decreases monotonically. This can be explained as in Equation 13.

$$\hat{e}(k + 1/k) = \lambda_r e(k), \dots, \hat{e}(k + n_p/k) = \lambda_r^{n_p} e(k) \quad (13)$$

where,

$\lambda_r$  : Reduction ratio of trajectory's error.

As a result, expected change in the controlled variable can be specified as in Equation 14.

$$\Delta \hat{y}(k + n_p/k) = (1 - \lambda_r^{n_p}) e(k) = (1 - \lambda_r^{n_p}) [y_r - y(k)] \quad (14)$$

which corresponds to the change of the model output that is shown in Equation 15.

$$\Delta \hat{y}_m(k + n_p/k) = \hat{y}_m(k + n_p/k) - y_m(k) \quad (15)$$

The general structure and working principle of predictive functional control is given in Figure 11.

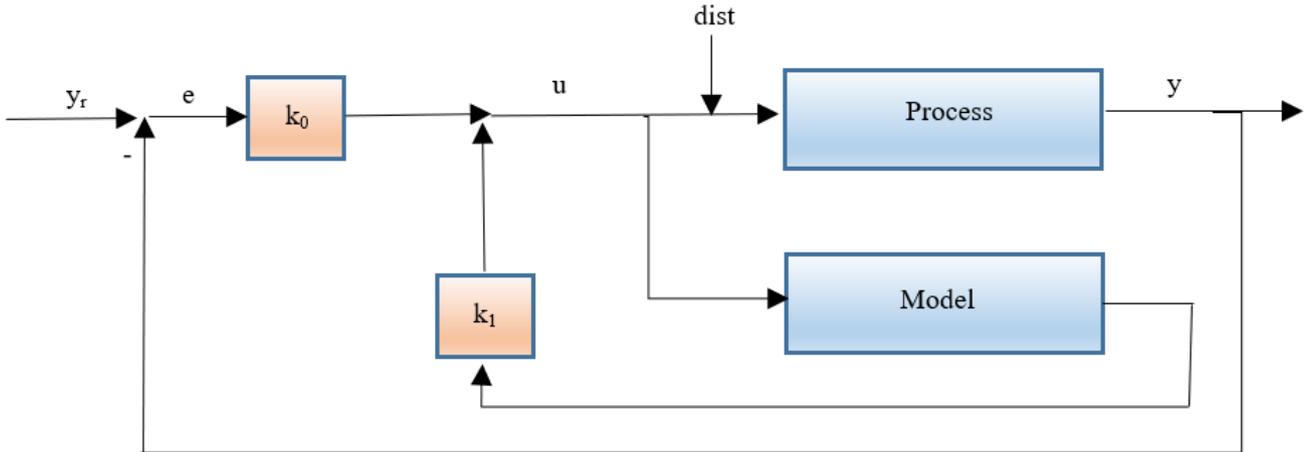


Figure 11: Predictive Functional Control working principle scheme.

### 2.3.7 Control Signals

As per the principle given in Figure 2, to control the system and provide the feedback it is needed to control the input/output signals accordingly. In different examples it is possible to see various signals for different applications that is given in Table 2.

Table 2: Examples of signal types used for various applications.

Signal Type	Signal Direction	Used for	Purpose	Reference
PWM	Input of the pump control system	Pump control	Control the pump control board threshold voltages up to 12V	[18]
PWM	Input of the pump	Peristaltic pump	Controlling the pumping speed	[19]

	control system	p control		
Digital signal	Input/Output	Pump control	Starting data acquisition	
Digital signal	Input/Output	Pump control	Start/stop	
Digital signal	Input/Output	Pump control	Direction	
Analog signal	Input	Sensors	Temperature	
Analog signal	Input	Sensors	Ph	
Analog signal	Input	Sensors	Humidity	
Analog signal	Input	Sensors	Alcohol	
PWM	Input of the servo motor driver	Driver	Controlling the x and y axis of servo motor	[20]

Analog signal	Input	Sensors	Ultrasonic sensor for water	[21]
Analog signal	Input	Sensors	Light Sensor for waste in water	
Analog signal	Input	Sensors	Object distance sensor for waste	
PWM	Motor	Driver	Controlling the motor in the system	
Digital signal	Input	Sensors	Soil moisture sensor input (Through ADC)	
Digital signal	Input	Sensors	Ultrasonic distance sensor (Through ADC)	
Digital signal	Input/Output	Relay	Switching the relay on/off to control the water pump	

## 3 System Design

### 3.1 Hardware, Software and Communication Technologies

#### 3.1.1 SBCs

SBC is a piece of hardware that must be programmed before being used appropriately in a project. A programming tool, which consists of programming software and a programming hardware unit, is used to program it. Integrated development environments (IDEs), which include an editor, compiler, simulator, and programmer, are the most common types of programming software tools [22].

The SBC is the key element of this work as it will be the main responsible of this automation process and the management of control and feedback operations. In order to maintain these operations properly it is important to have an SBC which can process with minimum latency, reliable, low-cost, user friendly and compatible with various devices and technologies. Table 3 gives the results of a market research done for this MSc thesis about the commonly used and available SBCs and their SoC/CPU based on the similar projects as well as purposive to the targeted project.

Table 3: Comparison of SBC and CPUs

Name	Model	SoC/CPU	Memory	Supported OS	Price
Orange Pi	4	6-Core (Dual-core Cortex A72+Quad-core Cortex-A52) ARM 64-bit processor up to 2.0 GHz	Dual 4 GB LPDDR4+ 16 GB EMMC Flash	Android 8.1, Ubuntu 16.04,18.04, Debian 9	EUR 56.55
Banana Pi	R18 AVS	Quad-core Cortex A53 @ 1.15 GHz	1 GB DDR3+8 GB EMMC	Linux	USD 129

ODROID	H2+	Intel Quad-core J4115 (4nm) with 4MiB Cache up to 2.5 GHz (Single thread) or 2.3 GHZ (Multi Thread)	Dual-channel Memory DDR-PC19200	Ubuntu	EUR 167.99
ODROID	N2+	Amlogic S922X Processor (12nm) / Quad-Core ARM Cortex-A73 CPU cluster + Dual Core Cortex-A53 cluster	DDR4 4 GB or 2 GB	N/A	EUR 109.20
ODROID	XU4Q	Samsung Exynos-5422 : Cortex-A15 and Cortex-A7	2 GB LPDDR3 RAM	Ubuntu 16.04, Android 7.1	EUR 70.79
UDOO	X 86II Advanc ed Plus	Quad-core Intel Celeron N3160 2.24 Ghz	4 GB DDR3L Dual Channel	Windows 10, 8.1, 7, Linux, Android	USD 174
Asus	Tinker Board	Rockchip Quad-Core RK3288 processor	2 GB Dual Channel DDR3	TinkerOS, Android, Linux	EUR 60.90
Huawei	HiKey 960	4 Cortex A73 + 4 Cortex A53	3 GB LPDDR4 SDRAM	Linux	USD 250
NVIDIA	Jetson Nano	CPU Quad-core ARM A57 @1.43 GHz	Memory 2 GB 64-bit LPDDR4	N/A	USD 59

Khadas	VIM3 Pro	Amlogic A311D x4 A73 at 2.2GHz + x2 A53 @ 1.8GHz	4 GB LPDDR4		EUR 121.8 1
Raspberry Pi	4B	Broadcom BCM2711, quad-core Cortex-A72 (ARM v8) 64-bit SoC @ 1.5GHz	8 GB LPDDR4 SDRAM	Raspbian	EUR 75

Considering the anticipated computational requirements of the devices considered in the research work to which this thesis relates, the software support, the price, the availability on the market, as well as the already ongoing use of SBCs for research activities at the university, the Raspberry Pi 4B is the preferred candidate board at this point.

### 3.1.2 Software

In this section, software which are used in this Master Thesis will be explained further.

#### 3.1.2.1 MATLAB

Matlab is one of the high-level technical computing languages and interactive environments for developing algorithms, visualizing data, analyzing data, and performing numerical computations. Technical computing problems can be solved faster with Matlab than it can be with traditional programming languages including C, C++, and Fortran [23].

Within this environment two following elements are used in this thesis:

- System Identification Toolbox is a tool that builds mathematical models of complex systems based on measured input-output data. It enables to create and use models of complex systems that are difficult to model using first principles or requirements[24].
- Curve Fitting Application is an application and set of functions for fitting curves and surfaces to data. It allows to conduct exploratory data analysis, pre- and post-process data, compare candidate models, and eliminate outliers. It is possible to perform regression analysis using the given library of linear and nonlinear models, or defining custom equations [25].

### **3.1.2.2 Simulink**

Simulink is a multidomain simulation and model-based design environment built on block diagrams. It helps with embedded system design, simulation, automatic code generation, and continuous testing and verification.

Simulink is a tool for modelling and simulating dynamic systems that includes a graphical editor, customizable block libraries, and solvers. It is compatible and integrated with Matlab [26].

### **3.1.3 Communication Technologies**

Communication technologies are one of the key elements of the embedded systems to maintain the communication regarding the control & feedback process. In this case, node-to-node or cloud-to-node transfer can be made for using the input and output of the system according to the purpose.

Based on the current system, my MSc co-supervisor had already performed a detailed research regarding available communication technologies, including the specification of each technology. In addition to that, advantages and challenges are given together with the specifications. For reference, the results of that study are reported in Table 4 for the comparison of short-range communication technologies, and in Table 5 the comparison of long-range communication technologies is given.

