DAS Departamento de Automação e Sistemas
CTC Centro Tecnológico
UFSC Universidade Federal de Santa Catarina

Development of an Evaluation Program for Low Pressure Plasma Processes

Monografia submetida à Universidade Federal de Santa Catarina

como requisito para a aprovação da disciplina:

DAS 5511: Projeto de Fim de Curso

Leandro Feltrin Zanellatto

Florianópolis, março de 2012

Development of an Evaluation Program for Low Pressure Plasma Processes

Leandro Feltrin Zanellatto

Orientadores:

Henrik Behm, -Dipl. Ing.

____

Assinatura do Orientador

Prof.*Hector Bessa Silveira*

____

Assinatura do Orientador

Este relatório foi julgado no contexto da disciplina
DAS5511: Projeto de Fim de Cusro
E aprovado na sua forma final pelo
Curso de Engenharia de Contrôle e Automação

Acknowledgements

It would be unfair to start these acknowledgements without thanking my parents Carlos and Adriana who always supported me and gave me the best education that they could not only technical knowledge but moral values.

To all my friends who were with me in this journey from the beginning and the new ones that I made, thanks for being with me in the good and bad moments.

And finally I would like to thank God for listening my prayers, giving me strength and for putting so many amazing people in my way.

Abstract

Plasmas can be seen in many daily applications such as televisions and light bulbs, they can also be used for the modification of various surfaces.

Recent researches in plasma application to surface modification are being developed at Institutfür Kunststoffverarbeitung (IKV)[1]– Aachen to increase the coating barrier and the surface properties of polymers using plasma.

The coating barrier of polymers is been studied to inhibit the interaction between the inside with the outside of a PET bottle. With this inhibition the PET bottle can store other kinds of liquids without losing taste and increasing the expiration date. This study is made due the fact that PET bottles are cheaper and easier to be recycled.

It is known that the control of plasma process is really important because a small variation in an internal parameter can compromise the whole process.

To procedure with the evaluation it is important to know the behavior of the intensity during the time of each gas, this is important because when plasma is generate by a gas it is not a hundred percent pure, so a lot of other gases are mixed due to different reasons.

To guarantee the control of the process a lot of equipments were used, among them the Optical Emission Spectroscopy (OES), which is a sensor that allow one to obtain information about plasma composition.

This sensor gives a lot of data and information, and it is required to extract only the important and useful information, In order to accomplish such goal, a computer program was developed to make this data available into a decent and easy format.

This project developed a computer program with a friendly interface to show the sensor data in a graphical format and to allow interaction with a SQL based database which is stored at the IKV server.

[1]Institute of Plastic Processing

It is important to have this interaction with the database because it stores process parameters and the final user can chose the suitable evaluation by checking those parameters into the database.

The computer program developed in this work makes it possible to analyze the plasma that was generated and evaluate the accuracy and quality of the low pressure plasma plants.

Resumo Estendido

Tecnologia de processamento de plasma é uma atividade de extrema importância para indústrias de manufatura de todo mundo, especialmente quando se trata da indústria de polímeros ou plásticos. Neste tipo de indústria em específico, plasma é utilizado para refinamento de superfície como criar barreira para gases, melhorar a adesão e aumentar a resistência contra rachaduras.

O instituto de processamento de plásticos (IKV),no qual esse projeto foi desenvolvido, trabalha com pesquisa nessas três áreas de aplicação dos processos de plasma na indústria de plásticos, contando com vários equipamentos, dentre os principais encontram-se câmaras de plasma de baixa pressão para gerar o plasma para utilização na pesquisa e o Optical Emission Spectroscopy[2],(OES) que é um sensor não invasivo utilizado para colher informações sobre a composição do plasma.

É possível modificar a superfície do material dependendo do gás precursor utilizado em conjunto com o controle do processo. Tal técnica é de extrema importância em aplicações como a criação de barreiras em embalagens plásticas, para impedir a troca de gases entre o meio interno e externo mantendo sabor e aumentando a data de validade, e a melhoria de superfícies aumentando a adesão de superfícies ou diminuindo a adesão, de água, por exemplo, em superfícies plásticas para as mais diversas aplicações.

Qualquer pequena variação num parâmetro interno pode causar grandes mudanças no desempenho do plasma. Assim é importante e necessário monitorar os parâmetros do plasma e seu comportamento para garantir que as propriedades desejadas sejam mantidas.

Para efetuar o controle de qualquer processo, a primeira coisa que se deve fazer é monitorar as variáveis controláveis e garantir que elas se comportem da maneira que se espera atuando no processo.

[2]Espectroscópio de Emissão Ótica.

Esse monitoramento, ou melhor, medição de variáveis é feito através do sensor OES, porém este é um sensor muito complexo que gera muitas informações. Então se faz necessário filtrar essas informações e representá-las de uma forma de fácil visualização e análise.

Por exemplo, para avaliar a qualidade e exatidão do plasma gerado é importante saber o comportamento da intensidade de cada gás durante o tempo durante a geração. Isto é importante devido ao fato da geração de plasma ser feita a partir de um gás e durante o processo uma mistura de gases pode entrar na câmara pelas mais diversas razões. Por isso é necessário analisar se esses valores de intensidade dos gases estão dentro dos padrões estabelecidos.

Em posse dos dados obtidos e gerados pelo OES, um programa para avaliar processos de plasma de baixa pressão foi desenvolvido utilizando a plataforma LabView, avaliando a composição do plasma gerado pelas câmaras de baixa pressão, extraindo somente as informações úteis para a análise e representando esses dados de uma forma gráfica e visual.

Esta idéia de extrair os dados do OES para avaliar os processos de baixa pressão já existia em dois mil e dez quando um programa de Matlab foi desenvolvido para fazer essa representação gráfica da composição do plasma gerado.

A principal diferença entre o programa desenvolvido em 2010 e este que será apresentado neste projeto é o fato do segundo interagir e exportar dados para uma base de dados. Isso possibilita escolher o processo a ser avaliada a partir de alguns parâmetros tais como, a máquina que de plasma, a data de medição ou até mesmo a ferramenta de medição via OES.

O projeto descrito neste documento desenvolveu um programa computacional de manipulação e disponibilização de dados. A partir da interface gráfica deste programa podem-se extrair dos arquivos de texto gerados pelo sensor OES as variáveis e parâmetros úteis. A partir desses dados é possível gerar gráficos e tabela se exportá-los para o Excel e para um banco de dados baseado em SQL.

A variável mais importante extraída dos arquivos é o comportamento da intensidade do comprimento de onda dos gases ao longo do tempo. O OES ao efetuar sua medição gera vários arquivos de texto, e cada um deles armazena o valor da intensidade para um determinado comprimento de onda, cada arquivo de texto representa um instante de tempo na medição.

Os parâmetros possíveis de serem extraídos dos arquivos de medição são a máquina em que o plasma foi gerado, a data que a medição foi efetuada, e a conexão de entrada da ferramenta de medição que afeta diretamente a precisão da medição.

Com o programa computacional desenvolvido neste projeto, o usuário terá à sua disposição uma ferramenta de auxílio para o monitoramento e controle dos processos de plasma de baixa pressão, avaliando a sua precisão e qualidade a partir do comportamento da intensidade dos gases, identificados pelos seus comprimentos de onda, ao longo do tempo.

Summary

Chapter 1: Introduction

1.1: Motivation

Plasma processes constantly gain importance in the field of plastics processing. They are used in large scale for surface refinement such as scratch resistance, barrier or adhesion improvement.

The processes are influenced by production parameters on the one hand and the substrate itself on the other hand. The properties of polymers vary substantially depending on their processing conditions, their history or environmental conditions. Therefore the influence of the polymer properties, for instance molecular orientation and crystallinity, on plasma processes is investigated fundamentally.

Plastics processing, polymer properties, surface modification processes, plasma properties and surface properties are correlated via plasma-substrate-interaction.

"Plasma processing technology is vitally important to several of the largest manufacturing industries in the world. Plasma-based surface processes are indispensable for manufacturing the very large scale integrated circuits used by the electronic industry. Such processes are also critical for the aerospace, automotive, steel, biomedical, and toxic waste management. Materials and surface structures that are fabricated are not attainable by any commercial method, and the surface properties of materials can be modified in unique ways [LL05]".

"Plasma treatment has been used throughout industry for many processes, including surface cleaning, roughening, activation, film deposition, and patterning [SWS97]".

Some examples of plasma application commonly use are fluorescent light bulbs, welding torches, plasma TV and so on.

Plasma, here at the AG Plasma research group, is used to surface properties modification. These modifications are made in order to apply plasma as coating barrier for pat bottles mainly. More about the activities from the research group will be explained in the next chapter.

“The non-uniformity of most plasma processes and sensitivity to environmental changes make it difficult to maintain process stability from day to day. Any manufacturer that relies on plasma as a process step, whether for cleaning, etching, surface enhancement, patterning, etc. is vulnerable to process non-uniformity. Plasma processes are susceptible to external factors such as environmental humidity and internal factors such as power, pressure, gas purity, mechanical failures, and electrical failure. Plasma processes often fail to perform as expected [SWS97]”.

A small deviation in any internal factor can cause large shifts in plasma performance, and that is why it is necessary to monitor the plasma parameters and behavior in order to ensure that the required properties are have been attained.

1.2: Project

A new plasma-reactor was built to be used for fundamental research of plasma-surface-interaction. A gas excited in a way that electrons gain enough kinetic energy for inelastic impacts on molecules, so that the gas becomes ionized. Partially ionized gas is called plasma. In order to treat and coat polymers, low pressure plasma is used.

Depending on the used precursor gas and the process control it is possible to modify the surface of a material. Thus the control of the process is substantial for a repeatable modification or coating of good quality. The process is monitored by means of optical emission spectroscopy (OES) as well as a Langmuir probe.

The OES measures the emission of excited particles in the plasma. With a Langmuir probe it is possible to measure the time resolved electron density. The measurement data are recorded in a file. However, the file contains a lot of data.

The main objective of the present project is to write a *LabView* program capable of getting the data from the optical emission spectroscopy (OES), extract only the needed information and transform this information directly into a defined format, in order to evaluate the low pressure plasma process.

The main variable that OES provides is the intensity of each gas contained in the plasma. With this information it can be verify if the plasma generated was uniform and if the gases have the specified intensity. It is necessary due to the fact that projects where plasma is applied, plasma has to have an very specific and unique property, so it is important to verify the composition of the plasma analyzing the intensity of the gases.

This project idea is not a new idea, there was a *Matlab* program developed in 2010 which got the OES data and plotted graphs. The Later did not have the possibility to interact with the database, which is the main difference between this old program and the program developed and described in this project. Also this former was not developed to interact with the starter program of plasma reactors, differently of the program presented in this project which was thought to connect with the starter program later.

This program also has to interact with a SQL based database located at the AG Plasma server, AG Plasma is the research group of IKV where this present project was developed, server. All the data and the relevant information about all the projects are stored into this server, so it is important to have also the OES data stored there in order to have an easy access to it. Also this program should be in module and easy to concatenate with an existing *LabView* program that runs the plasma-reactor, in order to get the data in real time.

The interface must be easy to use and self-explanatory.

1.3: Developed Activities

In that five month of the project a LabView program was written, this program is capable of getting the data from the OES and releases those data in an easy way to manage them.

In the beginning it was studied the plasma processes and learned how to procedure with the measurements with OES. After getting OES data it was decided which information was important to store and to work with. At the same time a few LabView initial programs were developed in order to get practice with the software.

In the second month the programming step was initialized. The first step was to do some measurement with the OES and write a program to store those data into a three dimension matrix with the three important variables of the OES measurement, time, wave length and intensity of wave length.

Now that the data are stored into a matrix an interface was developed to plot the graphs to make the evaluation easier for the user and to give the possibility to make some operations with the data, as apply a calibration, load a reference or normalize in a point, also the interface developed by me give the possibility to export the data to a Microsoft Excel file or to a database.

A database specifically for those data was created into the research group database server to store the data and in the future procedure with the evaluation faster and without running all the data from OES again.

After the database structure (tables and rows) was decided a program was written to export the data from the three dimension matrix to the database. Also an interface was created by me to import the data from the database and use it to analyze the process.

With this program now it is possible to analyze the plasma that was generated and evaluate the accuracy and quality of the low pressure plasma plants. This evaluation is made by a user using the graphs and the tables generated by this program together with some work station parameters stored into the database.

1.4: Document Structure

The document is organized as follows.

In chapter two, a quick explanation about the IKV and the research activities at AG Plasma are described. Also a brief concept of plasma is present in this chapter.

Chapter three is the one which gives the idea of the project, describing and explaining the problem and the solution proposed together with the used methodology to solve the problem. This chapter does not presents technical details about the solution, those details are presented after in this document.

In chapter four the equipments used to get the data and the tools used to write the program are presented with their importance to the final objective of this project.

From the chapter five to seven the program developed is shown with comments about the results, the difficulties found while the programming step and about the modifications that took place during the project.

In chapter five the main user interface is presented with the LabView code used to get to the final version. In this chapter technical details about how the problem was solved are presented.

Chapter six presents the first version of the database programming which later was discarded. In this section will be explained why this part was not used in the final version of the software.