Table 4: Comparison of short-range communication technologies

Communication Technique	Data Rate	Range	Power Consumption	Network Topology	Security	Advantages	Challenges
Zigbee	250,40, 20 kbps	10-75 m	Low	Star or peer-to-peer	128 bit Advanced Encryption Standard (AES)	Fairly good range, low-cost, Low-power consumption, Network Scalability	less Secure, low data rate
Bluetooth/BLE	1 Mbps, 54Mbps (High speed operation)	10 m	Medium/Low	Master-slave, no fixed limit for LE	Enhanced Authentication (Security Mode 1 & 2)	High data rate, low-cost, easy-to-use, low power consumption (BLE)	Short-range, less secure, high interference
ANT+	60 kbps	30 m	Low	peer-to-peer, star, connected star, tree and fixed	8 byte Network Key, 12 8-bit AES	scalability, Ultra-low power, ease-of-use, efficient	Low data-rate

				mesh topologies			
UWB	110Mbps	150m	Low	peer-to-peer	Extended Security Layer	High-resilience to Interference, High data-rate due to concurrent operation, low-cost, high data-rate	Antenna Design
RFID\NFC	Proximity-coupling(106kbps), Vicinity-coupling (27kbps), 424kbps(NFC)	max .5m, <10cm (NFC)	Low/Ultra-Low	Star or peer-to-peer	PIN controlled data access, Device Validation	Low Power consumption, ease-of-use	Low data-rate, Less Secure, short range
WiFi	600-1800Mbps-(WiFi6)	50-150m	medium	ad-hoc	WPA2-PSK	High data-rate, fairly good range,	High Interference, fairly

						easy-to-use	higher power consumption, less secure
WirelessHART	250kbps	100 m	Low	Self-organizing mesh	128-bit AES	Reliable, Secure	Low coverage
6LoWPAN	250,40,20 kbps	10-200 m	Low	Star or simple 2-point routing	AES	low power consumption	short coverage
Z-Wave	9.6,40,100 kbps	max . 100 m	Low	peer-to-peer	128-bit AES, 128-bit zero temporary key, Single Network Wide Key	good resilience to interference	Low data-rate, high-power consumption, expensive
IrDa	2.4kbps - 1Gbps	10-30 m	Low	peer-to-peer	—	High data rate, low power consumption	not scalable, short-range, No

						n, secure	device movement during transmission
DASH7	13,55,200 kbps	1-2 km	Ultra-Low	Tree or peer-to-peer	128-bit AES private key, crypto public key	low latency, Ultra-Low power consumption, secure	

Table 5: Comparison of long-range communication technologies

Communication Technique	Data Rate	Range	Power Consumption	Network Topology	Security Technique	Advantages	Challenges
LoRaWAN	0.3 kbps - 50 kbps	~15 — 20 km	Low	Star	symmetric cryptography (128-bit network session key & 128-bit (AppSKey))	Low power consumption, good resilience to interference	Low data rate
Sigfox	~100 bps	~48 km — (rural — areas)	Low	Star	Advanced Encryption Standard (AES)	High range, good resilience to interference, low latency	Low data rate, less secure

NB-IoT	50 kbps	<15 km	Low	star	TLS or BEST, L2TP or IPsec, MPLS	low power consumption, good range	Low-data rate, latency, poor mobility
LTE-CAT-M1	200-400 kbps	<11 m	Low	star	TLS or BEST, L2TP or IPsec, MPLS	low power consumption, low-cost, good coverage	latency, low data rate
Wi-SUN	300 kbps	~4 km	Low	mesh	AES	High Range, low latency	Low Data Rate

As indicated earlier in this section, the Raspberry Pi 4B is a preferred candidate SBC board from the computational power perspective. Considering that it also features 10/100/1000 Mbit/s Ethernet, b/g/n/ac dual band 2.4/5 GHz WiFi, and Bluetooth 5.0, it is also deemed suitable from the connectivity perspective, at least for prototyping activities. Although this thesis does not aim at completing a prototype, the choice of this SBC is deemed suitable for the future work that will be conducted by the research group after this MSc thesis work ends.

This chapter has presented some design considerations, in particular an overview of the hardware, software, and connectivity aspects. The various criteria led to choose the Raspberry Pi 4B SBC for future possible prototyping activities.

## 4 Experimental Results and Discussion

This section presents the experiments conducted in this MSc thesis, their results, and the discussion thereof. As indicated earlier, the overall purpose is to investigate the potential of MPC and gain a better understanding of its behavior when tuning its parameters. Note that at this stage, the experiments are conducted on a PC with Matlab tools.

### 4.1 Experiments for a heating pad

In order to model the behavior of the system, controlled via MPC the Matlab System Identification Toolbox was used. This toolbox can help to identify the mathematical model based on inputs and outputs of the system as an initial step, the behaviour of heating pad was modelled using the data provided in the datasheet [27], based on given values as  $\text{voltage\_in} = [5 \ 7.4 \ 9 \ 12]$  and  $\text{temperature\_out} = [40 \ 75 \ 91 \ 110]$ . However, as the data regarding the heating pad in the datasheet is limited, it was needed to use the Matlab Curve Fitting feature to vary and provide the function and proper values to input and output arrays. The obtained curve in Matlab curve fitting is given in the Figure 13. Eventually, temperature ( $T$ ) values as a function of the voltage ( $V$ ) is formed as shown in Equation 16 and the values are given in Table 6.

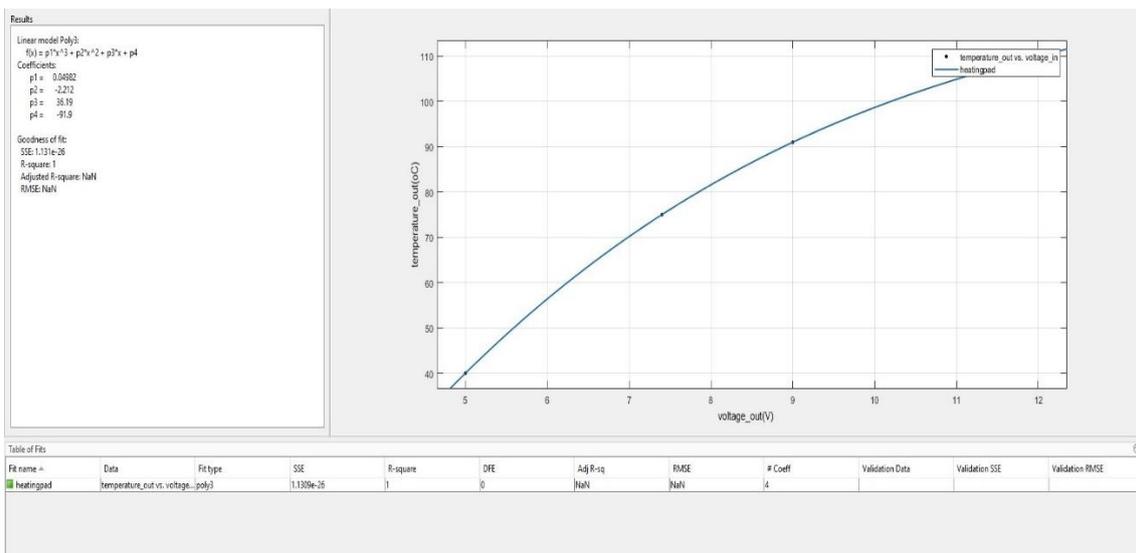


Figure 13: Obtained curve for heating pad in Matlab Curve Fitting

$$f(T) = 0,04982 \times V^3 - 2,112 \times V^2 + 36,19 \times V - 91,9 \quad (16)$$

Table 6: Generated values based on heating pad temperature Equation 16

Voltage_in [V]	Temp_out [°C]
5	39.9775
6	56.36912
7	70.13026
8	81.55984
9	90.95678
10	98.62
11	104.8484
12	109.941
13	114.1965
14	117.9141
15	121.3925
16	124.9307
17	128.8277
18	133.3822
19	138.8934
20	145.66

Voltage\_in is provided as the input for the system and temp\_out is provided as the output of the system in Matlab. After providing the input and output, the Matlab System Identification Tool is used in order to obtain the transfer function, as illustrated below with the copies of Matlab console messages:

```
>> voltage_in=[5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20];
>> temp_out = [39.9775 56.36912 70.13026 81.55984 90.95678 98.62 104.84842
109.94096 114.19654 117.91408 121.3925 124.93072 128.82766 133.38224 138.89338
145.66] ;
```

*tf1* =

```
From input "u1" to output "y1":
  143.4 (+/- 1777) s + 13.71 (+/- 193.1)
-----
s^2 + 5.942 (+/- 53.02) s + 3.429 (+/- 45.14)
```

Accordingly, the transfer function is provided to Matlab as given below.

```
>> sysvt=tf([143.4 13.71],[1 5.942 3.429]);
```

Following that, the main structure of the MPC object is being defined in the system that is given below;

```
>> mpcobjvt=mpc(sysvt,Ts,p,m);
```

where;

T<sub>s</sub>:            Sample Time  
 p:                Prediction Horizon  
 m:                Control Horizon

Initially, as the reference temperature is given as 23 °C, the constraints are set from minimum 22°C to maximum 23°C:

```
>> mpcobjvt.MV=struct('Min',22,'Max',23);
```

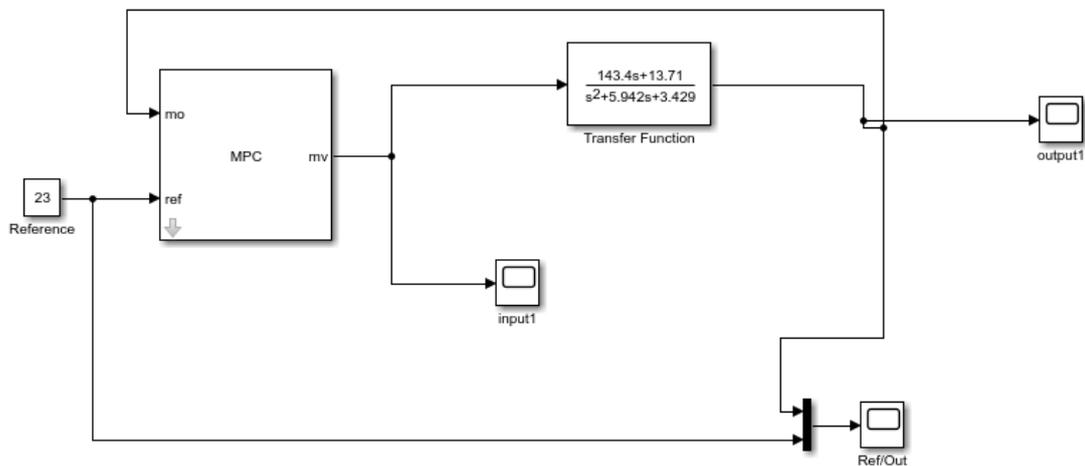


Figure 14:Single input single out MPC Algorithm in Matlab Simulink

The System is designed in Matlab Simulink, using the MPC algorithm as single input, single output system. This structure is given as in Figure 14.

It is demonstrated that in the sampled data setting, a proper design of the running costs can result in improved performance bounds and, as a result, stability for significantly shorter prediction horizons [28].

In order to perform tuning of the MPC object, features in the main structure should be tackled based on the experimental approach. These experiments are suggested to be formed around the principles, separately for each feature [29].

Firstly, principles regarding the decision of control horizon( $m$ ) are given as below:

- The control horizon is the number of MV moves that must be optimized at each control interval., or  $m$ .
- The control horizon is situated between 1 and  $p$ , which is the prediction horizon.
- Default value of  $m$  is equal to 2.

Secondly, principles regarding the decision of prediction horizon( $p$ ) are given as below:

- It is assumed that the control interval is equal to  $k$ .
- When optimizing its MVs at control interval  $k$ , the prediction horizon,  $p$  is the number of possible control intervals that the MPC controller should calculate by using prediction.

In addition to that, it is suggested to increase  $p$  until further increment affect performance in a small way[29].

Finally, principles regarding the decision of sample time ( $T_s$ ) are given as below:

- Qualitatively, by the decrease of  $T_s$ , unknown disturbance is normally improved and then reaches the plateau phase. The value of  $T_s$  at which they reach plateaus are based on the dynamic features of the plant. But the computing effort increases significantly as  $T_s$  decreases. Therefore, a combination of efficiency and computing effort is the best option.
- The horizon of prediction,  $p$ , is also a crucial factor in MPC. While the prediction time is constant,  $p$  should differentiate inversely with  $T_s$ . Many of the array sizes are  $p$  proportional. As  $p$  increases, the memory needs of the controller and the QP, which is an optimization problem with the aim of minimizing or maximising a quadratic function under a series of linear equality and inequality constraints[30], solution time are increased.

Eventually, the MPC object is initially designed with 1 second sample time, prediction horizon as 50 moves, and the control horizon is 2 steps.

#### **4.1.1 Tuning the sample time for the heating pad**

The Matlab command for this process is the following:

```
>> mpcobjvt=mpc(sysvt,1,50,2);
```

The obtained plot of the change of temperature (°C) in time (seconds) based on MPC with 1 second of sample time, prediction horizon with 50 moves, control horizon with 2 steps is given in Figure 15.

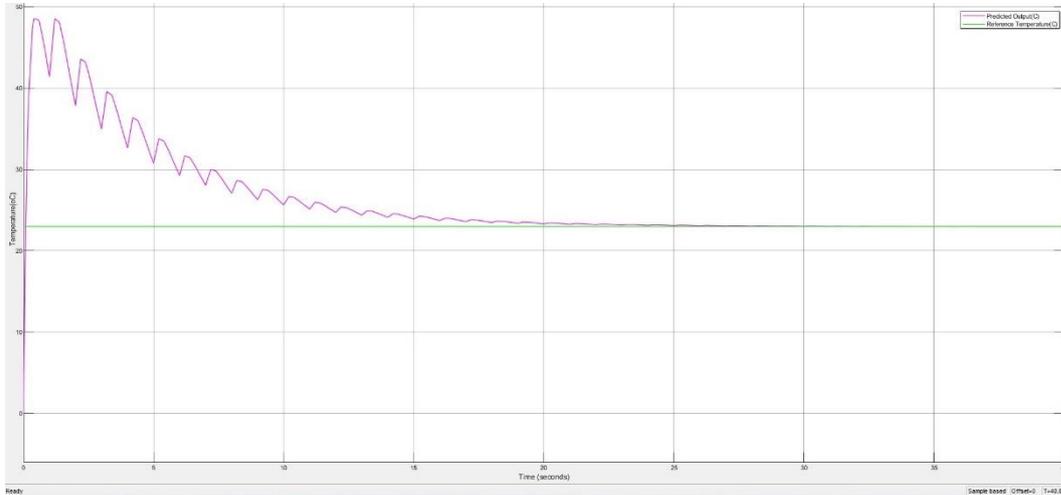


Figure 15: Graph of change of temperature in time, based on MPC with 1 second of sample time, prediction horizon with 50 moves, control horizon with 2 steps.

In Figure 15, significant (damped) oscillations are observed while the temperature is converging to the reference temperature. Based on the principle and observation, it is recommended to decrease the sampling time to reduce these variations and improve the control of the temperature curve. Therefore, in the following experiment, the sampling time is set to 0.8 seconds to refine the results.

The Matlab command to do so is the following:

```
>> mpcobjvt=mpc(sysvt,0.8,50,2);
```

The new obtained graph of the change of temperature ( $^{\circ}\text{C}$ ) in time (seconds) based on MPC with 0.8 second of sample time, prediction horizon with 50 moves, control horizon with 2 steps is given in Figure 16.

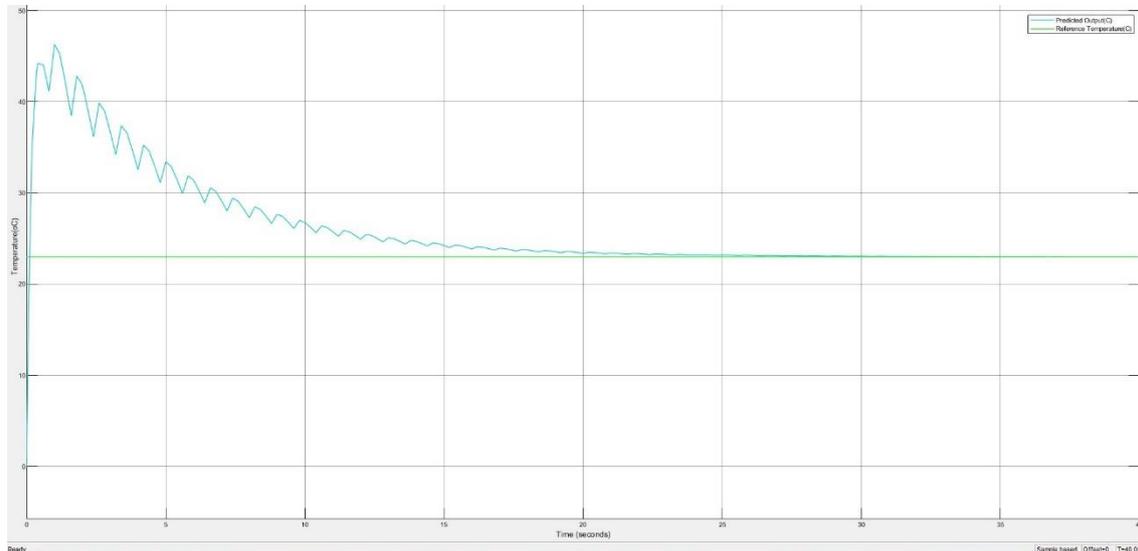


Figure 16: Graph of change of temperature in time, based on MPC with 0.8 second of sample time, prediction horizon with 50 moves, control horizon with 2 steps.

Although it is observed that the curve is slightly smoother, the oscillations are still too large. Therefore, it is recommended to reduce the sampling time in line with the principles.

To do so, the Matlab command is the following:

```
>> mpcobjvt=mpc(sysvt,0.5,50,2);
```

Obtained graph of the change of temperature in time based on MPC with 0.5 second of sample time, prediction horizon with 50 moves, control horizon with 2 steps is given in Figure 17.

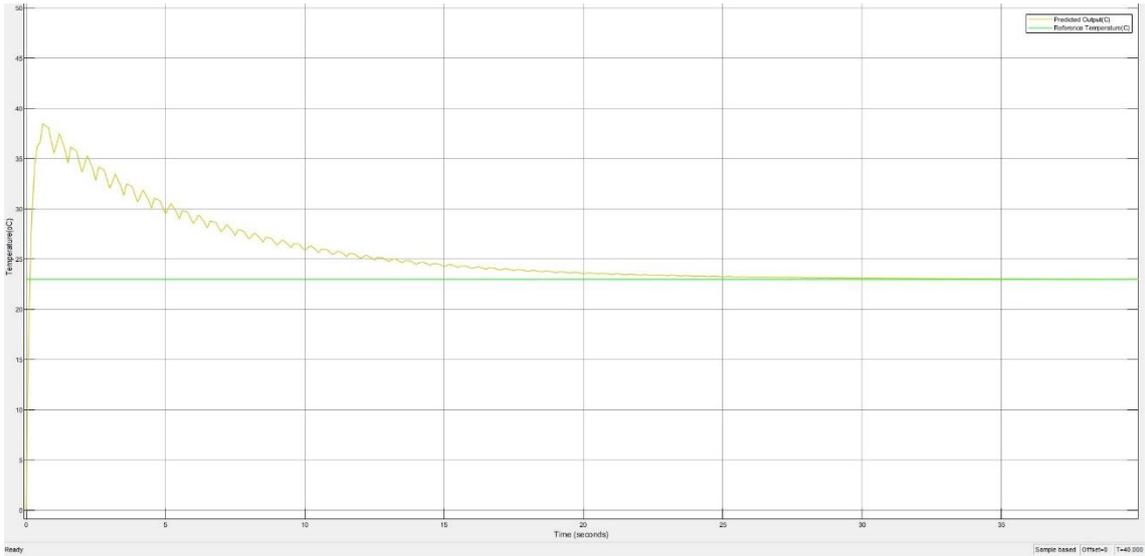


Figure 17: Graph of change of temperature in time, based on MPC with 0.5 second of sample time, prediction horizon with 50 moves, control horizon with 2 steps.

As per comparison to the previous graphs, a smoother curve is obtained by reducing the sampling time to 0.5 second. This indicates that reducing the sampling time may lead to even better results. Hence, the Matlab command is configured accordingly:

```
>> mpcobjvt=mpc(sysvt,0.2,50,2);
```

The obtained graph of the change of temperature in time based on MPC with 0.2 second of sample time, prediction horizon with 50 moves, control horizon with 2 steps is given in Figure 18.