The seventh chapter shows the final version of the database program used in this current project. This chapter explains the alterations made; the program made with LabView to export and read the data, shows the tables and parameters stored into them; and explains the choice of this database structure.

The results and how the project helped the research group are presented in chapter eight. In the last chapter is shown the final considerations and a few comments about the future of this project.

Chapter 2: Plasma Research at IKV

2.1: Introduction

In this section it will be explained the institute where this project was developed and the research group and its activities.

To understand more about the activities with plasma, a small section explaining plasma and cold plasma was written.

This chapter will situate the reader in the work environment where this project was developed and contextualize the activities developed in the research group with the project described in this document.

2.2: Plasma

“Plasma is known as the fourth state of matter. It was Langmuir in 1929 that used the term “plasma” for the first time to describe ionized gases. Plasma can be defined as a quasi-neutral gas of charged and neutral particles characterized by a collective behavior [Gri93]”.

Therefore, plasma is an ionized gas consisting of positively and negatively charged particles with approximately equal charge density and in an unbound gaseous state.

Most of the material in the visible universe is in the plasma state, but in the Earth’s atmosphere it is not very common due the low temperatures and high pressures.

“Plasma can be generated by adding energy to atoms of a gas; this energy must be higher than the ionization energy, producing ions and electrons. Also the fourth state of matter can be obtained when energy is provided to a liquid or a solid that causes their vaporization and ionization [Gri93]”, but the most common way to produce plasma is passing by an electrical discharge through a gas.

“There are many ways to excite a gas and give it and sustain energy to it, like direct current, radio frequency and microwaves.

In this work plasmas are generate via direct current and microwaves.

The plasma is broadly characterized by the following basic parameters:

- The *density of the neutral particles*, n_n;
- Plasma density n; where n is the density of electrons and ions in a quasi-neutral state;
- The energy distribution of the neutral particles, $f_n(W)$; ions $f_i(W)$; and electrons $f_e(W)$ [Gri93] “.

2.2.1: Plasma Types

“Taking into account the wide ranges of parameters, the plasmas are classified into several categories:

- Plasmas in complete thermodynamic equilibrium - CTE plasmas: In a CTE plasma all temperatures (gas, electron, radiation temperature) are equal. They have no practical importance because they do not exist in controlled laboratory conditions.
- Plasmas in local thermodynamic equilibrium - LTE plasmas: These are plasmas in which all temperatures, except the radiation temperature, are equal in each small volume of the plasma.
- Plasmas that are not in any local thermodynamic equilibrium - non-LTE plasmas. These plasmas, also named cold plasmas, are the plasma used in this project [Gri93]”

The plasma type used in this project is not in any local thermodynamic equilibrium.The advantage is a low process temperature.

2.3: Cold Plasma

In the non-LTE plasmas the temperature of the electrons (10^4-10^5°K) is much higher than gas temperature (300 °K). Therefore this kind of plasma is called cold plasma.

The cold plasmas have been developed specifically and purposefully based on their non-equilibrium properties and their capability to cause physical and chemical reactions with the gas at relatively low temperatures.

2.4: IKV

The Institute of Plastics Processing (IKV) is an institute affiliate at RWTH Aachen University and it is one of the biggest institutes engaged in this field of research.

“In organizational terms, the IKV is divided up into the four specialist departments of Injection Molding and PU Technology, Extrusion and Further Processing, Part Design and Materials Technology, and Composites / PU Technology [URL00]”.

The IKV Part Design / Materials Technology department covers many areas of research, among them: Barrier Coating of Polymers and Functionalisation of Plastic Surfaces. These are the areas of the research group in which the present project was developed.

2.5: AG Plasma

AG Plasma is a research group leading by –Ing. Henrik Behm and-Ing. Karim Bahroun that develops activities related to surface modifications and improvement using plasma. As said before the leading projects are Barrier Coating of Polymers and Functionalisation of Plastic, below a little explanation about those projects and how my project relates with them.

It is important to say that the research group has three plasma reactors, named as Boot, Kiaba and Lamps, each one with a specific application.

2.5.1: Barrier Coating of Polymers

The central aspect of the work is to generate novel surface properties in order to compensate restrictive features such as e.g. inadequate high gas permeability.

In order to accomplish the perfect coating some equipments are required such as microwave excited low-pressure plasma plants, and low frequency excited low-pressure plasma plants and so one. Also to get the perfect coating is necessary to monitor and analyze the process.

The computer program developed in this work makes it possible to analyze the plasma that was generated and evaluate the accuracy and quality of the low pressure plasma plants.

2.5.2: Functionalisation of Plastic

The main focus is to modify plastic surfaces in order to generate novel surfaces properties, being independent from the bulk material. For instance the adhesion characteristics, barrier performance, or abrasion properties can be considerably improved.

“The adjustment of adhesion properties is the most important application of plasma technology. Thus plasma treatment is commonly used for activation and cleaning of polymer surfaces before further production steps [URL02]”.

Also this project counts with a lot of measurement equipments and one of them is the OES.

It is important to point out that the OES program developed is used to evaluate low pressure plasma process, because all of the plasma plants used for both projects are excited by low pressure gases. But the data generated by OES and processed into the evaluation program developed can be applied for other plasma process.

When plasma is generate by a gas it is not a hundred percent pure, so a lot of other gases are mixed due to different reasons. To procedure with the evaluation it is important to know the behavior of the intensity during the time of each gas, to make sure that the plasma generated is within the standards stipulated.

The main research activities for the both projects, Barrier Coating Polymers and Functionalisation of Plastic are:

- Plant development and up-scaling for industrial application;
- Process monitoring and analysis;

- Quality control and accelerated process development
- Process development for: modified and new plant technologies and novel substrate materials.

Now that the activities of the research group were presented and the project was contextualized with them it is time to exhibit the problem. The next chapter will introduce and describe the problem, explaining it and presenting the solutions used to solve the problem.

Chapter 3: Definition of the Problem

3.1: Introduction

This chapter describes and explains the problem which will be treated and solved in this project. Also in this section it will be exposed the solutions and the methodologies to solve the problem.

With this information the reader can understand the problem and the solutions implemented and described in the next chapters.

3.2: Describing the Problem

The problem consists in making available, in a clear and objective way, the data acquired by the OES. In possession of those data the user can analyze the accuracy and quality of the plasma, generated by one of the three plasma reactors, rapidly and visually. Those data also should be stored into a database. Those data should be read and process from the database by the evaluate program.

This project is basically a data manipulating project. The main objective was to collect only the relevant information from the OES in order to give the final user a powerful tool to analyze if the plasma has the amount of a specific gas required and if their intensity is between pre determined limits, to verify the uniformity of the plasma.

The OES sensor provides important information to analyze the plasma, which is the intensity of the gases contained in the plasma. With this information the user can decide if the plasma was generated properly or if there was any kind of flaw.

It is important to AG Plasma having this information to evaluate the quality of the plasma that the research group is using for its majors projects. To have a perfect barrier of PET bottles it is necessary that the plasma generated be uniform and follows a pattern. The same thing to do surface modifications, it is important to keep the plasma uniform to have specifics properties.

However the OES provides that information in a text file which is confusing and exhausting check each one manually, so to get the information automatically a program was built. And with this program an option to export these data to a database to store the data and work with them by the database and it is also important to keep all the samples informations to compare them.

To solve this problem of making the data available, a LabView program was developed to help the final user with the process of evaluation, fulfilling a few requirements asked by the user. In the next chapter it will be explained why the software LabView was chosen to build the program.

The requirements to the user interface are:

1. Graphical display of intensity vs. time at a specified wavelength
2. Graphical display of intensity vs. wavelength at a specified time
3. The possibility to load calibration files and apply a calibration
4. Output of certain wavelengths defined by the user to MS Excel
5. Output of certain wavelengths that are characteristic for commonly used plasmas to MS Excel
6. Export of the data matrix into a SQL-based database

The flow of the data is illustrated below:

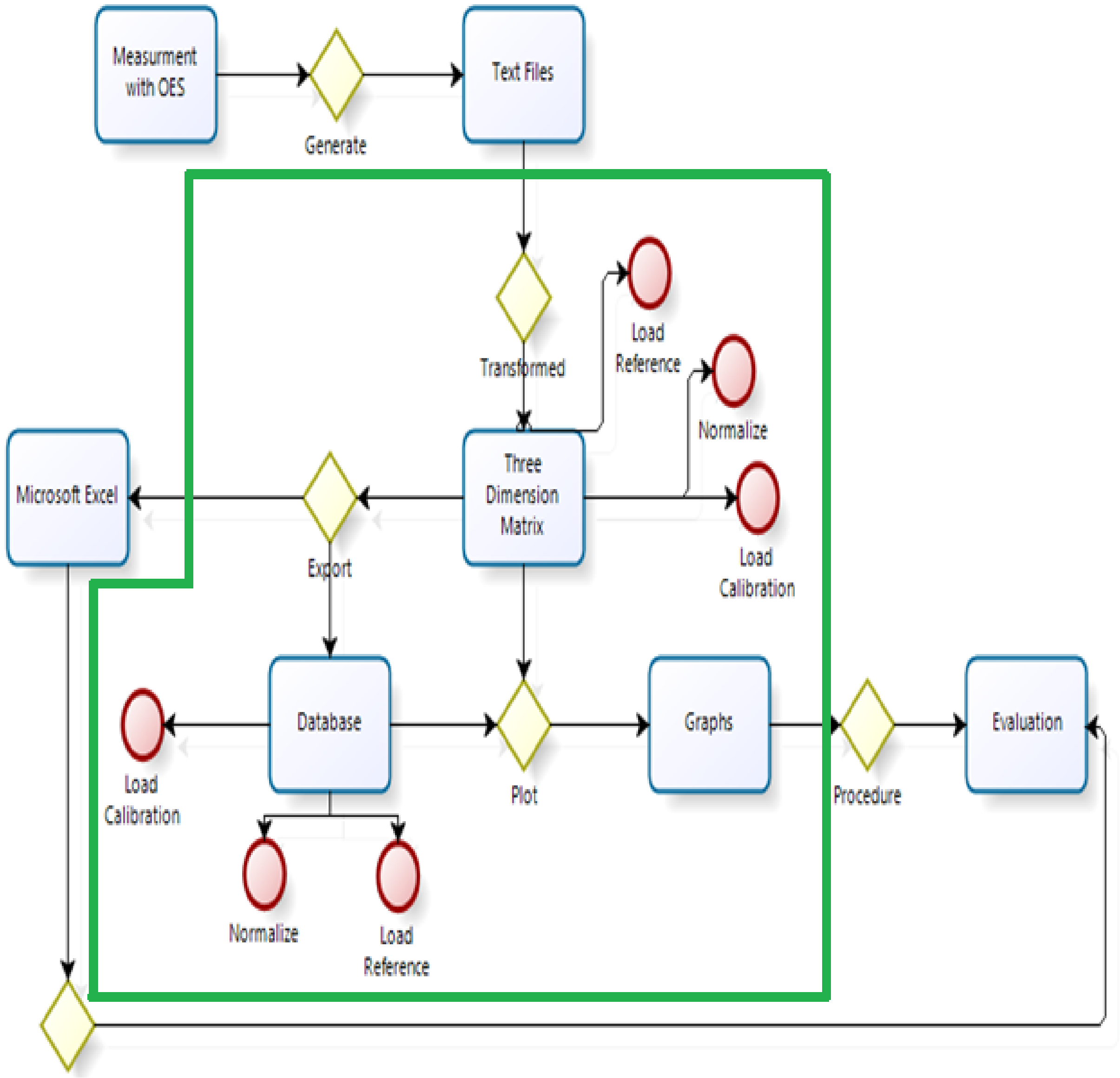

Figure 1 – Data flow until the evaluation

This picture represents the flow of the data from the measurement until the evaluation. Also the figure 1 shows how the problem was broken into small problems to solve each problem at the time. In the sequel it will be explained each step of the chart with more details.

The highlighted part in green represents the program and the operations that can be done with the data, the procedure with evaluation step needs a human user to execute, also to do the measurement with OES is made by user who can be different from the final user of the program. The OES sensor output is already a text file, so this step occurs automatically.

In conclusion, after the data leaves the sensor, it passes for a treatment until get to the evaluation form and every single operation can be controlled by the user via user's interface.

As said before the figure 1 shows the parts where the big problem was divided, now it will be explained each step, and why it was divided this way.

The general idea is making available through a friendly interface the data collected with the OES sensor. After an operator taking the measurement with the sensor he has a group of text files, each file representing a time interval of measurements. Inside those files there are two important informations that are necessary later, the wave length of gases and the intensity of these gases.

3.2.1: Transforming Text Files in a Three Dimension Matrix

Here is when the program starts to act. Due the fact that the information extracted from OES comes in separated and confusing files, it was necessary first to organize those data.

The solution used here was to build a three dimension matrix to store those three essential variables to the final evaluation process: Time, Wave Length of the gases and Intensity of the gases with each dimension representing one variable.

Each element of the matrix represents the intensity of one wave length in a specific time.

It is worth remembering that the intensity is the most important variable to analyze, since is with it that the final user can decide if the plasma generated by the reactor has the intensity of the gases in between the limits established for each gas.

Later in this document it will be explained how this step was programmed.

Having only the information stored in a three dimension matrix is not enough to an easy evaluation of the data. The easiest way to analyze data from a matrix is in the graph form. So the program gives the possibility to plot two types of graph: Time vs Intensity for each Wave Length and Wave Length vs Intensity for each sample.

3.2.2: Plotting Graphs

With that, the final user can proceed with the evaluation analyzing the behavior of the intensity from one wave length during the time and how the intensity of all wave lengths behave in each sample.

As said before it is important to verify the uniformity of the plasma generated, by checking the intensity behavior during the time and it is better analyzing the data from a graphic form.

Those graphs are important because as said before the user can verify if some gas appears with more intensity than it should and evaluate in each point of the measurement this is happening. In this step the first two user's requirements were fulfilled.

Also with those graphs it is shown an output array showing the values of wave length where the peaks are, and the values of intensity of this peaks. This is not essential information, but it helps to localize where the values which exceed the mean value of intensity are identifying the peaks.

How these operations were programmed will be explained later in this document.

3.2.3: Exporting the Matrix to MS Excel

The fourth and fifth user's requirements are about exporting data to MS Excel. Later it also be explained how this operation was programmed.

Now it is only important to know that this operation is made by the program and is important for the user for many reasons such as having those graphs and tables for an offline evaluation later, or simply for having the images and tables into a better format to make a presentation or a report.

Exporting the data to MS Excel is not the only way to store the three dimension matrix. The user interface gives the user the possibility to store the data in a SQL based database.

3.2.4: Exporting the Matrix to a SQL based Database

Here in this section is where the new program differs from the previous *Matlab* evaluation program developed in two thousand and ten. While the previous program had no interaction with the database at all the new program is capable to export the data, import and generate graphs and do some operations and export back if necessary.

First it was decide to have just one database containing all the 3D matrixes, each table of the database would represent a 3D matrix. The columns of the tables are: Intensity, Wave_Length and Relative_Time and the table would be named by the date of the measurement.

This occurred because initially the only objective was to have these data available into a SQL based format into a file located on the computer. Just to save it if needed later, as a backup in case that the text files get lost.

Then it came to the conclusion that if it is possible to have the data into a database format, why not work with this format instead of running the text file each time that the user want to get the information? So an interface was built with SQL queries suggested by the user to get the data from the database and procedure with an off line evaluation, without needing to run the program again to get the matrix from the text files.

It comes to the conclusion that this "backup" is not much safe and only the matrix would not be enough to have a proper evaluation. So a new database structure was created and the new database was stored at the AG Plasma database server.

Now it is noteworthy that not only the matrix would be stored at the database, but some important parameters from the measurement such as the data that the measurement was taken, the work station where the plasma was generated, the trial name and the tool used to get the measurement from OES.

So the database structure would be a Description table with those parameters and another table who is connected with the Description table via foreign key containing the 3D matrix.

These parameters are important to save because the user can chose a trial to evaluate from them, does not needing to run the program to get the 3D matrix from the text files again.

Different from the previous database, now the user knows more information about the samples that were collected, enabling him to decide whether the samples collected are good or need another measurement (the reactor may have undergone a maintenance in a certain day and the measurements may have been affected (date parameter), or the tool who collects the date from the OES does not gives the necessary accuracy (tool parameter)). Also allows him to know in which reactor the plasma was generated, if he wants to evaluate the plasma for a specific purpose.

This is also better than working directly with the text files, because in the second one you also can know this information, but you can only get them after running the program and reading the files. With the database it is possible to choose the trial to evaluate using these parameters.

Once the 3D matrix is in the database together with these parameters the user can also generate graphs and make comparisons between two samples via graphic interface.

The implementation of both database solution are showed later in this document, together with the users interfaces.

The main goal of giving the user a friendly interface to evaluate the process via graphs was achieved, but only plots the graphs were not enough to a proper evaluation.

3.2.5: Operations with the data

One of the requirements were to have a possibility to load a calibration file and apply this calibration, due the fact that sometimes the measurement is not reliable due to calibrations errors. This is shown as the red circles at figure 1. Those operations are important for the main objective for different reasons that will be explained in the sequel.

That is why a function was created to read a "pure" spectrum file and do an operation with it adjusting the data around it.

It is also possible to normalize the values of intensity in a defined point. This point can be the mean value but mainly it's normalized into the integration time. This is important to give the user the possibility to compare samples with different integration time, bringing them to the same value.

A reference file is used so it can be seen how close the plasma spectra which is been evaluated is from the reference, considered the right value. This reference file can be any kind of plasma, depending what the user wants to observe.

All these operations change the value of the intensity which is the main variable at the analysis.

These operations are at the graphic interface and the user can choose whether to use them or not. Also if these operations were made using the data from the database not from the matrix, the interface gives the option to export the new data back to the database.

After all these processes and operations done with the data the final user has a friendly interface to manipulate the data and procedure with his evaluation. At this point it has to be clear that what the program does is to prepare the data and provide them in an easy format to read them and take conclusions about it.

After exposing the problem and the solutions to solve it, a chapter showing and explaining the equipments used to measure the plasma spectrum and the tools and softwares used to solve the problems is presented in sequence.

Chapter 4: Tools and Equipment

4.1: Introduction

In this chapter will be presented the equipments used to extract the data from the plasma reactor, the softwares used to solve the engineering problem of provide the user an easy and friendly way to evaluate those data from the measurement, the mains tools of these softwares and why these softwares were chosen to solve the problem.

4.2: Optical Emission Spectroscopy (OES)

"OES is a sensor that allows getting plasma information in a non-intrusively way. It gets the data via an external view port or window not disturbing the plasma environment. Since this tool monitors the plasma external to the reactor, it does not perturb the plasma field or affect the performance of the reactor in any way [SWS97]".

OES allows for real time monitoring of the gases in plasma without interfering with the process itself.

"It examines the optical emission of the plasma discharge and separates the emission into its spectral components [SWS97]".

The main plasma parameter that is observed with OES and using in this project is the intensity over each wave length and this behavior over time.

The OES equipment has a channel to connect a measurement tool; it is this tool that gives the accuracy. This accuracy is related with the parameter Wave Length.

In this project it was used two tools to get the measurements, the USB2G11038 with one digit accuracy giving Wave Length values from 195nm until 875 nm, with 0,2 nm as step, and the other EC080402 with two digits accuracy, giving Wave Length values from 198 nm until 860,98 nm, with 0,02 nm as step.

This is the main equipment of this current project. At the beginning of figure 1 it can be seen its importance as the measurement equipment which provides the data from the analyses. This sensor gives the data in a text files format.

4.3: Matlab Program

The idea of the project starts a few years ago, and a *Matlab* program was developed in two thousand and ten, in order to help this evaluation process.

This *Matlab* program was very useful in order to evaluate some results obtained from the *LabView* program, once the *Matlab* program was approved before.

The first step of the project was to study how this previous program works, studying its code, in order to have an idea of how to proceed for start the new program.

4.4: LabView

LabView was the software chosen to execute all the operations highlighted in green at Figure 1. With this software a program was built to get the text files and organize the variables in a three dimension matrix, plotting graphs from this matrix and exporting these data from the matrix to MS Excel or to a SQL based database.

The idea of writing a *LabView* program starts with the need of integrate de starter plasma-reactor program with the evaluation program, in order to get the data in real time and procedure with the evaluation right on time when the plasma is being generate. Also *LabView* is a powerful tool to work directly with database due its Database Connectivity Toolkit extension and to operate with text files.

As explained before, despite the fact that already had a similar program to do this evaluation, the software chosen was *LabView*. In this section it will be briefly explained some tools of LabView that are not common in other programming languages and that were vastly used in this project.

The advantage of using *LabView* over *Matlab* is that the first one is a better tool to work with data acquisition and data writing. As said before, *Matlab* is more powerful when becomes to mathematic operations with data, but in the case of this program it was not need so much of this property once it has the data read.

4.4.1: Event Structure

The greatest advantage of this structure is to allow synchronization between the diagram and the user's interface.

4.4.2: Cluster

A cluster is a data type just like an array; the difference is that the cluster can store any kind of value, such as Booleans, strings, doubles and array.

Beyond this facility bundling several data elements into clusters eliminates wire clutter on the block diagram and reduces the number of connector pane terminals that subVIs[3] need reducing the memory storage [MAN02].

4.4.3: In Place Element Structure

"This structure is used to increase the software efficiency. Many *LabView* operations require copy and maintain data values in memory, thereby decreasing execution speed and increasing memory usage. The In Place Element structure performs common operations without making multiple copies of the data values in memory [MAN02]".

Beside the facts cited in 4.4 for choosing LabView as the software to write this program, those three structures and tools give better software performance and increase the memory usability of the program justifying even more the use of LabView.

[3]subVI are non-defined LabViewfunctions, built by the programmer, which are used by other majors LabView programs

4.5: Structured Query Language (SQL)

Structured Query Language is a specific language designed for managing data in relational database and the most used database language.

Having as a user’s requirement export the matrix to a SQL based database, SQL is very important in this project because LabView allows write functions such SQL queries, this functions were most used at the first version of the database.

Also to use the Database Connectivity Toolkit, available for LabView, it was important to have knowledge of SQL to make some basic functions.

4.6: Database Connectivity Toolkit

The LabView Database Connectivity Toolkit contains a set of VIs[4] with which you can perform both common database tasks and advanced customized tasks.

This toolkit is an important part of the database step shown at figure 1, because it was with this tool that the final version of the database was built.

Some main features of the Database Connectivity toolkit:

- Maintains a high level of portability. In many cases, you can port an application to another database by changing the connection information you pass to the DB Tools Open Connection VI;
- Permits the use of SQL statements with all supported database systems, even non-SQL systems;
- Creates tables and selects, inserts, updates, and deletes records without using SQL statements.

The advantage of using this toolkit is that it has a few SQL statements ready and easily usable. So a programmer who does not know SQL language can easily write simple functions. Also this toolkit makes the connection between LabView and the database very simple.

[4] VIs are LabViewdefined functions

That is why this extension was important to this current project, to make simple the connection and the SQL statements.

4.7: MS Excel

The MS Excel software was important in this project due the fact that was a user’s requirement to have the option of exporting the graphs and tables in a easy format to analyze them later or simply to build presentations and reports.

This software was chosen because of the facility to export the data from LabView to it and because it is known as good software to build tables and graphs from this tables.

4.8: MS Access

The MS Access software was important at the first version of the database. It was used due its facility to work with; it comes together with the Microsoft Office Package and it is a SQL based platform. This last condition fulfills the last user condition of the graphic interface of exporting the data to a SQL based database, so the first choice was this software.

Chapter 5: The Program

5.1: Introduction

In chapter four it was explained the choice of LabView as software to develop this program now in this section it will be explained how the solutions of the problem presented in chapter three were implemented. In this chapter the code of the program that corresponds to the green highlighted part in the figure 1 will be presented.

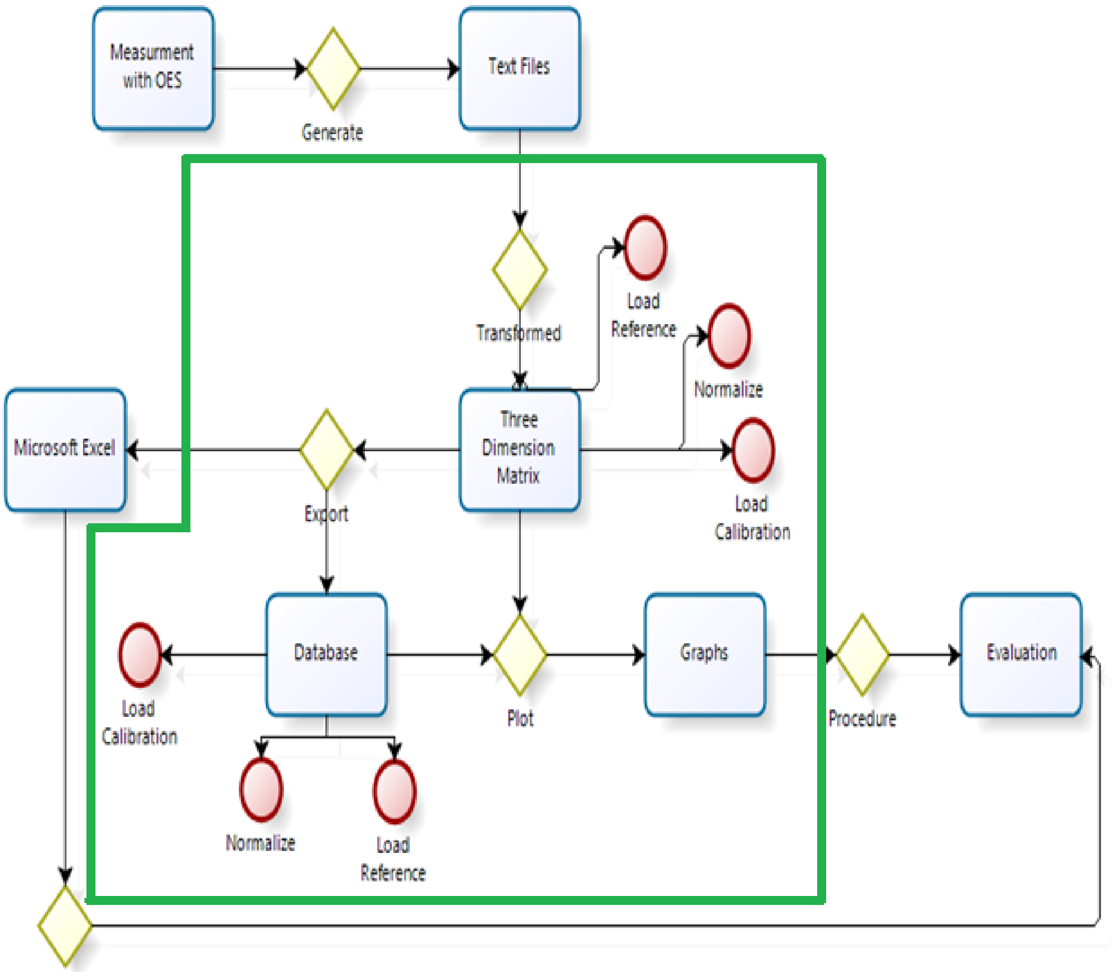

Figure 2 – Data flow until the evaluation

As presented before, the highlighted part of Figure 2 represents the program developed in this project. And here in this chapter, will be showed how these three tasks were implemented and why:

- Transform the text files into a 3D Matrix;
- Export the data from the 3D Matrix to MS Excel;
- Plot Graphs from the 3D Matrix.