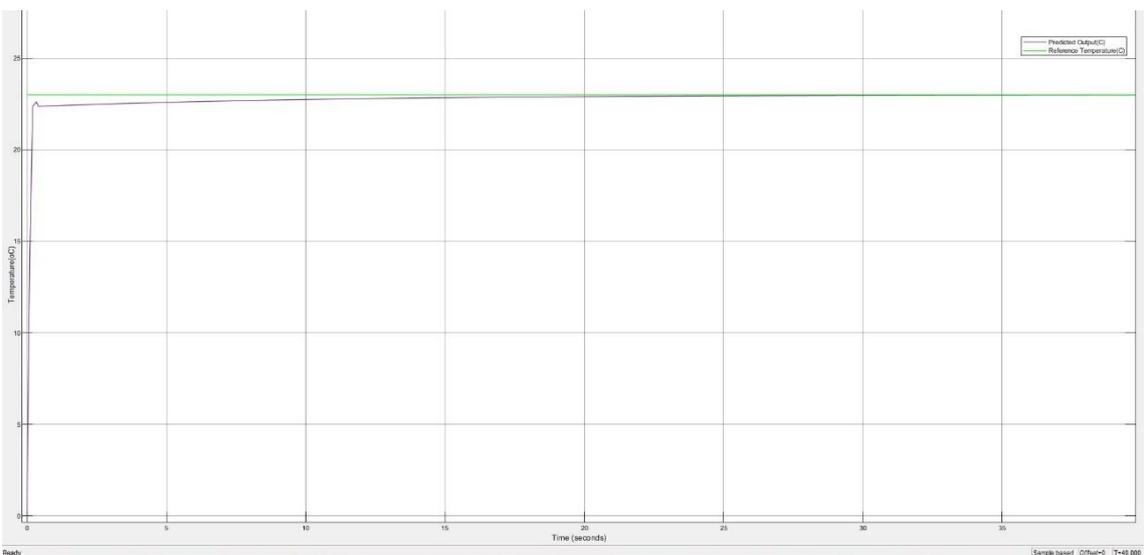


Figure 18: Graph of change of temperature in time, based on MPC with 0.2 second of sample time, prediction horizon with 50 moves, control horizon with 2 steps.

Finally, looking at Figure 18, it is obvious that the results are improved in a tangible level and the shape of the curve is much more in line with our expectations. For this reason, 0.2 second is the sampling time that is considered to use in the system and next experiments.

#### 4.1.2 Tuning the prediction horizon for the heating pad

Following the tuning of the sampling rate, it is suggested to tune the prediction horizon in order to further improve the results. As per the principles and the suggestions regarding prediction horizon tuning [29],  $p$  is initially set to 30 moves and it is recommended to increase the prediction horizon until the increase have a negligible impact on the results in system.

Accordingly, Matlab command to perform these changes is configured as follows:

```
>> mpcobjvt=mpc(sysvt,0.2,30,2);
```

The obtained graph of the change of temperature ( $^{\circ}\text{C}$ ) in time (seconds) based on MPC with 0.2 second of sample time, prediction horizon with 30 moves, control horizon with 2 steps, is shown in Figure 19.

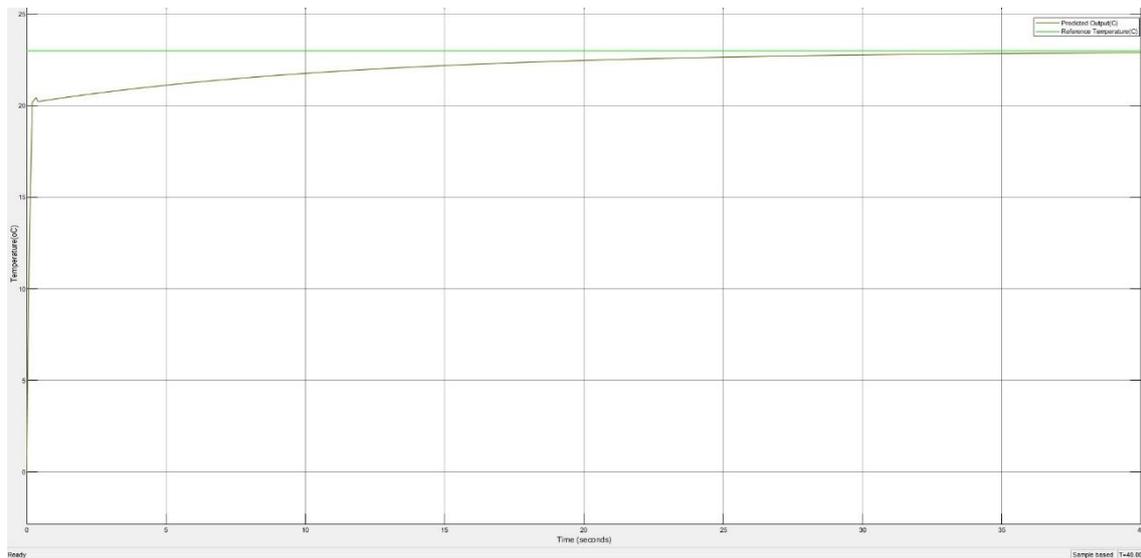


Figure 19: Graph of change of temperature in time, based on MPC with 0.2 second of sample time, prediction horizon with 30 moves, control horizon with 2 steps.

For the purpose of performance improvement, experiments are followed by increasing the prediction horizon to 40 moves, control horizon with 2 steps and the results is observed. The Matlab command to perform these changes is configured as follows:

```
>> mpcobjvt=mpc(sysvt,0.2,40,2);
```

The obtained graph of the change of temperature ( $^{\circ}\text{C}$ ) in time (seconds) based on MPC with 0.2 second of sample time, prediction horizon with 40 moves, control horizon with 2 steps is given in Figure 20.

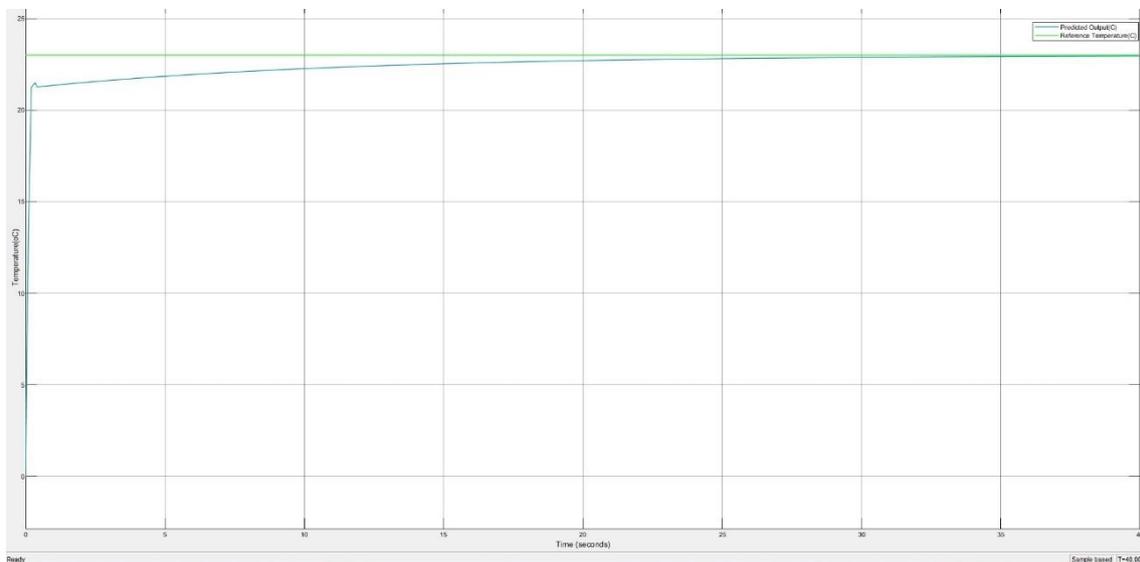


Figure 20: Graph of change of temperature in time, based on MPC with 0.2 second of sample time, prediction horizon with 40 moves, control horizon with 2 steps.

By the results that are shown in Figure 20, the system has relatively better results compared to the experiments based on a prediction horizon of 30 moves; however, the results are not better than for the experiments which are based on the prediction horizon of 50 moves shown in Figure 18.

Considering better computational performance and lower cost for the system, it is recommended to perform an experiment based on the prediction horizon of 60 moves. The Matlab command to perform this change is configured as follows:

```
>> mpcobjvt=mpc(sysvt,0.2,60,2);
```

The obtained graph of the change of temperature in time based on MPC with 0.2 second of sample time, prediction horizon with 30 moves, control horizon with 2 steps is given in Figure 21.

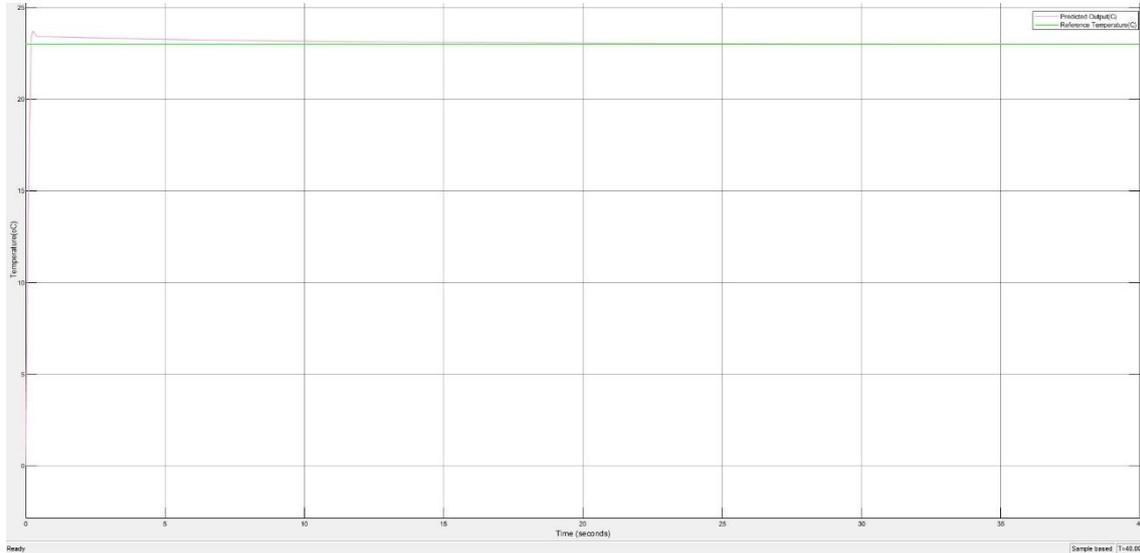


Figure 21: Graph of change of temperature in time, based on MPC with 0.2 second of sample time, prediction horizon with 60 moves, control horizon with 2 steps.