Another chapter specific for the database will follow this one because there are some SQL and database details to be explained deeply.

Also in this chapter it will be explained the three main operations that can be done with the intensity value of the matrix.

5.2: The Program

This program was built in *LabView* and it gets all the data acquired from Optical Emission Spectroscopy (OES) during a low pressure plasma process, which are written into text files, and organize them into a three dimensional matrix (composed by time (ms), wave length (nm) and intensity (-)) for later to be easy export the data to a SQL based database and to MS Excel.

The data are separated by files which each one indicates a time interval.

Also this program has to provide an interface easy to work with in order to proceed with the analysis.

The requirements to the user interface were:

1. Graphical display of intensity vs. time at a specified wavelength
2. Graphical display of intensity vs. wavelength at a specified time
3. The possibility to load calibration files and apply a calibration
4. Output of certain wavelengths defined by the user to MS Excel
5. Output of certain wavelengths that are characteristic for commonly used plasmas to MS Excel

5.3: Transforming Text Files in a Three Dimension Matrix

To organize all the files into a matrix 3D it was made a specific subVI called "get_data".

This is the part of the program which gets the text files from the OES and after does some treatments with them and then transforms them into a three dimension matrix.

To understand how this subVI works it is important to have in mind a few things:

1. When the OES sensor generate the text files these files are stores into a folder defined by the operator of the sensor.
2. There are two important informations inside the text files: The Wave Length of the gases and the intensity of each wave length.
3. The Wave Length and the Intensity are separated by a semicolon in the text files.
4. Each file represents one time interval, but this time it is recorded as Absolut time.

Organize the data into a better format and easily to work with is very important for the final objective. It is simpler to plot graphs and export data from an organized matrix than from text files. Is from this matrix that all the other operations can be done.

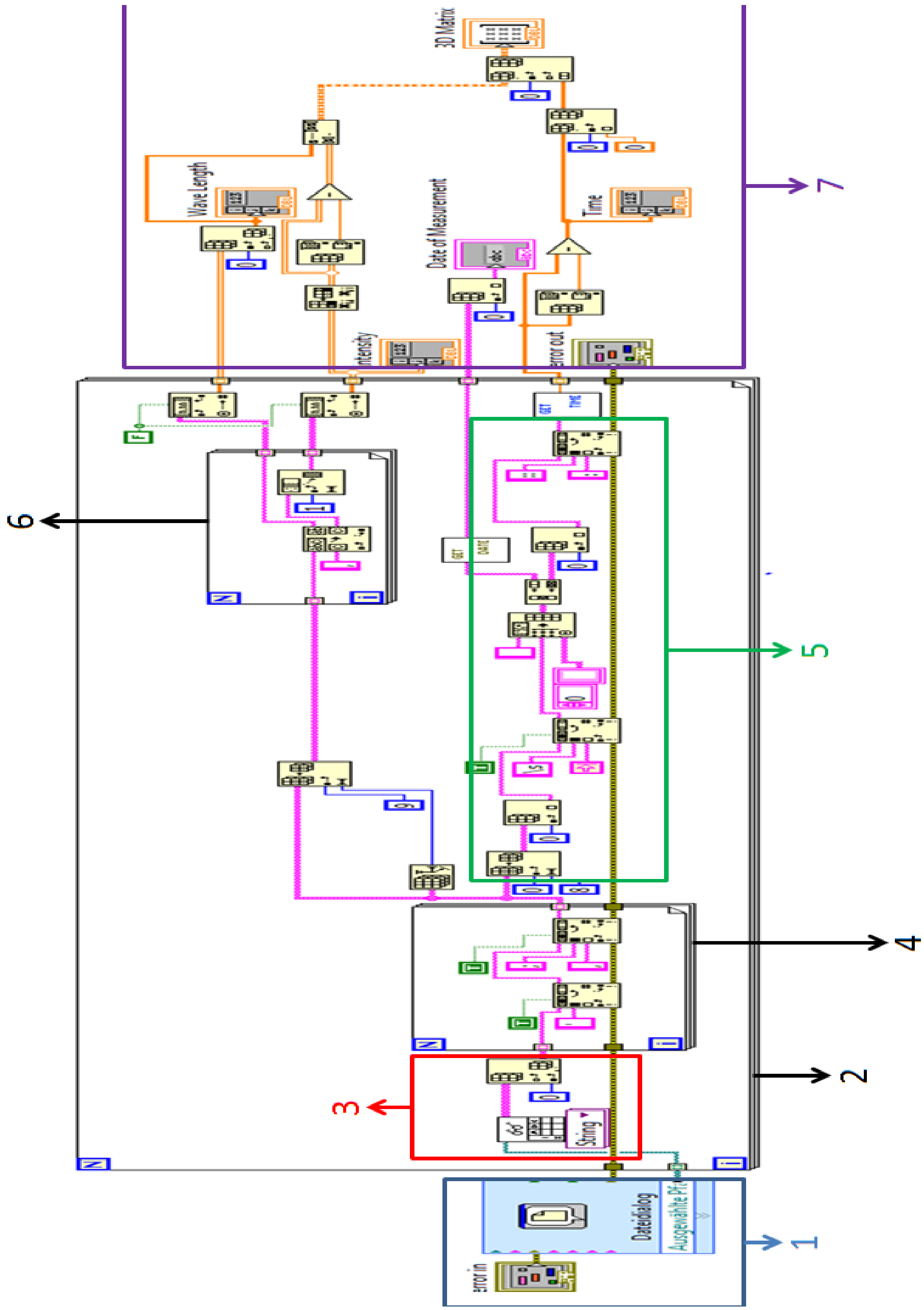

Figure 3 – SubVI get_data, which converts the text files into a 3D matrix

When running, the subVI opens all the files selected from the folder that the measurement was saved. This is shown in the part 1 at the figure 3.

The for loop pointed as number two of figure 3 shows that the operations contained inside the loop will occur for all the files in the folder.

The part pointed as 3 at the figure 3 transform all the content of the files into a table of strings using the space values inside the files as a tabulator for the table, and each line of the file represents one row at the table. This happens because OES generate .txt files, with headers and without tabulation between the values, so the whole file is treated as string.

The loop marked as number 4, indicates that those operations will occur for each line of this table.

The first nine lines are headers, so from these nine lines is just necessary to get the time and the date information. The header of the file is treated in the section pointed as number five; the date is gotten and saved in a string variable.

To build the three dimension matrix, the time should be in relative values, with means the first time being zero and the next ones being the time the measurement was taken after the first. So another subVI was created to convert the Absolut time in hours, minutes, seconds and milliseconds in relative time only represented by milliseconds.

The for loop, shown as number six, separates the wave length values from the intensity values. As said before those values are separated in the text file by a semicolon, so now this step just put the first value of the semicolon (the wave length value) in an array and the second one in another.

Now this operation will be repeated for each file with the loop marked as number two. The results of this loop are three arrays, each one with the three main variables explained before. Now in the step marked as number seven only puts these variables together in a matrix.

In some samples the wave length was divided in decimal points and in others in centesimal. When the samples are in centesimal digit the size of the matrix becomes too big (in order to thirty thousand rows). This happens due to the OES tool used to take the measurements.

The software *Labview* does not have all the mathematical power found in *Matlab*, but despite of taking more time to read the files, it can be done.

The results in this stage were perfect; all the files generated by OES and tested here were organized properly. The matrix was generated correctly with the corresponding relative time, and the correctly date information.

It has to be said that when the wave lengths have centesimal digits the program takes three times as long to read the data and generate the matrix as when the wave lengths just have the decimal digits.

This is not a problem, because despite the fact of the number of rows exceed thirty thousand, the program can read the files and build the matrix without a problem, but it would take a lot of time, so once these kind of files were read it is important to save them in a database, because as will be explained later, reading the data from the database is faster than from the text files.

But this is not the final version of the matrix, before the program save this matrix in a variable; it has to pass by another operation.

5.3.1: Defining Limits

It is important for the user to define a minimum value of the intensity that he wants to see. Depending of the samples extracted and the type of evaluation the user wants to do there are some information about the intensity of some samples that are not needed. For example, if is known that it is normal for a certain type of plasma have for all gases the intensity value until 20 – (except for the gas that was used for generate the plasma), it is only important to evaluate the results from the gases with intensity over 20 to know each gas exceeds this value and check why this is happening and in which point.

This limit reduces the size of the matrix making the program run faster.

In a simple way, a pop up window shows up with a graph of the average intensity versus the time and its possible to choose the minimum value of intensity that is interesting to the process of evaluation of the specific plasma. In some cases the first measurements files do not contain sufficient information for different reasons, as so the plasma-reactor was in the beginning of plasma production, the sensor was not in the right position when the measurement starts and so on.

The task of the subVI is, in possession of the value that was inserting in the field; search in the matrix where are the points (indexes) that the average intensity is higher than this value. With these indexes it is possible to create a new matrix, and the size of this matrix will be defined by the first and the last indexes. It looks like a simple task but it is essential in a process of evaluation data.

This first step of the program just shows the graphs of the average intensity (-) over time (ms). With this graph plotted the appraiser can select the region of his interest by choosing a minimum value of intensity.

When the user chooses a minimum value of intensity, another graph showing the chosen time interval, appears and the user can see if that window is enough and procedure with the evaluation.

As following the subVI that does this important operation and the results graphs.

First two arrays are created, one of the indexes from the values of average intensity array which are bigger than the inserted minimum value and another Boolean array with the results of this comparison.

Then these two arrays have to be verified. First the program searches for true values in the Boolean array. If all the values are less than the inserted minimum value than the Boolean array will have only false values, so the subVI interrupts its execution and the program keeps the matrix without any change.

In case of the Boolean array do not be entirely false, the part two searches for the index of the first value true (which value from the average intensity is the first value bigger than the minimum value) and the last value true (this searches is made using the indexes array, getting the maximum value of the array).

The case if is true just plots a graph of the average intensity using the first and the last indexes found previously.

After this process the user can decide if the limits are good, if they do not, the process can be repeated as many times as needed.

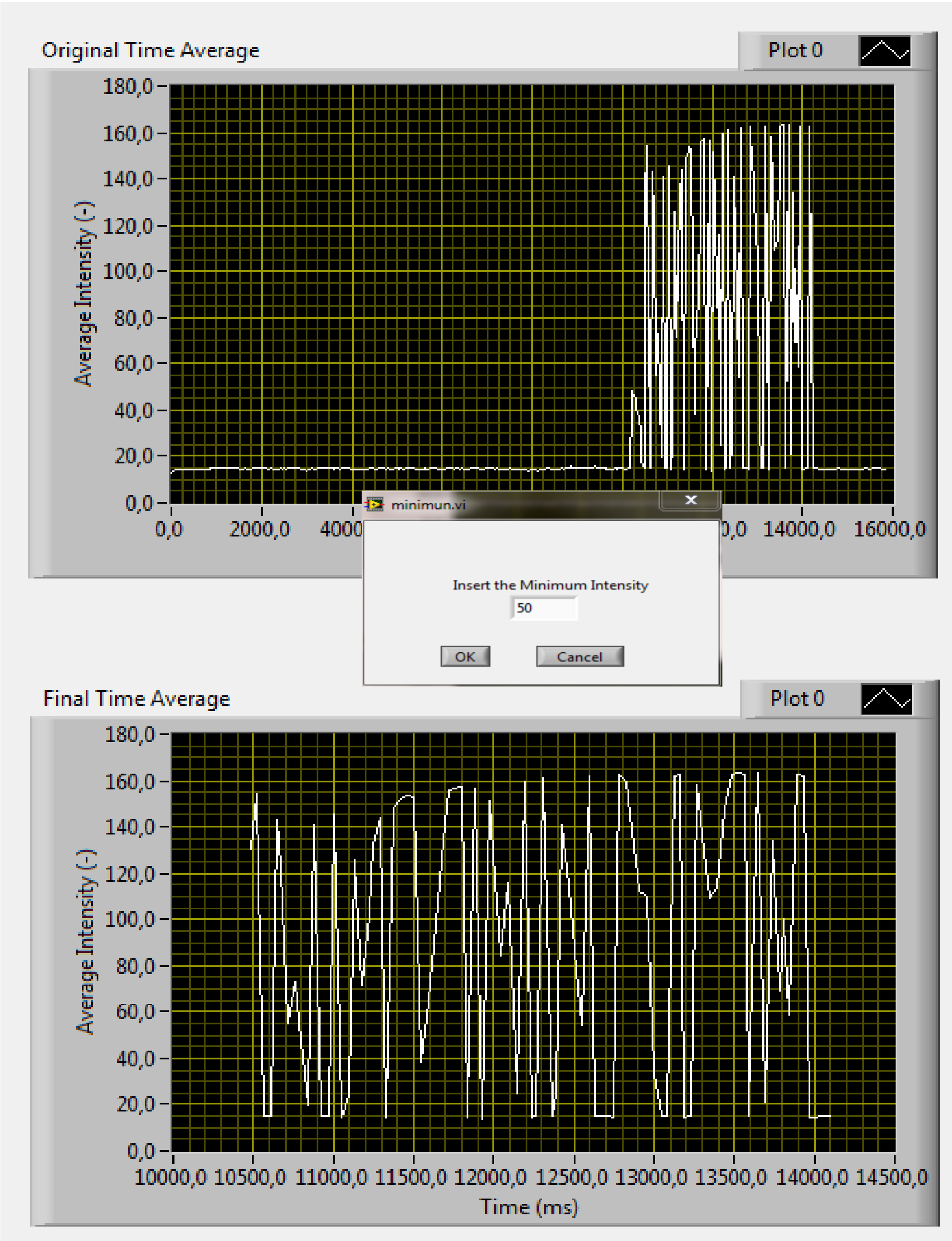

Figure 4 (a) – Original graph, (b) Graph after chosen a minimum intensity

The figure 4 top shows the original graph of average intensity versus time with all the samples extracted from OES. With this information it is possible to choose the samples that will be evaluating. In this case it was chosen the minimum intensity of fifty (-), resulting in a graph showed at figure 6bottom.

Now the three dimension matrix is ready for being work with. So it is time for the next step of the figure 2, export the data from the matrix to MS Excel.

5.4: Export Graphs and Data

The graphs are important tools to the evaluation process, by the fact that they are visual representation of the data table. It is much simpler to analyze a graph than a three thousand rows table. But both are important to export in order to give the user the opportunity to have a better look on the data, in an *off line e*valuation. In possession of the data and the matrix it is possible to make reports, presentations and so on. All the data are exported to *MS Excel*.

The essential data to export it is the behavior of the intensity for each wave length. For each type of plasma it is expected a certain intensity of gasses, so with the data provide a comparison can be made.

In order to proceed with the exportation there are two buttons and a cluster that are used on this step.

The buttons Export Graph and Export Data cancel each other out. When one button is pressed the other one is set false, in order to execute one operation at the time and does not happen conflicts and overload of memory.

To export the data, the program offers a possibility to choose a lot of parameters to export. To make this functional it is used a cluster. Different from an array that only store one type of variable, the cluster can store any type of variable. In this program case the cluster has Booleans, doubles and an array of clusters.

The figure bellow shows the structure of the cluster.

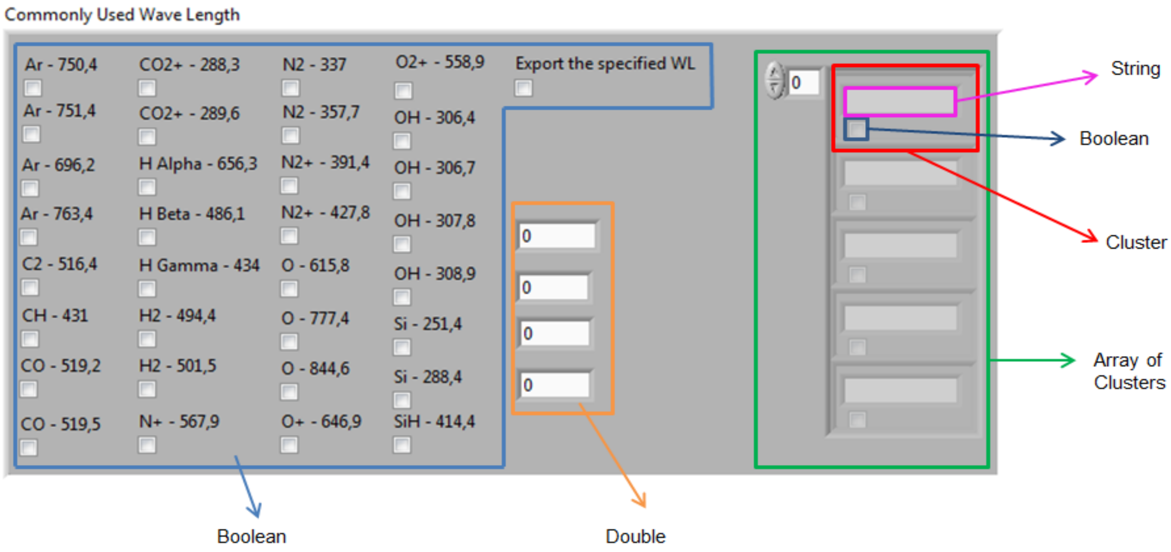

Figure 5 – Cluster structure

A cluster can be only control (same idea of inputs) or indicator (outputs), it cannot be both.

When an element is added in a cluster it is placed as element zero (0), the second is element one (1), and so on. If you delete an element, the order adjusts automatically, this order is the order that the element appears at the Bundle and Unbundle terminals on the block diagram.

The Boolean values of the cluster are the mostly commonly used wave length in the low pressure plasma process, the fifth column is the Boolean value for export user's specific data and graphs.

The "Export the specified WL" gets the closest value of wave length inserted by the user and exports the data or graph for this wave length. Later it will be shown that there are other two ways to export users specified wave length.

In the double values camp the user is able to insert values of wave length that are needed and are not in the most commonly used wave length. As said before, this is another way to export data specified by the user. As in every part of the program that requires user interact, the program searches for the closest value of the value inserted. In this case, differently of the case explained before, if the wave length value is off the limits of the values measured, a pop up shows up asking the user to put a valid value of wave length. If there is a valid value on this camp the program will export this data, if it’s not, nothing will happen.

The array of cluster is an alternative for the user to insert more values that he needs. In this array the user can insert as much values as he wants, as long as it is a valid one. There is a special field to insert them and also a button to clear the array. The difference here it is the program will just export the data marked as true in the Boolean field of the clusters inside the array.

It was said that there is three ways to export the same value of wave length; it is not a hundred percent true, because no matter how many times the same value is inserted or where it is put, the program will export just one time.

There is a subVI which export the table and another to export the graphs. If there is any Boolean true or double inside the limits the program will open an Excel file and then it will check which one of the cluster’s input is true (it can be more than one) to procedure with the exportation. Each value corresponds as one input of the cluster, so there is an index for each input.

The groups of variables are treated separately. The checkboxes matches a previously known wave lengths, so the subVI already has the label of the value just needing the index value to check if it is true or false. At the double field the program does not know what will be written until the button Export be pressed, so then it is taken this value and turn it into a label, the same thing happens with the array.

All of these operations happen inside the In Place Element structure, which is a structure that is used to manage memory in the program.

The structure inside the main program is the same for export data and export graph. In fact the structures inside the subVIs are very similar, changing a few things.

Differently of the Boolean values, the double values and the array of clusters need to pass an extra evaluation, before being exported.

For each value it has to be checked two things: First it has to be checked if the value is in between the minimum and the maximum value of wave length. Second, it is verified if the value typed is equal the value inserted at specified WL field. If this second verification is true, the program checks if the Boolean value of specified WL is true or false. If it is true, then nothing will happen, in case false then the program will call the subVI to export. The same thing happens in the array section.

It can be noticed in this picture that it is been add thirty-three at the index (thirty-seven at the array section). This happens because the subVI is the same for the whole cluster. So the first 33 values (from zero to thirty-two) are for the Boolean values. From thirty-three to thirty-six are for the double values, and after this is for array.

5.4.1: SubVI Export

The inputs of the SubVI are: date, index, wavelength, report and the 3D matrix.

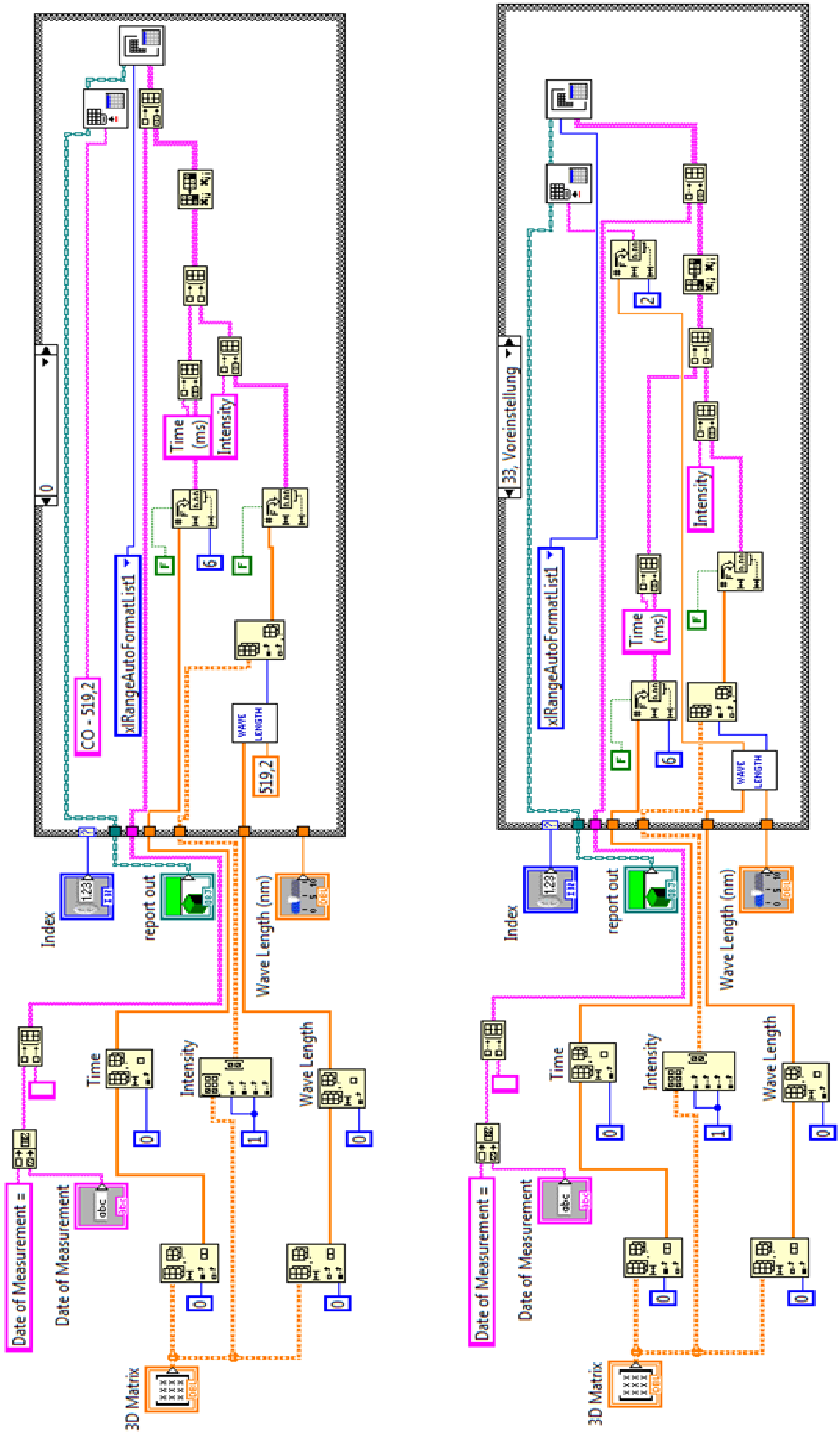

Figure 6 – (a) subVI export for Booleans values, (b) subVI export for double values and array

The figure above shows the difference between the Boolean and the double/array. The first one, from the Boolean values, each index has its own value. In figure 6 (bottom) there is an extra double to string transformation to get the value typed and transform it in the name of worksheet. It can be noticed that in this subVI the table is exported to MS Excel as strings. It was choose this way, because it was easier to concatenate the header with the data.

The results in this stage were satisfactory, all the tables and graphs were exported properly.

Of course the final time to export the data changes with the amount of data exporting, but if all the options from the cluster were selected to export, the final time to conclude this operation was acceptable.

5.4.2: Add Wave Length

In order to have more possibilities to export data, an array without memory limit was created inside the cluster. It was already explained how it works to export. In this subsection will be explained how new values are added in this array.

The user is able to put any value at pleases him in the add wave length field, but the program will check if this value is in between the minimum and the maximum wave length to insert it in the array.

Also the program is capable to fill empty spaces inside the array, and search in it if the number inserted already exists do not inserting it.