It is visible in Figure 21 that the temperature exceeds the reference temperature and then it converges back to the reference temperature. Accordingly, we can observe that the system behaviour uses computational resources in order to reach to the reference temperature, as well as the performance will be reduced due to this reason. As a result, it is possible to expect that further increase in the prediction horizon will not impact the system positively.

## 4.2 Experiments with a PCB heater and thermistor

Following to the set of experiments based on the datasheet of the heating pad in the previous section, experiments are now performed based on a PCB heater provided by the research team to which this MSc thesis is related and NCP21XV103J03RA PCB heater and thermistor [31]. Image of the PCB heater fitted with the NCP21XV103J03RA thermistor is given in Figure 22.

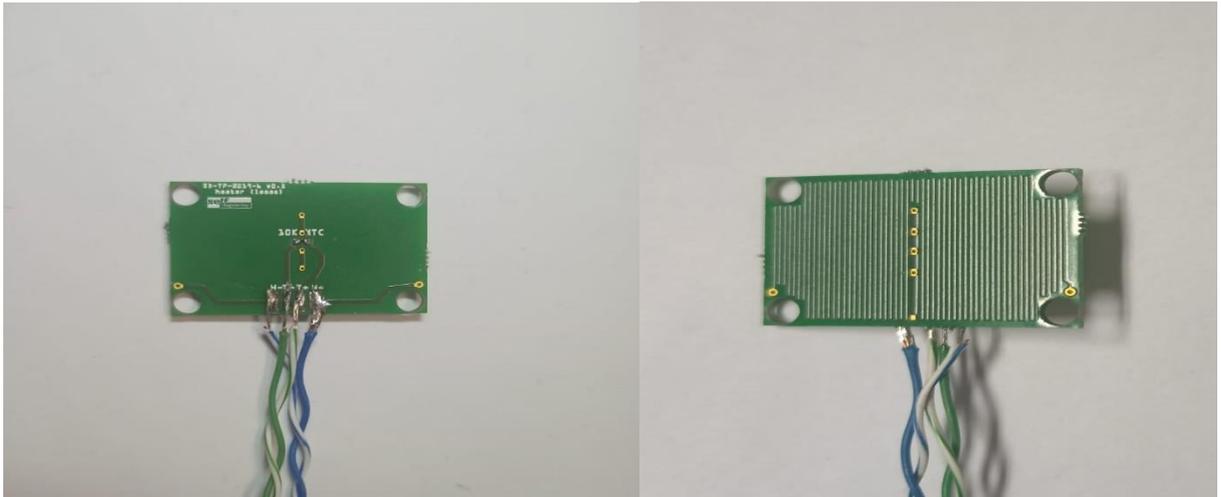


Figure 22: Image of PCB heater and NCP21XV103J03RA thermistor

A precision power supply is used to provide a voltage to the PCB heater and a digital multimeter is used to measure the resistance on the thermistor. Measurements are done with an initial input voltage of 300 mV and increasing by 50 mV. For each input voltage value, the value of the thermistor is measured, until the voltage reaches 2150 mV where the projected output value of the thermistor is reached (corresponding to the temperature 75 C°[31]). The experimental setup is shown in Figure 23.

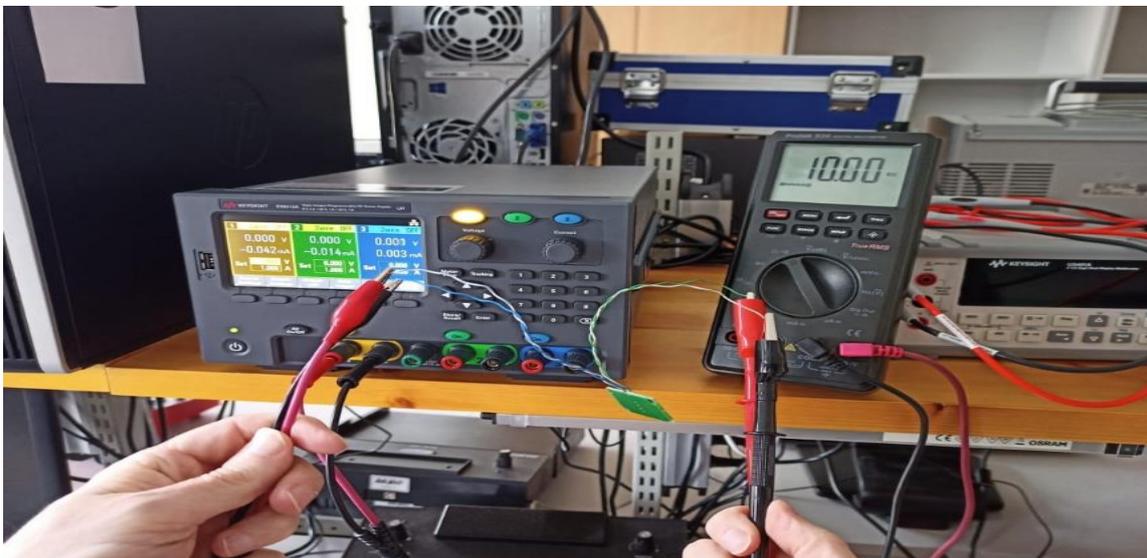


Figure 23: Measuring the output resistance of NCP21XV103J03RA fitted on the PCB heater

The measurement results are processed by using the Matlab Curve Fitting function, to form the temperature function that is based on resistance. The obtained curve in Matlab curve fitting is given in the Figure 24. As a result, the function is formed as in Equation 17, and the values are given in Table 7.

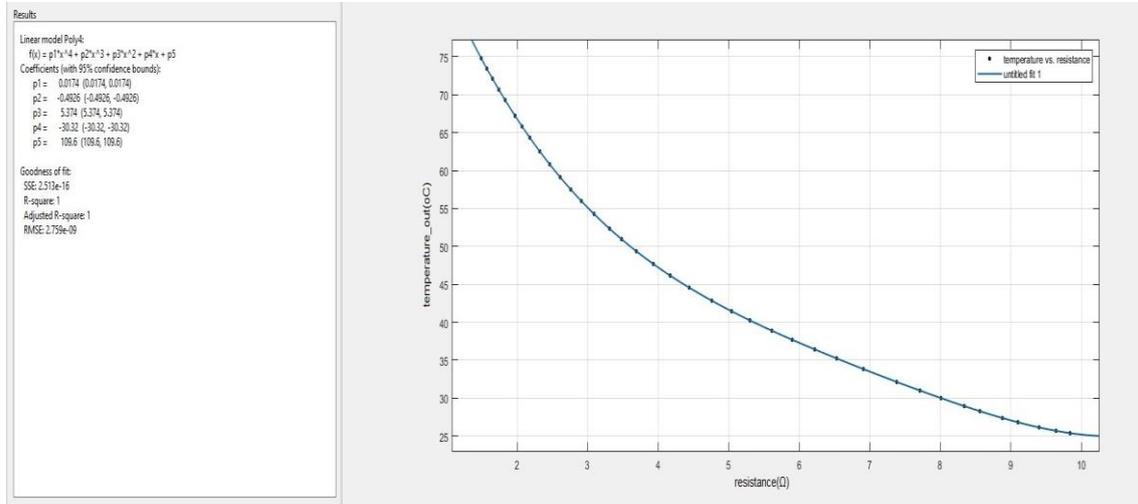


Figure 24: Obtained curve of temperature (°C) as a function of resistance (Ω) for NCP21XV103J03RA in Matlab Curve Fitting

$$f(T) = 0,0174 \times R^4 - 0,4926 \times R^3 + 5,374 \times R^2 - 30,32 \times R + 109,6 \quad (17)$$

Table 7: Measurement results of PCB heater and NCP21XV103J03RA thermistor

Voltage (V)	Resistance (Ω)	Temperature (°C)
0.3	9.84	25.38889988
0.35	9.64	25.69226802
0.4	9.4	26.14328064
0.45	9.1	26.82029754
0.5	8.88	27.38357528
0.55	8.56	28.28451727
0.6	8.34	28.94971376
0.65	8.01	30.00242338
0.7	7.71	31.00456941
0.75	7.38	32.1457125
0.8	6.91	33.82888558
0.85	6.53	35.23803821
0.9	6.22	36.4250732
0.95	5.9	37.69545274
1.00	5.61	38.89773722

1.05	5.3	40.25228674
1.1	5.04	41.45794279
1.15	4.76	42.84432868
1.2	4.44	44.56569728
1.25	4.17	46.15558365
1.3	3.93	47.69391457
1.35	3.69	49.36814644
1.4	3.48	50.95937573
1.45	3.31	52.34352323
1.5	3.09	54.27549387
1.55	2.91	55.98536765
1.6	2.76	57.5067605
1.65	2.61	59.12224569
1.7	2.46	60.83801413
1.75	2.32	62.53552064
1.8	2.18	64.33133164
1.85	2.07	65.8148876
1.9	1.97	67.22151396
1.95	1.83	69.28763919
2.00	1.74	70.67798847
2.05	1.65	72.11886313
2.1	1.57	73.44338105
2.15	1.49	74.81028361