When the button is pressed, it is verified if there is something in the array. If true the program first check if the value inserted is a valid value. After, it search in the array if there is already the same number, if there is, nothing will happen. In negative case a search will be made in the array looking for empty spaces, if there is the value is inserted at this empty space, if there is not, and then the value will be put at the end of the array.

In case of there is nothing in the array, it is just checked if the value inserted is a valid number. If it is, then the number is put in the array.

5.4.3: Clear Wave Length Array

If the user wants to do a new exportation of data and do not need the values inside the array he can unmark the Boolean values or clear the whole array. Clearing the array is a better option to reduce the memory usage. This step was easily made just unbundling an empty cluster in the array.

5.5: Plot Graphs

All the important variables of the process might be obtained and displayed by the interface, since the results from the whole process depends on them. In this case the most important variable is Intensity and its behavior in time, also its variation according to the wave lengths.

To plot the graphs, the matrix had been separated in two one dimension arrays, one contained the values of wave length and one contained the relative time. The matrix remaining it is the matrix just with the intensity values. As explained before, *LabView* it is a very graphical tool and one of its best qualities is make it easy to work with graph plots.

The whole matrix is stored in a variable (keeping in mind that the LabView variables work different than the others programming languages variables). So anytime the double value inputs “Insert Wave Length” or “Insert relative time” changes their value, the code active an event structure, which is a piece of code that answer to stimulus from the front panel, and with the new value plots a new graph, getting the index in the array for the wave length or the time and searching the respective value at intensity matrix.

Also in this interface it is shown an output array showing the values of wave length where the peaks are, and the values of intensity of this peaks. This is not essential information, but it helps to confirm if the plasma was produced properly or there were some gases that were not supposed to be there.

5.6: Normalize

Labview has a tool specifically to normalize matrix into the mean value, so when this button is true, the subVI gets the matrix and simply give back the normalized matrix. It is a simple task, but it helps in some evaluation process.

Also a linear approximation was created to normalize the intensity values to the Integration time of the calibration files.

This is important when the calibration is applied because usually the integration time of the samples is different from the calibration files, so it is important to normalize all the values at one same point to have the calibration valid.

5.7: Reference File

The reference files are files just like the samples that will be evaluated on the program, an endpoint.

A reference set of spectra may be obtained by recording the emission of the cleaning gasses without any sample present and then compared to the emission observed when the contaminated samples are present.

There are two operations that can be done with the reference files. You can divide the whole matrix by the reference files or subtract the matrix by them.

The reference file option is used for the user to compare the sample, of plasma generated by a gas that he is analyzing with another sample with the patter values of intensity for this kind of plasma generated by the same gas.

This is used to verify how close the sample that is being analyzed is from the patter value.

There are two operations that can be done with the reference files. You can divide the whole matrix by the reference files or subtract the matrix by them. Divide by the reference.

When the user divides by reference he is looking for the number one as result for the proximity, when he subtract he is looking for zero as the value.

5.8: Load Calibration Files

The load calibration option applies the operation used in other plasmas diagnostics softwares. Using a “pure” spectrum as the calibration the program can adjust the value of the intensity of the measurement files using this operation.

As said before, in some cases the measurement is not perfect, so it is necessary to load files known as calibration files, to fix this problem. Those calibration files are text files extracted from OES (as same as the measurement files), but they are gotten using a “pure” plasma with a specific gas, giving the “pure” value of intensity for each wave length.

One in possession of those files the program gives the possibility to load them and the operation that is applied is multiply the intensity of the samples by its respective “pure” value squared. In other words: The data with calibration will be equal of the data without calibration times the pure value squared.

5.9: Interface

Now that the tasks of the program were explained it is time to show the interface which commands all of these operations.

The interface contains all the user’s requirements and a few more options. As the main part of the project the interface stays in a while loop running the program, and just stop running when the “Stop” button is pressed.

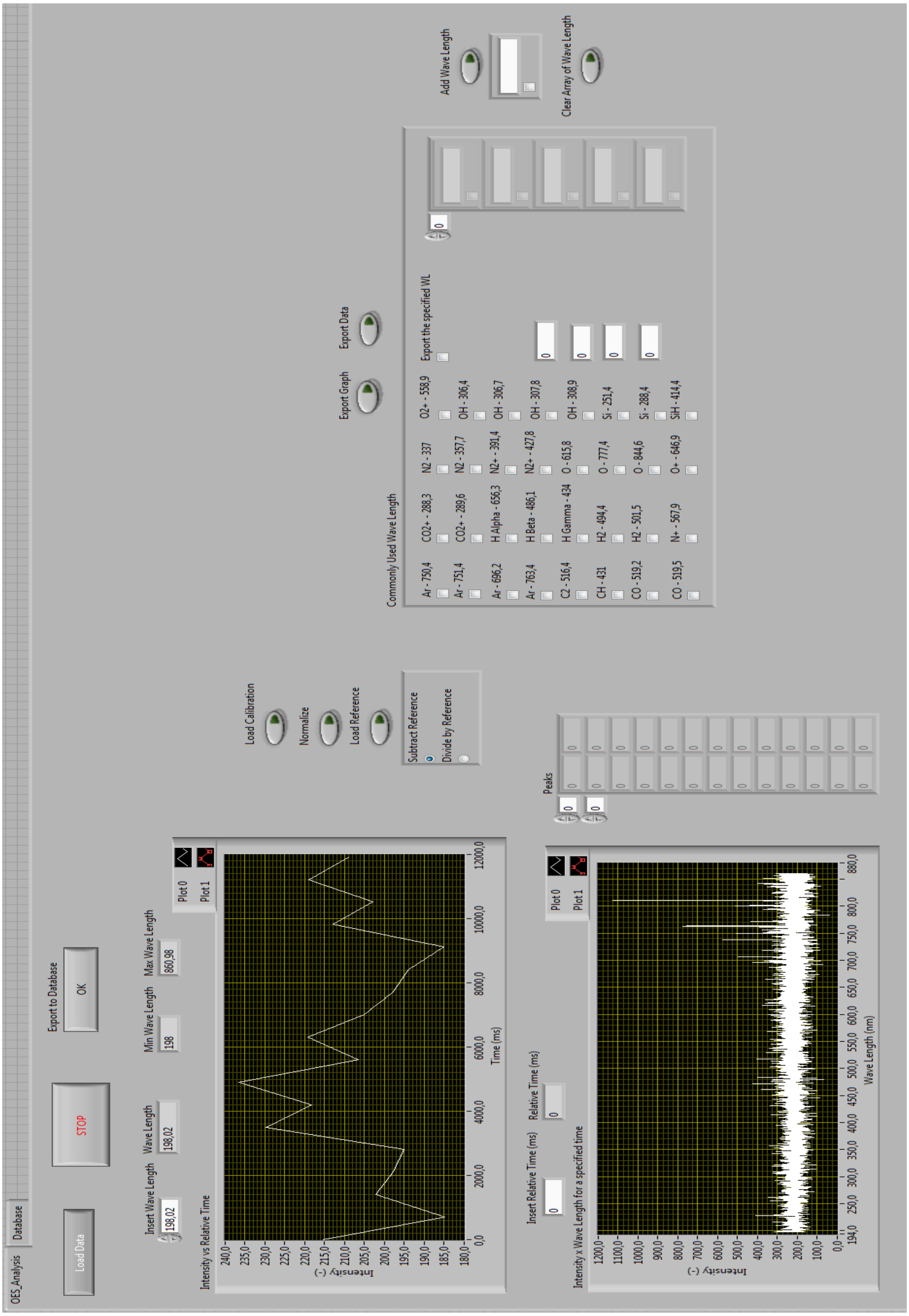

Figure 7 – User's Interface

5.9.1: Interface Tabs

As can be notice at figure 7, the user's interface has two tabs. The first one (that is being showed at the figure) concerns about the OES evaluation program. With it, it's possible to load the data, export to a database, export the graphs and the tables to a Microsoft Excel file, do some important operations with the data or just see it in the graph form.

The second tab is to treat the database option, reading the data directly from the database not from the OES files anymore, this interface will be showed later in this document.

To represent the value changes of the buttons at the interface the event structure was used. This structure responds of each value change at the user's interface and runs the appropriated part of diagram. It can be seen as an interruption in the written languages, waiting for the signal to execute. In all the cases of the project the signal is a value change.

All the operations shown in this chapter can be controlled via user interface.

5.10: Conclusion

The main goal of this program was achieved; a simple and very easy to use interface was developed with all the tools working properly. The user now can proceed with the evaluation using the data as better suits him.

The execution time is satisfactory and the results obtained met all the design requirements.

One of the advantages at the development of this program was that the results could be compared with the *Matlab* program that was developed before, to check if the matrix was generated correctly; the resulting graphs are also correct and to verify their reliability without checking all the text files one by one to find out if the values corresponding are correct.

In this chapter was presented the program which resolves the problem presented at picture 1, Also its interface that controls the operations and show the data, were presented with the first five user's requirements.

In the next two chapters it will be explained how the last user's requirement, Export of the data matrix into a SQL-based database, was developed and programmed. In the next chapter, it will be presented the first version of the database which fulfills the requirement but has some limitations. After this, another chapter will explain the second and final version of the database and how those limitations from the first version were solved.

Chapter 6: Database

6.1: Introduction

Another requirement for the program it was to export of the data matrix into a SQL-based database. This requirement is treated in a different chapter because explain a little about database, how the connection between *LabView* and *Microsoft Access* Database was made and how the process parameter were imported.

At the beginning it was just to have a backup version of the three dimension matrix generated from the text file, but then it was decided to work with the data from the database in an off line evaluation, making relevant queries. Reading data from database was faster than getting them from the text files directly, so it was an intelligent solution.

6.2: SQL based Database

Database consists of an organized collection of data. Most modern Database Management Systems store data in tables. The tables are organized into records, also known as rows, and fields, also known as columns. Every table in a database and every field must have a unique name. The records must have a unique identification known as keys.

6.2.1: Data Manipulation Language

As the name says those SQL keywords are used to manipulate data in the database. Four main tasks can be done such as INSERT values and inputs into the table; SELECT specified values from a table; UPDATE the table adding columns and entrances; DELETE values from a table.

The SELECT operator can be classified as a Data Query Language once it only makes queries and does not chance any value in the table.

To make the select operation come true is also necessary to add a clause to the query.

6.2.2: Data Definition Language

Those keywords have the task to define the data, allowing creating new tables and associate elements. There some basic commands but the only one from this category used was CREATE.

6.3: Creating the Database

There are a few ways to create this connection between the software and the database. The one that was chosen was to create a connection and a database when the user requires at the main interface.

The subVI that does this is showed in the figure below:

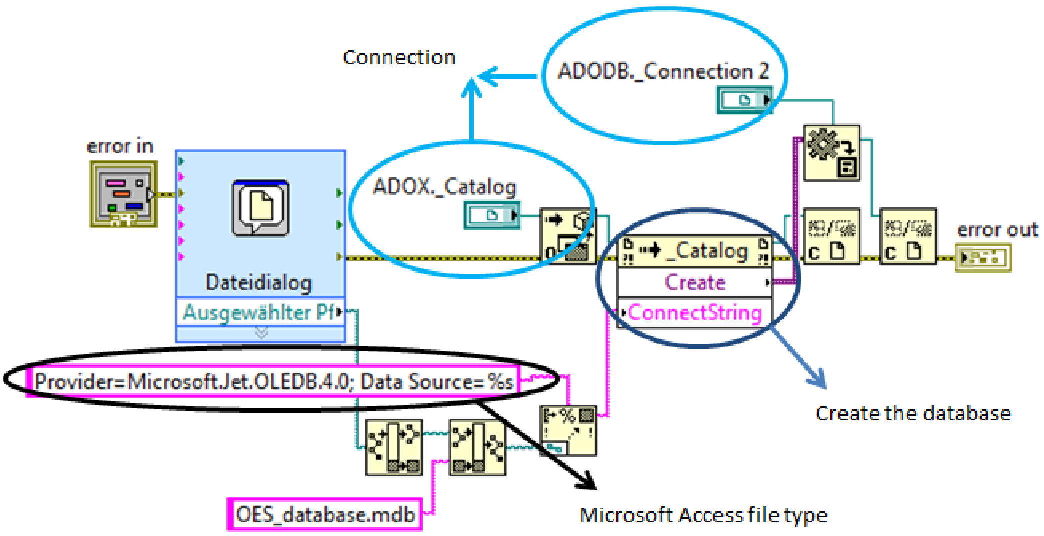

Figure 8 – Creating connection and a database

This subVI create a file with mdb extension (database file from Microsoft Access) called OES_database.mdb, with the Microsoft.Jet.OLEDB.4.0 type, showed in black in the picture, which is the *LabView* definition for a connection between *LabView* and *Microsoft Access*.

Those connection and file just have to be done once, because all the tables will be in the same database. It was made this way in order to give the user the possibility to create new databases if it is needed. To create a new database it is just necessary to change the name of the file with mdb extension.

This just creates a database from LabView in a Microsoft Access file.

6.4: Creating and Filling tables

Once the database was created it is time to create the tables and fill in them. Two different subVI were creating to do those operations. Both of them use the database created as explained before.