Finally, by using the approach as for the heating pad in the previous section, the following Matlab commands are used in order to obtain the results for the PCB heater and thermistor. Voltage values which are set manually through the power supply and temperature value based on the calculations through the resistance output of the component is provided to the system as follows:

```
>> voltage=[0.3 0.35 0.4 0.45 0.5 0.55 0.6 0.65 0.7 0.75 0.8 0.85 0.9 0.95 1 1.05 1.1 1.15
1.2 1.25 1.3 1.35 1.4 1.45 1.5 1.55 1.6 1.65 1.7 1.75 1.8 1.85 1.9 1.95 2 2.05 2.1 2.15];
>> temperature = [25.38889988 25.69226802 26.14328064 26.82029754 27.38357528
28.28451727 28.94971376 30.00242338 31.00456941 32.1457125 33.82888558
35.23803821 36.4250732 37.69545274 38.89773722 40.25228674 41.45794279
42.84432868 44.56569728 46.15558365 47.69391457 49.36814644 50.95937573
52.34352323 54.27549387 55.98536765 57.5067605 59.12224569 60.83801413
62.53552064 64.33133164 65.8148876 67.22151396 69.28763919 70.67798847
72.11886313 73.44338105 74.81028361];
```

Afterwards, based on the input voltage and output temperature, Matlab System Identification Tool is used to obtain the transfer function of the system:

```
>> sysVT=tf([178.3 16.76],[1 3.145 0.5206]);
```

#### 4.2.1 Tuning the sample time for the PCB heater and thermistor

Although it is observed that MPC provides better results with shorter sampling time (which leads to high computational cost), it is recommended to experiment and obtain the results with certain values from 0.5 to 0.2 seconds to analyze if system could provide tolerable performance at higher sampling time (lower computational cost). In addition to that, the prediction horizon is considered again until the favorable performance is observed in the graph. Therefore, initially the sample time is set to 0.5s, prediction horizon is set to 50 moves and control horizon is set to 2 steps:

```
>> mpcobjvt=mpc(sysvt,0.5,50,2);
```

The obtained graph of the temperature (°C) change of PCB heater and NCP21XV103J03RA thermistor in time (seconds) based on MPC with 0.5 second of sample time, prediction horizon with 50 moves, control horizon with 2 steps. This graph is given in Figure 25.

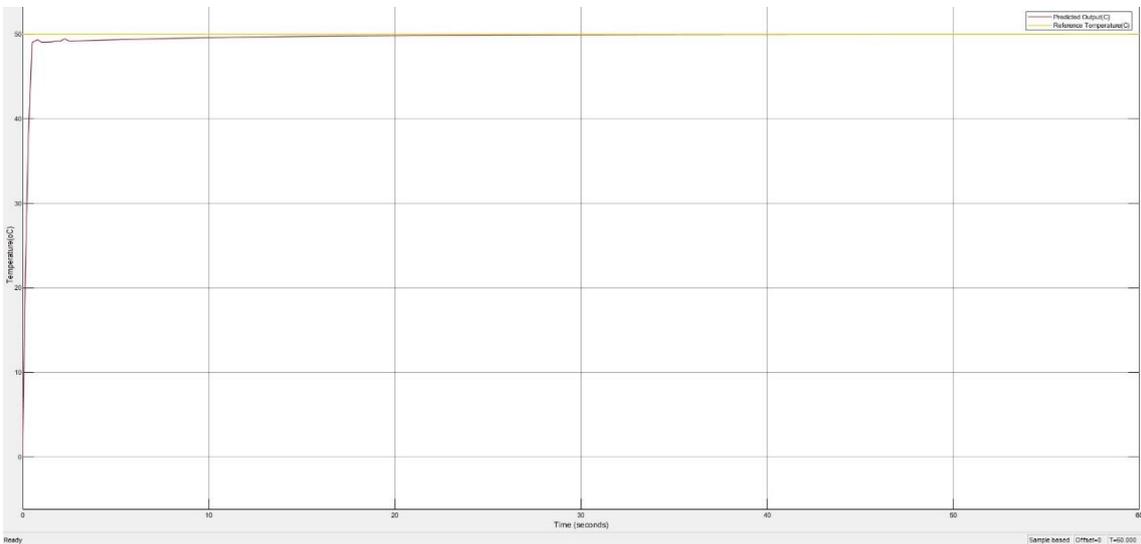


Figure 25: Graph of the temperature change of the PCB heater and NCP21XV103J03RA thermistor in time, based on MPC with 0.5 second of sample time, prediction horizon with 50 moves, control horizon with 2 steps.

Looking at Figure 25, there are minor oscillations observed with the 0.5 second of sample time which proved the previous point that system will respond better at small sampling

time. Therefore, to eliminate these oscillations, the sample time was reduced. To perform this experiment, the sample time is set to 0.2 second and the prediction horizon and control horizon parameters remains the same. This is provided into the system as follows:

```
>> mpcobjvt=mpc(sysvt,0.2,50,2);
```

The obtained graph of the temperature (°C) change of PCB heater and NCP21XV103J03RA thermistor in time (seconds) based on MPC with 0.2 second of sample time, prediction horizon with 50 moves, control horizon with 2 steps is given in Figure 26.

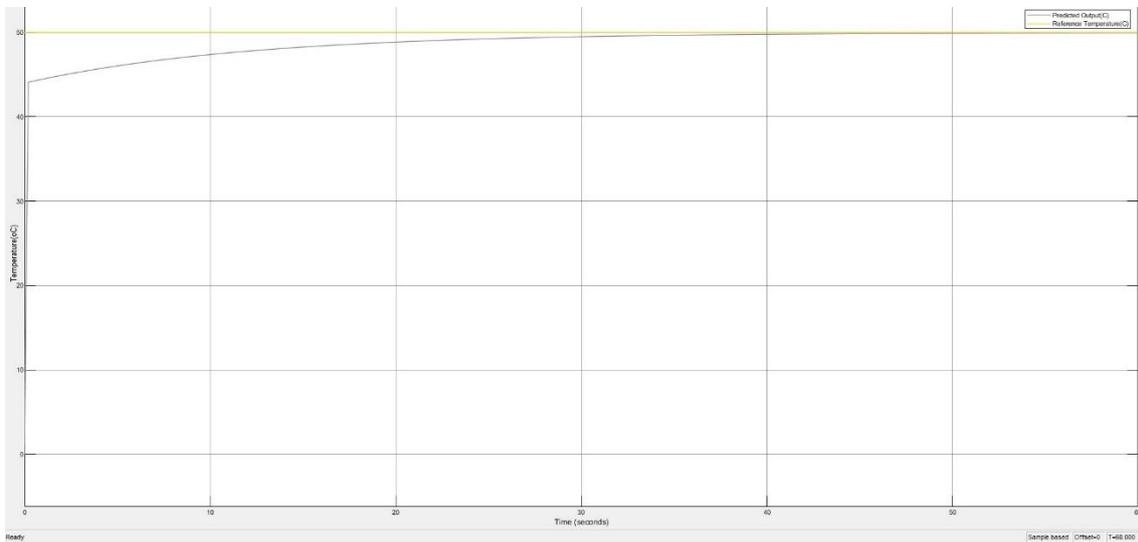


Figure 26:Graph of the temperature change of PCB heater and NCP21XV103J03RA thermistor in time, based on MPC with 0.2 second of sample time, prediction horizon with 50 moves, control horizon with 2 steps.

#### 4.2.2 Tuning the prediction horizon for the PCB heater and thermistor

Based on the previously obtained graph, it is observed that the oscillations are reduced and provides relatively better performance; however, it is recommended to perform the same experiment with decreased prediction horizon in order to compare the results and possibly obtain better results with reduced cost. As a result, the prediction horizon is set to 40 moves, sample time and control horizon remains the same. This is provided into the system as follows:

```
>> mpcobjvt=mpc(sysvt,0.2,40,2);
```

The obtained graph of the temperature ( $^{\circ}\text{C}$ ) change of PCB heater and NCP21XV103J03RA thermistor in time (seconds) based on MPC with 0.2 second of sample time, prediction horizon with 40 moves, control horizon with 2 steps is given in Figure 27.

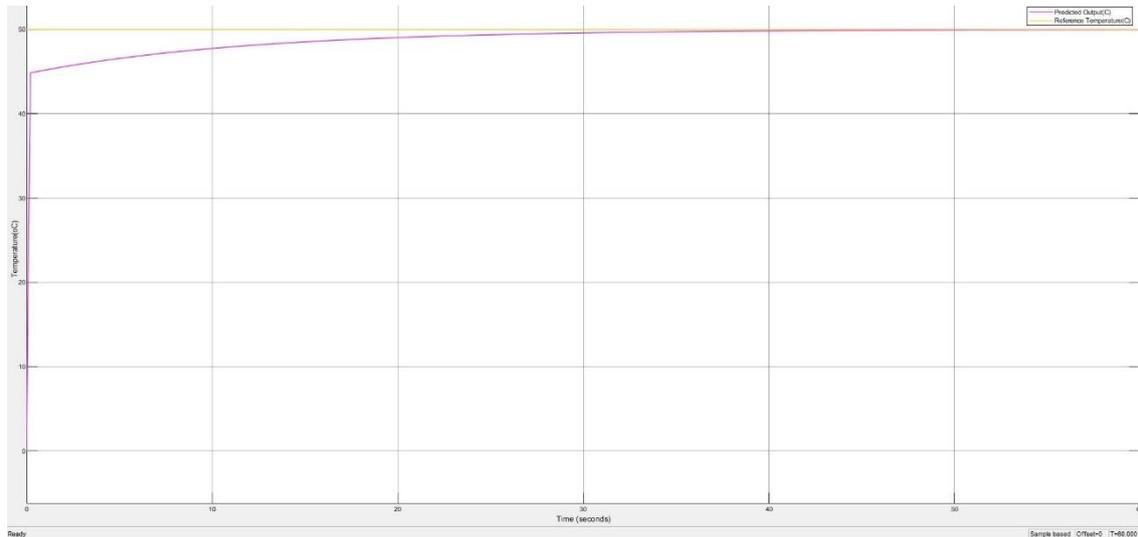


Figure 27: Graph of the temperature change of PCB heater and NCP21XV103J03RA thermistor in time, based on MPC with 0.2 second of sample time, prediction horizon with 40 moves, control horizon with 2 steps.

Following that, it is suggested to proceed with experiments by decreasing the prediction horizon to observe the impact of this change into the system. To do that, sample time and control horizon remains the same and prediction horizon is set to 30 moves. This is provided into the system as follows:

```
>> mpcobjvt=mpc(sysvt,0.2,30,2);
```

The obtained graph of the temperature ( $^{\circ}\text{C}$ ) change of PCB heater and NCP21XV103J03RA thermistor in time (seconds) based on MPC with 0.2 second of sample time, prediction horizon with 30 moves, control horizon with 2 steps is given in the Figure 28.