It was decided to do just one table for each group of samples naming the table as the date of the measurement, here it was assumed that only one valid measure is taken for day, so each table in the database would represent a 3D matrix generated by the program.

This decision was made thinking in making the queries simpler despite of having a huge table and the long time that takes to fill the table completely. But once the table is ready, is much easy to do the operations with it.

So to make the queries the user would need just to enter the name of the table that he wanted to get the data from, and then he would have the 3D matrix in a table format.

The columns of the tables are Wave_Length, Relative_Time and Intensity, having Wave_Length and Relative_Time as primary keys. This happens because the choice of using just one table and the database rule says that for each tuple it has to have a unique identification so Wave_Length or Relative_Time could not be primary keys alone because their values repeat in the table.

These two entrances combined give just one element in the 3D matrix, so the obvious though in order to create a valid and unique table it was that those entrances should be the composite primary key.

Figure 9 shows how the program creates a table in the database using LabView and SQL statements.

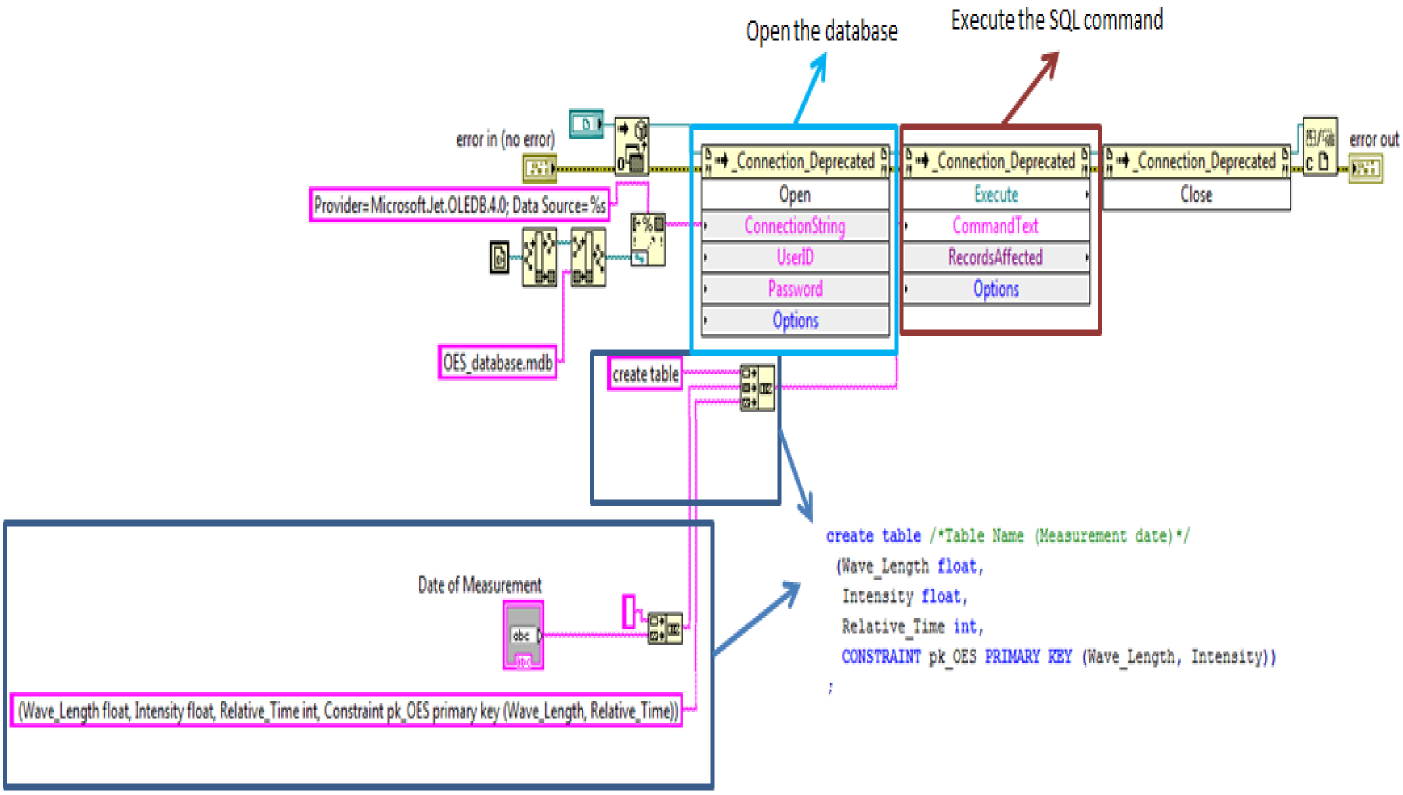

Figure 9 – Creating table into a database and the SQL command

It starts by opening the connection with the database, in the same way as did before to create the database, and specifying the name of the database (the name must to be an existing database name with mdb extension). This is showed at the non highlighted part of the picture.

With this connection opened the blue part opens the database file, and the brown highlighted part executes a SQL statement in this opened database, after this statement executed the connection is closed.

The part highlighted in dark blue is the LabView code for an SQL statement to create the table that has to be passed as a String. The Date of Measurement parameter is passed by the subVI.

So after running this subVI a table named as the date of measurement, with the three variables (wave length, relative time and intensity) of the three dimension matrix was created into the database and now is ready to be filled with the data from the matrix.

The subVI that fills up the table also starts opening the database and making the connection between the *LabView* and the *Microsoft Access* database.

After that, in*for loop* the program executes the insert operation for each single value of the matrix. The matrix is passed as input and divided in the three variables. For each wave length and for each relative time a row is added in the table with the respective intensity value.

It has to be mentioned that the double values from the matrix have being transformed into string values in order to fix in the command field, which only accepts strings. But in the query those values were changed again for their original form.

The highlighted part in dark blue shows this conversion from double to string and the SQL statement of insert data in the table.

The SQL statement is:

insert into /*Name of the table*/ (Wave_Length, Intensity, Relative_Time) values (..,..,..)

These values are each element of the 3D matrix, that is why the for loop is needed. This is a recursive way, but unfortunately the only way to export data from LabView to a database is passing row by row.

The fact of inserting one value of the 3D matrix at the time makes the transition to the database take a lot of time, but does not use much memory and has to be done just once for each group of files.

At the end of the fill operation the subVI closes the connection with the database. It is important to notice that the connection is only available when it is really needed.

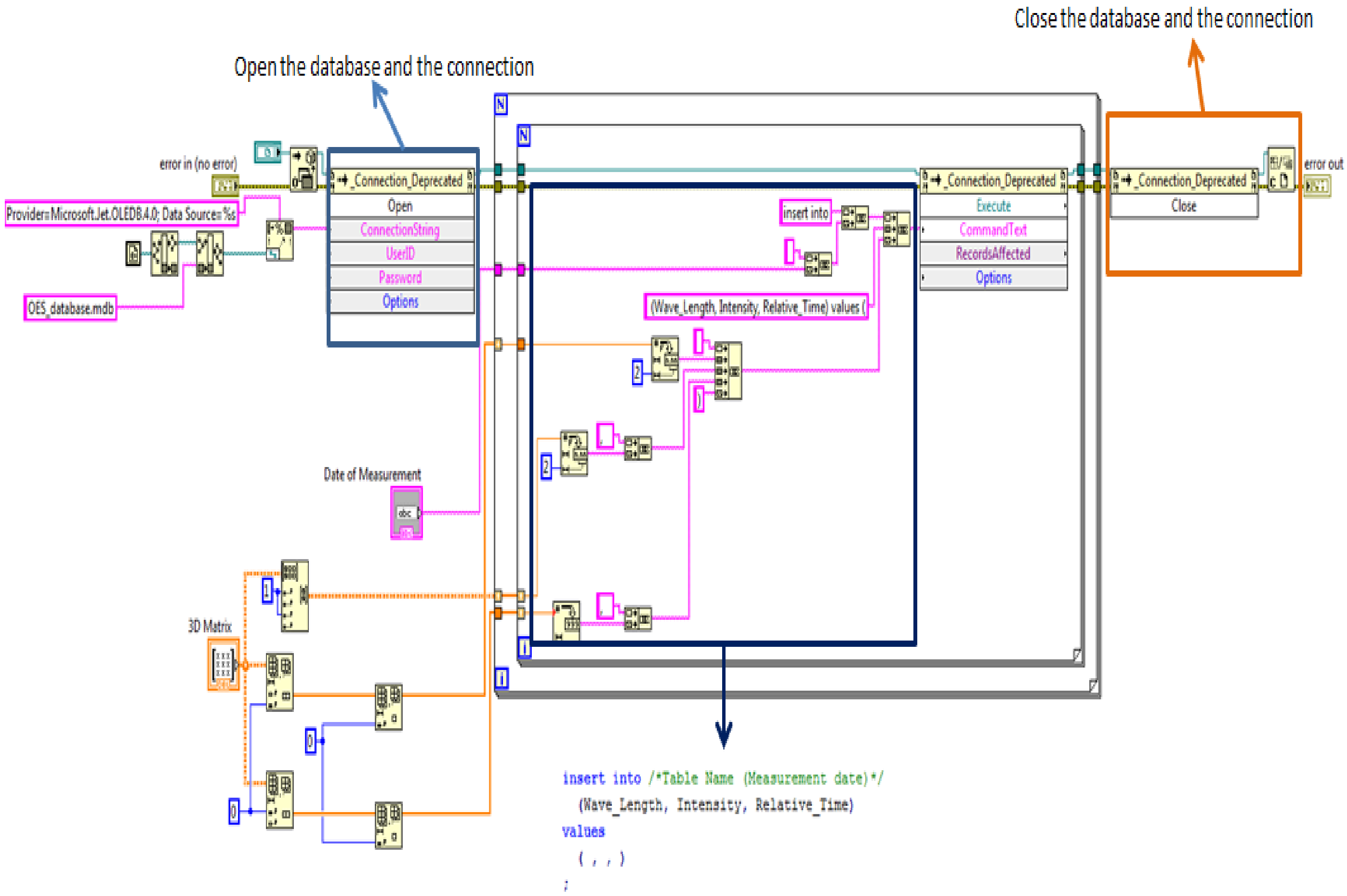

Figure 10 – Fill up table and SQL command

6.5: Query Interface

Considering that the final user of the whole program does not know *LabView* programming and SQL language a simple interface with some basic queries was developed.

The huge advantage of the use of this interface is the *off line* evaluation. Which means that the data is already read, compiled and stored in a database, so it is not necessary to run the whole program again in order to procedure with an evaluation.

It was asked for the user the main queries and the most used in order to implement this facilities, basically the queries exposes in a table form what the main interface shows as graphs.

This interface also has a SQL field to do others queries if the user feels necessary and knows SQL language.

The interface is show in the picture bellow:

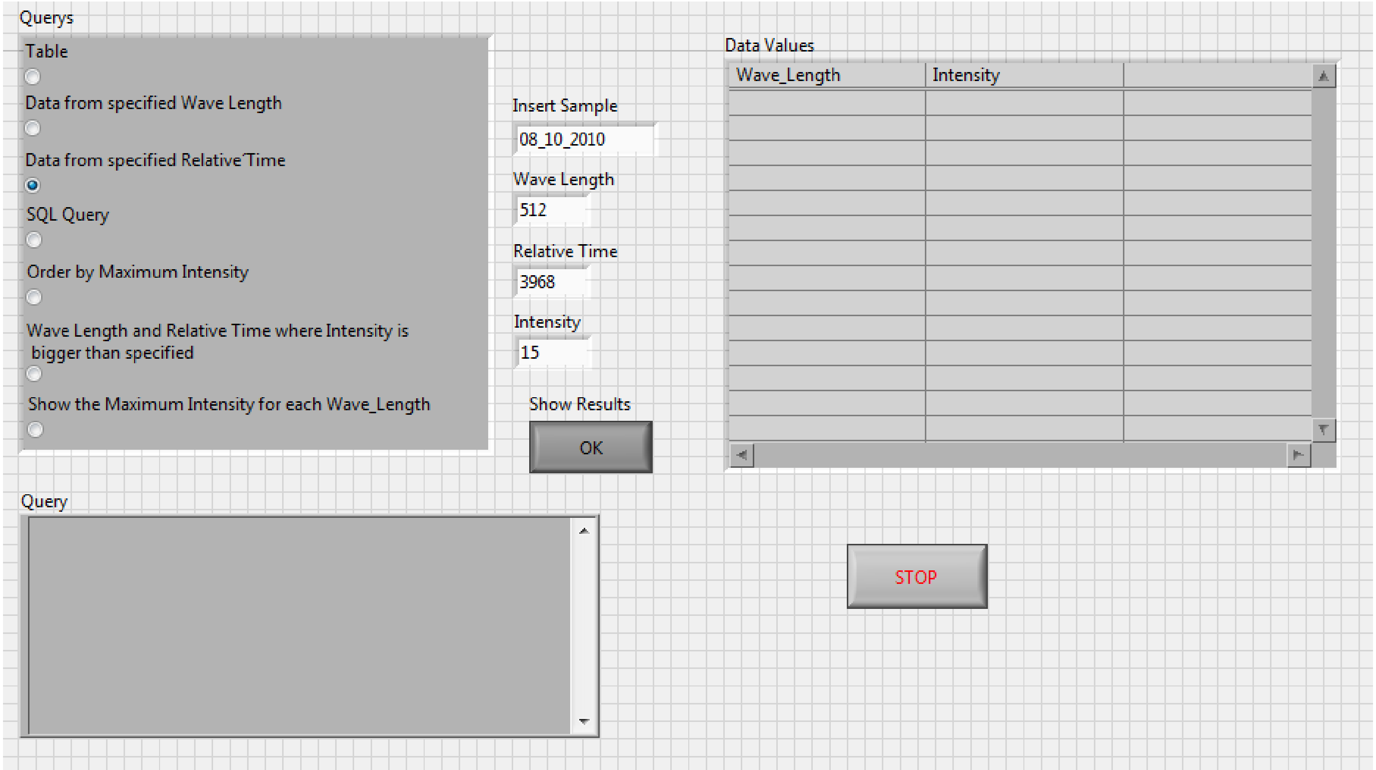

Figure 11 – Database User's Interface

The SQL queries used to develop this interface are shown below:

- Show all the table:

Select * from /* Table Name */

- Data from specified Wave Length:

Select Intensity, Relative_Time

From /* Table Name */

Where Wave_Length = /* User's specified Wave_Length */

- Data from specified Relative Time:

Select Intensity, Wave_Length

From /* Table Name */

Where Relative_Time = /* User's specified Relative_Time */

- Order by Maximum Intensity:

Select *

From /* Table Name */

Order by 1/Intensity

- Table where Intensity is higher than specified value:

Select *

From /* Table Name */

Where Intensity > /* User's specified Intensity */

- Shows the Maximum value of Intensity for each Wave Length:

Select MAX(Intensity). Wave_Length

From /* Table Name */

Group by Wave_Length

This table format is not the best format for the user evaluates the process, but can give some ideas about it. These queries show the most important information for the user. Such as know the intensity of a specific wave length during the time, to know how the gas with that wave length behaves.