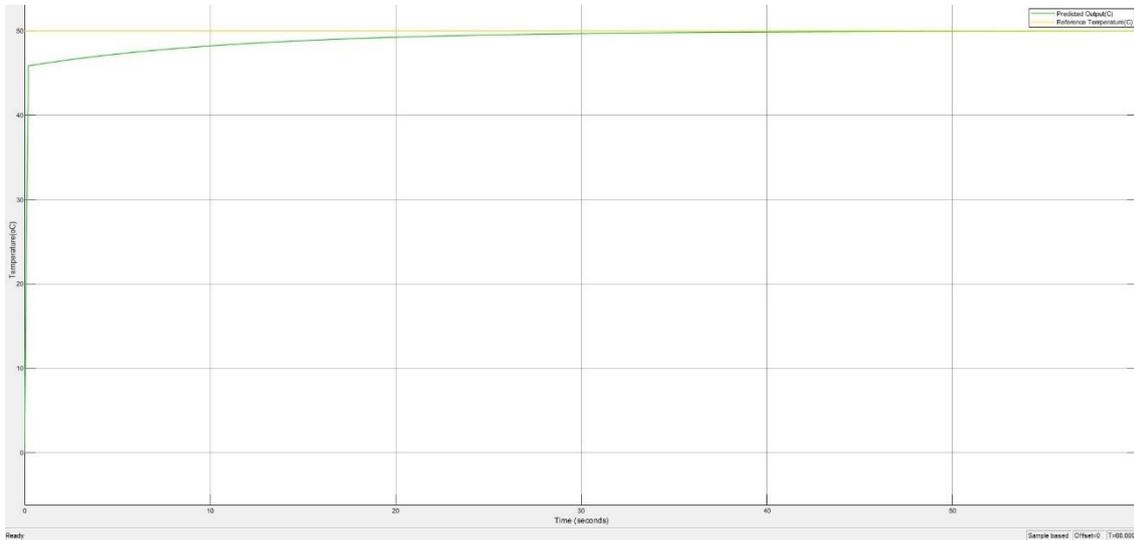


Figure 28: Graph of the temperature change of PCB heater and NCP21XV103J03RA thermistor in time, based on MPC with 0.2 second of sample time, prediction horizon with 30 moves, control horizon with 2 steps.

It is evident that the temperature converges faster when the prediction horizon is reduced and it can be considered for obtaining better performance and reduced computational cost; however, based on the principles and suggestions [29], it is recommended to hold control horizon  $\ll$  prediction horizon due to the following reasons:

- Smaller  $m$  indicates that there are fewer variables to calculate in the Quadratic Problem (QP) solved for every control interval, resulting in faster computations.
- Small  $m$  supports a controller that is internally stable.

In addition to this, as it was previously mentioned above that “it is suggested to increase  $p$  until further increment affect performance in a small way.” [29].

As a result, by taking into account the highlighted points, it is suggested to consider the number of moves which are above 50 moves to observe the changes and eventually to define the optimal value. To do that, the sample time is set to 0.2 and the control horizon is set to 2, and the prediction horizon is set to 60. This is provided into the system as follows:

```
>> mpcobjvt=mpc(sysvt,0.2,30,2);
```

The obtained graph of the temperature (°C) change of PCB heater and NCP21XV103J03RA thermistor in time (seconds) based on MPC with 0.2 second of sample time, prediction horizon with 60 moves, control horizon with 2 steps is given in Figure 29.

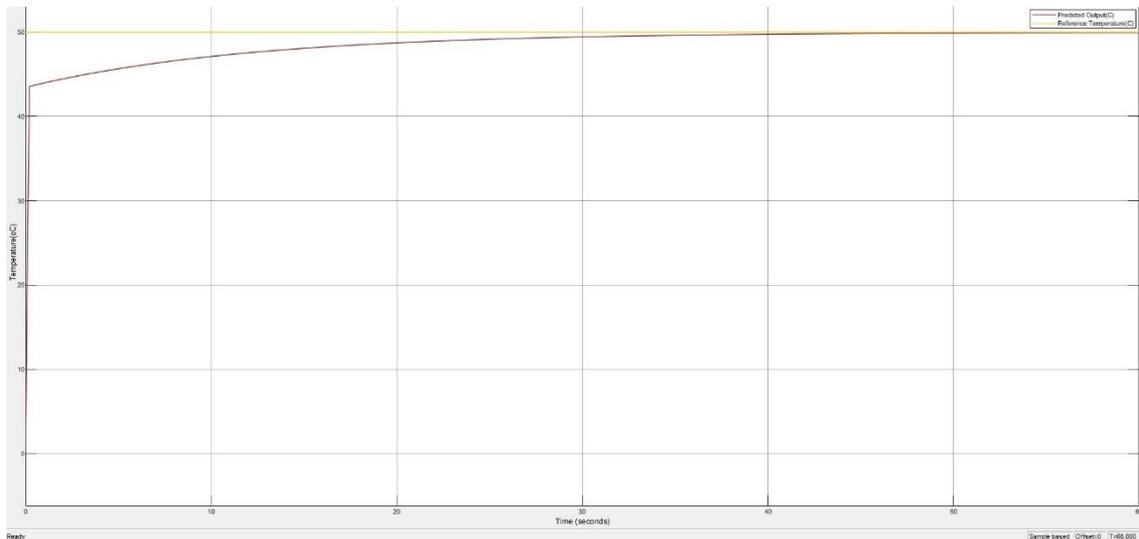


Figure 29:Graph of the temperature change of PCB heater and NCP21XV103J03RA thermistor in time, based on MPC with 0.2 second of sample time, prediction horizon with 60 moves, control horizon with 2 steps.

Based on Figure 29, there is only a slight difference between 50 and 60 moves in terms of performance and convergence. Therefore, setting the prediction horizon to 50 moves will be suitable; however, it is still necessary to confirm this statement by further experiments by setting the prediction horizon to 70 moves. To do that, the sample time is set to 0.2 and the control horizon is set to 2, and the prediction horizon is set to 70. This is provided into the system as follows:

```
>> mpcobjvt=mpc(sysvt,0.2,70,2);
```

The obtained graph of the temperature (°C) change of PCB heater and NCP21XV103J03RA thermistor in time (seconds) based on MPC with 0.2 second of sample time, prediction horizon with 70 moves, control horizon with 2 steps is given in Figure 30.

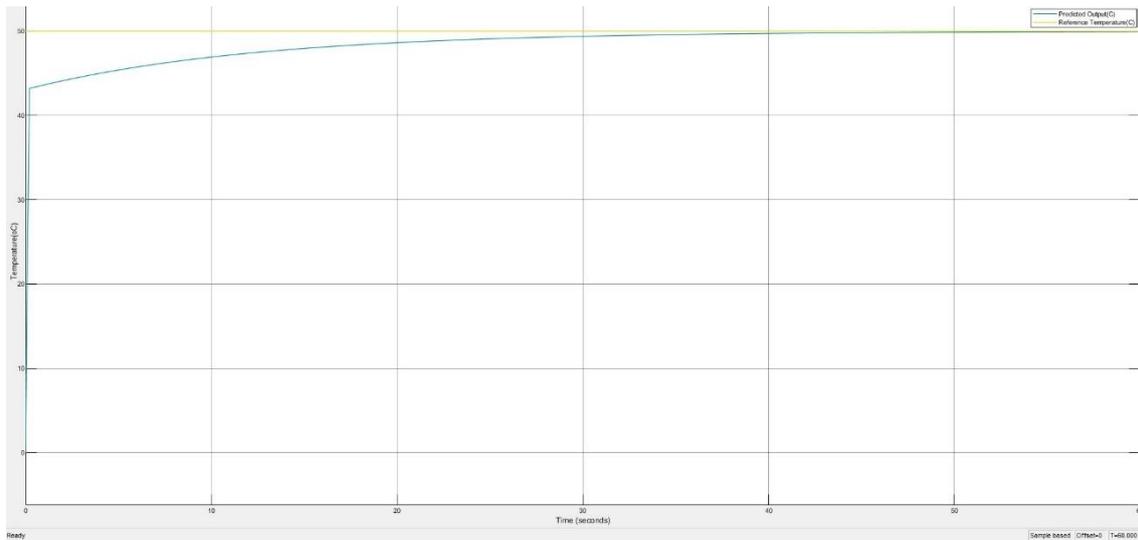


Figure 30: Graph of the temperature change of PCB heater and NCP21XV103J03RA thermistor in time, based on MPC with 0.2 second of sample time, prediction horizon with 70 moves, control horizon with 2 steps.

Compared to the graphs with 50 moves and 60 moves, it is observed that the results are similar to the previous ones. This means that performance and convergence of the system is not significantly affected anymore; therefore, prediction horizon of 50 moves can be considered as the optimum value.

Finally, it is concluded that for our use-case, MPC should be set to 0.2 second sample time, prediction horizon set to 50 moves, and control horizon set to 2 steps. with this tuning of the parameters, favorable performance and computational cost can be achieved for our use-case.

This chapter has presented various experiments conducted to investigate the behavior of MPC when tuning its parameters. Two heating elements have been considered, i.e. heating pad and PCB heater with thermistor. For both, the sample time and prediction horizon have been tuned following the established principles and suggestions for MPC. Doing so enabled to progressively improve the results.

## 5 Conclusion

### 5.1 Summary

The starting point of this MSc thesis originated from ongoing research activities at Tallinn University of Technology related to bioanalytical devices based on lab-on-a-chip/droplet microfluidics. These activities aim at providing fast response to bacterial threats and infectious diseases by means of in-the-field/PoC testing to help saving human lives and reduce epidemic risks. As part of those research activities, proofs of concept for low-cost portable bioanalytical devices are being developed. This MSc thesis specifically focused on the control aspect of such devices.

In particular, the thesis focused on MPC because of its advantages (ability to optimize the current timeslot and at the same time to take the next ones into account; work with a time-horizon that is finite, but only implement the current timeslot and repeat the optimization; solve an optimization problem i.e. can predict the future plant outputs as close as to the desired output reference value).

This MSc thesis sought to investigate the potential of MPC for resource-constrained platforms in view of realizing low-cost portable bioanalytical devices. After introducing two examples of such devices as well as presenting design considerations, experiments have been conducted to tune an MPC algorithm for two different heaters (heating pad and PCB heater with thermistor). Following the established principles and suggestions for MPC, the sample time and prediction horizon have been tuned to progressively improve the results.

As a result, it is deemed that the objective of this MSc thesis has been achieved. Given this, the team members of the research project to which the thesis are associated plan to continue investigating and implement MPC as part of their research activities. The following section provides suggestions for future work, which might also be useful for the research team.

## 5.2 Perspectives

The work that has been conducted in this MSc thesis has confirmed the potential of MPC for the design and implementation of bioanalytical devices. Suggestions for future work include the following items:

- Implement and test the MPC algorithm on the Raspberry Pi 4 SBC
- Integrate the MPC algorithm with the computing modules that are being developed in the research projects to which this thesis is related.
- Finally, extensive lab-testing of the fully implemented system should be conducted to evaluate its robustness in conditions similar to that of in-the-field operation.

## References

- [1] A. Adan, G. Alizada, Y. Kiraz, Y. Baran, and A. Nalbant, "Flow cytometry: basic principles and applications," *Critical Reviews in Biotechnology*, vol. 37, no. 2. Taylor and Francis Ltd, pp. 163–176, Feb. 17, 2017, doi: 10.3109/07388551.2015.1128876.
- [2] H. Sink, "Development of A Smart Temperature Regulator for Isothermal DNA Amplification Lab-On-A-Chip Systems,". MSc thesis, Tallinn University of Technology, 2019.
- [3] T. Pardy, *Microheating Solution for Molecular Diagnostics Devices*. 2018.
- [4] T. S. Hawley, R. G. Hawley, and A. L. Givan, *Flow CytometryFlow CytometryProtocolsProtocols*, vol. 263. 2004.
- [5] B. Misha Rahman, A. Lane, A. Swindell, and S. Bartram, "Introduction to Flow Cytometry Principles Data analysis Protocols Troubleshooting Your first choice for antibodies! Introduction to Flow Cytometry." Accessed: Jan. 19, 2021. [Online].
- [6] M. Brown and C. Wittwer, "Flow Cytometry: Principles and Clinical Applications in Hematology," 2000. Accessed: Jan. 19, 2021. [Online]. Available: <https://academic.oup.com/clinchem/article/46/8/1221/5641363>.
- [7] D. Kaygusuz, S. Vural, A. Ö. Aytekin, S. J. Lucas, and M. Elitas, "DaimonDNA: A portable, low-cost loop-mediated isothermal amplification platform for naked-eye detection of genetically modified organisms in resource-limited settings," *Biosensors and Bioelectronics*, vol. 141, Sep. 2019, doi: 10.1016/j.bios.2019.111409.
- [8] D. Ibrahim, *Microcontroller Based Applied Digital Control*. 2006.
- [9] Z. Artstein, "Discrete and Continuous Bang-Bang and Facial Spaces Or: Look for the Extreme Points," *SIAM Review*, vol. 22, no. 2, pp. 172–185, Apr. 1980, doi: 10.1137/1022026.
- [10] "Khayal, Osama. (2016). Control System Design for Continuous Stirred Tank Reactor Using Matlab Simulink. 10.13140/RG.2.2.17177.11362. ," Accessed: Dec. 14, 2020. [Online].
- [11] A. Nabi, T. Stigen, and J. Moehlis, "Firing rate control of a neuron using a linear proportional-integral controller Related content Minimum energy control for in vitro neurons," 2010, doi: 10.1088/1741-2560/7/6/066004.
- [12] K. S. Rao and R. Mishra, "Comparative study of P, PI and PID controller for speed control of VSI-fed induction motor," 2014, Accessed: Jan. 03, 2021. [Online]. Available: [www.ijedr.org](http://www.ijedr.org).
- [13] K. H. Ang, G. C. Chong, and Y. Li, "PID control system analysis, design, and technology," *IEEE Transactions on Control Systems Technology*, vol. 13, no. 4, p. 559, 2005, doi: 10.1109/TCST.2005.847331.
- [14] K. S. Holkar, K. K. Wagh, and L. M. Waghmare, "An Overview of Model Predictive Control," 2010. Accessed: Jan. 20, 2021. [Online].
- [15] A. Ramdani and S. Grouni, "Dynamic matrix control and generalized predictive control, comparison study with IMC-PID," *International Journal of Hydrogen Energy*, vol. 42, no. 28, pp. 17561–17570, Jul. 2017, doi: 10.1016/j.ijhydene.2017.04.015.
- [16] R. Haber, J. A. Rossiter, and K. Zabet, "An alternative for PID control: Predictive Functional Control - A tutorial," in *Proceedings of the American Control*

- Conference*, Jul. 2016, vol. 2016-July, pp. 6935–6940, doi: 10.1109/ACC.2016.7526765.
- [17] J. Mattingley and S. Boyd, “CVXGEN: a code generator for embedded convex optimization,” vol. 13, pp. 1–27, 2012, doi: 10.1007/s11081-011-9176-9.
- [18] C. Supervisor and O. Scheler, “Development of Smart Flow Regulation for Lab-on-a-Chip Applications Nutika voolu reguleerimise seadme arendamine kiiplaborsüsteemide rakenduste jaoks,” 2021.
- [19] G. R. D. Prabhu *et al.*, “Facilitating chemical and biochemical experiments with electronic microcontrollers and single-board computers,” *Nature Protocols*, vol. 15, no. 3, pp. 925–990, Mar. 2020, doi: 10.1038/s41596-019-0272-1.
- [20] M. Omamageswari, A. Mohanraj, S. Carolin Jeeva, A. Kishore Reddy, K. Thilagam, and U. Penchalaiah, “IoT based smart drainage monitoring and cleaning system for solid waste materials,” *Materials Today: Proceedings*, Feb. 2021, doi: 10.1016/j.matpr.2020.11.940.
- [21] J. Jiang, M. Moallem, and Y. Zheng, “An intelligent iot-enabled lighting system for energy-efficient crop production,” *Journal of Daylighting*, vol. 8, no. 1, pp. 86–99, Feb. 2021, doi: 10.15627/jd.2021.6.
- [22] D. Ibrahim, “Programming the ARM-based microcontrollers,” in *Arm-Based Microcontroller Multitasking Projects*, Elsevier, 2021, pp. 55–59.
- [23] J. O. Bauer, “The Basics of MATLAB.” Accessed: May 07, 2021. [Online].
- [24] “System Identification Toolbox Documentation - MathWorks Nordic.” <https://se.mathworks.com/help/ident/> (accessed May 07, 2021).
- [25] “Curve Fitting Toolbox Documentation - MathWorks Nordic.” [https://se.mathworks.com/help/curvefit/index.html?searchHighlight=curve%20fitting&s\\_tid=srchtitle](https://se.mathworks.com/help/curvefit/index.html?searchHighlight=curve%20fitting&s_tid=srchtitle) (accessed May 07, 2021).
- [26] “Simulink ® Getting Started Guide R2013b,” 1990. Accessed: May 07, 2021. [Online]. Available: [www.mathworks.com](http://www.mathworks.com).
- [27] “Non-Carbon Fiber Electrical Heating Textile.” Accessed: Apr. 20, 2021. [Online].
- [28] K. Worthmann, “Estimates of the prediction horizon length in MPC: A numerical case study,” in *IFAC Proceedings Volumes (IFAC-PapersOnline)*, Jan. 2012, vol. 45, no. 17 PART 1, pp. 232–237, doi: 10.3182/20120823-5-NL-3013.00037.
- [29] “Choose Sample Time and Horizons - MATLAB & Simulink - MathWorks Nordic.” <https://se.mathworks.com/help/mpc/ug/choosing-sample-time-and-horizons.html> (accessed Apr. 28, 2021).
- [30] G. Cornuéjols, J. Peña, and R. Tütüncü, “Quadratic Programming: Theory and Algorithms,” in *Optimization Methods in Finance*, Cambridge University Press, 2018, pp. 71–89.
- [31] “Product Search Data Sheet NCP21XV103J03RA.” Accessed: Apr. 22, 2021. [Online]. Available: <http://www.murata.com/en-us/products/productdetail?partno=NCP21XV103J03RAURL:https://www.murata.com/>.
- [32] “File:MPC scheme basic.svg - Wikimedia Commons.” [https://commons.wikimedia.org/wiki/File:MPC\\_scheme\\_basic.svg](https://commons.wikimedia.org/wiki/File:MPC_scheme_basic.svg) (accessed May 02, 2021).

# Appendix 1 – Non-exclusive licence for reproduction and publication of a graduation thesis<sup>1</sup>

I Dermansah Hazar Ersoy Yalin

1. Grant Tallinn University of Technology free licence (non-exclusive licence) for my thesis Investigation of Model Predictive Control in Bioanalytical Devices, supervised by Yannick Le Moullec, co-supervised by Kanwal Ashraf
  - 1.1. to be reproduced for the purposes of preservation and electronic publication of the graduation thesis, incl. to be entered in the digital collection of the library of Tallinn University of Technology until expiry of the term of copyright;
  - 1.2. to be published via the web of Tallinn University of Technology, incl. to be entered in the digital collection of the library of Tallinn University of Technology until expiry of the term of copyright.
2. I am aware that the author also retains the rights specified in clause 1 of the non-exclusive licence.
3. I confirm that granting the non-exclusive licence does not infringe other persons' intellectual property rights, the rights arising from the Personal Data Protection Act or rights arising from other legislation.

10.05.2021



---

<sup>1</sup> The non-exclusive licence is not valid during the validity of access restriction indicated in the student's application for restriction on access to the graduation thesis that has been signed by the school's dean, except in case of the university's right to reproduce the thesis for preservation purposes only. If a graduation thesis is based on the joint creative activity of two or more persons and the co-author(s) has/have not granted, by the set deadline, the student defending his/her graduation thesis consent to reproduce and publish the graduation thesis in compliance with clauses 1.1 and 1.2 of the non-exclusive licence, the non-exclusive license shall not be valid for the period.