The user also can specify a highest value to the intensity and verify if any gas exceeds this value, this is important to know which gas is not acting according it should. It can be seen when previous tests were run and the user already have the right value for the intensity of the gases.

Also together with the information explained in the previous paragraph there is a query to inform the maximum value of intensity for each wave length. This value can be compared with the "right" value found at the reference file to get the user an idea if the gas is closer or not of this value.

6.6: Conclusion

Basically all the subVIs, for the database section, opens a connection between *LabView* and the database tool *Microsoft Access,* and do some operation with the database (create table, fill table and consult from table and so on).

Despite of the advantage of the *off line* evaluation, the database takes too long to fill its table automatically due to the quantity of data and the fact that the program adds row by row executing an SQL code for each one at the time.

But this operation has to be done only one time, so this big time would made up in an “offline” evaluation, because you no more need to rescue the data from the text file (cannot be remember where it is or sometimes it was deleted) and it is faster read the data from the database than execute all the program again with the text files.

At the end this database program was discarded. It was concluded that only the 3D matrix would not be enough information to do a full evaluation, because the user would not have any information about the process and the means that the measurement was taken. He would have only the 3D matrix to evaluate without knowing what kind of process he is evaluating and with each tool the measurement was taken. Also for the new database an extra column was thought to save the modification on the intensity due to the operations that can be done with the data.

. Also this database created into Microsoft Access was located in the machine not in the server, which would difficult an integration with other process parameters.

The next chapter will explain the new database structure, explaining how this new structure is better than the one presented in this chapter and how it solves the storage and the read facility problem.

Chapter 7: Database Alterations

7.1: Introduction

This chapter will approach the new idea of database, in order to improve the restrictions from the previous database once decided that the database type, place and the information were not sufficient another proposal was created.

In first place this new database is better than the previous because is located at the AG Plasma server which gives the back up more security and stability.

About the structure, the new one does not have one table for each group of sample, but it has one table for each plasma reactor, plus a description table with data from the measurement. So every measurement taken in a work station goes to the same table, but with its parameters in a separated table linked by a foreign key.

Using not only the OES text files but also de monitoring file it was created this new database with those four tables:

Description (<u>ID</u>, Trial_Name, Work_Station, Tool, Date_of_Measurement)

Boot (<u>ID_description (FK), Wave_Length, Absolut_Time</u>, Intensity, Intensity_Modified)

Kiaba (<u>ID_description (FK), Wave_Length, Absolut_Time</u>, Intensity, Intensity_Modified)

Lamps (<u>ID_description (FK), Wave_Length, Absolut_Time</u>, Intensity, Intensity_Modified)

Now with this new structure the user can select the samples to analyze choosing it by the measurement parameters. The first database structure, the user only had the three dimension matrix and the date that the measurement was taken.

The first table is the one that identify the trial with some characteristics about the experiment. The column tool identifies the OES channel used to get the files and the column Work_Station identifies the other tables and can assume one of those three values: Boot, Kiaba, Lamps. Depending in which plasma reactor the OES measurement was applied.

The ID_description is a foreign key for the other three tables and relate itself with the Description table. Also to keep the SQL consistence of keys the columns ID_description, Wave_Length and Absolut_Time were chosen as compose primary key.

This structure of database it was considered good and sufficient for the analysis, therefore it is in its final format. In sequence it will be explained how it was implemented using the *LabView* Database Connectivity Toolkit.

A new subVI, which exports the data to a server and uses an entirely SQL basedData Base Management System, was developed.

7.2: LabView Database Connectivity Toolkit

In this section will be explained a few SQL statements that can be done with VIs from the database connectivity toolkit and were used to develop this project. With most of these VIs it is not need to know SQL to work with database that is why this tool was so important.

7.2.1: Connecting to a database

This toolkit supports multiple simultaneous connections to a single database or to multiples databases. In this project it was necessary to connect to a single database. The method used in this project was Data Source Name (DSN) because the database standard is ODBC.

7.2.2: Defining Tables

DB tool create table is a VI to create the table if specified and it does not already exist. However, use this VI at the highest level if it is wanted more control over the database fields such as specifying column names, data types and whether to allow Null values. There is a size field to fill that needs to be filled only in the case of the data type is a string.

The restrictions of this VI are that the table must have a single primary key, otherwise it can be possible to define the key of the table and that the table cannot have a foreign key. In those cases a special VI has to be used.

7.2.3: Manipulating Data

Now that it was already explained about the SQL statements it is easier to understand the process of writing and reading from the database. The Toolkit already has the VIs to execute the SQL manipulation functions. As the SQL language some parameters have to be passed to execute the function properly.

The INSERT VI has to obligations. The first one is obviously specify in which table the values are being inserted. The second one is related to the fact that this VI accepts a polymorphic value, that is why is necessary to pass the value as a cluster. For each row at the table a cluster has to be made. The order of the elements in the cluster must be the same order of the columns of the database table. Obviously the primary key cannot be null.

It is also possible to insert values for a specify column, but the primary key rule must be preserved (cannot be null and must be unique).

The DELETE function can be applied to drop the whole table or just some rows. The obligations are to define the table that are being modified and the conditions to identify the row.

With the SELECT function the user specifies the table and the columns that he wants to see or modify. Also it is possible to add conditions to do the queries. The resulted data is in variant format, and it is necessary a little transformation to have the data in its original format.

There is a special VI called DB Tool Execute Query that allows sending an SQL string to a database. With this it is possible to create tables with foreign keys and compose primary keys. Also it is possible to create some queries such as search for distinct values, count the numbers of occurrences and find the maximum value of the column and so on.

7.3: Connecting LabView program with the Database

The connection with the database is pretty simple it is just need to pass as parameters for the Connect VI the name of the data base and the login to access it. It is important to always close the connection after doing any operation.

7.4: Creating Tables

The Description table was created using the VI “Create Table” provided by the Database Connectivity Toolkit. For the other three a specified query had to be created because as explained before the VI “Create Table” does not accept foreign keys and compose primary keys. The query is showed below:

```
Create table Kiaba
(ID_Description in references Description(ID),
Wave_Length float,
Absolut_Time varchar(50),
Intensity float,
Intensity_Modified float,
Primary key (ID_description, Wave_Length, Absolut_Time)
```

7.5: Filling Tables

The VI “Insert Data” was used to pass the parameters to the database. The only problem is that this VI accept only one cluster at the time as input (it is not possible pass multiple rows at the time) and due to the fact that there are a lot of data to export this operation takes a lot of time to be realized.

The subVI called when the “Export to Database” button is pressed is showed below:

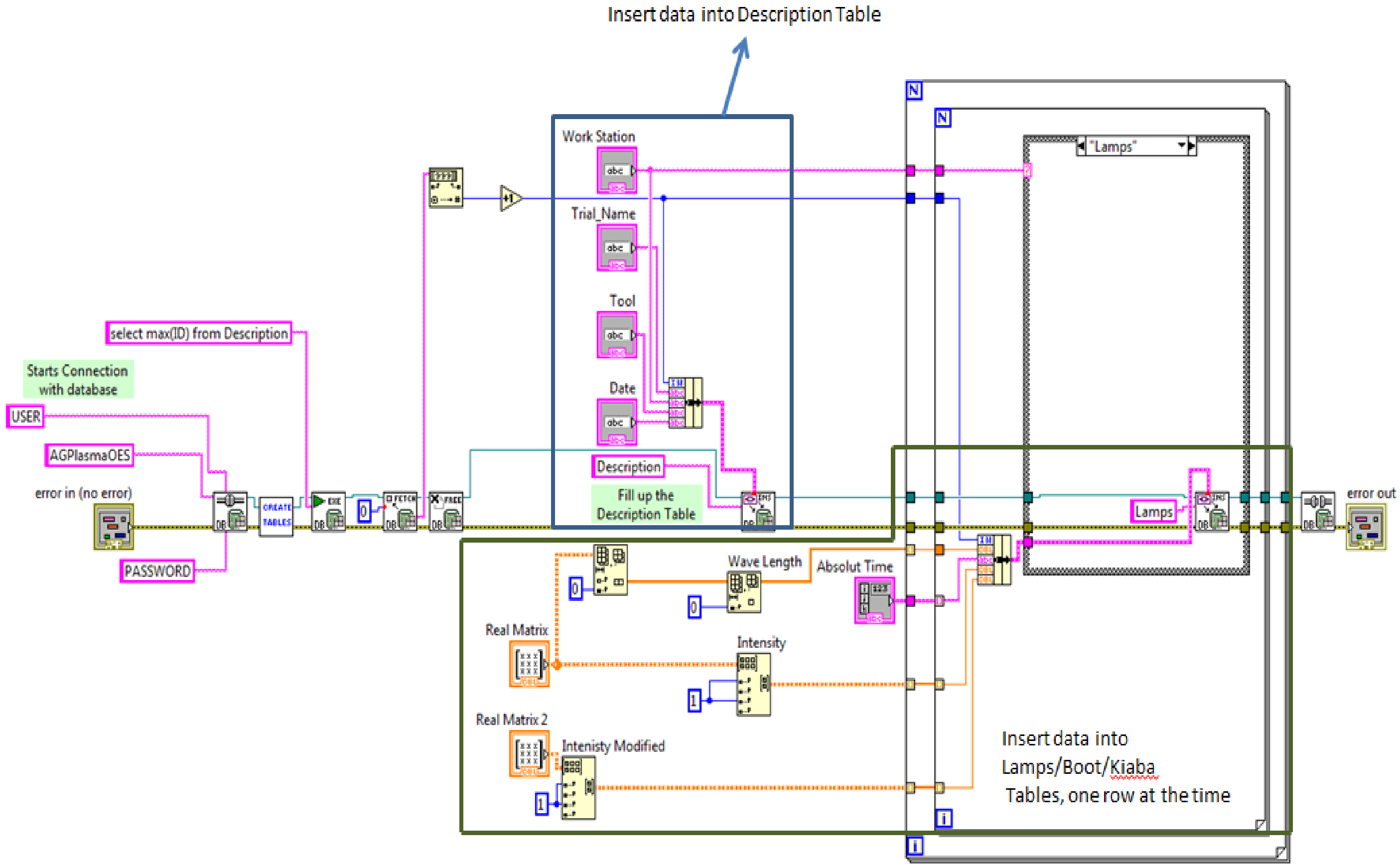

Figure 12 –subVI which fills up the tables in the database

The first thing that this subVI does is create a connection between the database server and the program. After that, the program checks whether or not the tables exist, if they do not exist they are created.

Then it is checked at Description table the last value of the parameter ID. The part highlighted in blue adds a new row at the table Description, with the ID value as the ID gotten at the previous operation plus one. The for loop, create a cluster to insert the values in the other tables. This for loop exists because as said before the VI “Insert Data” only accept one row at the time.

7.6: Database Interface

As the same way as the OES evaluation Interface, the database’s has a few operations to do with the data.

It can be chosen a trial to evaluation just like the OES program, plotting Time versus Intensity graphs for each Wave Length selected. Also the three main operations as Load Reference, Normalize and Load Calibration files can be done here with the data collected from the database and upgrade the corresponding table.

Another option is to compare two samples. This option is important when is being evaluating some plasma changes such as adding a substrate, or to analyze if the less accurate tool provides sufficient information without losing important points from the most accurate.

It is important to remember that with the less accurate tool, the amount of data is much smaller, making the program run faster.

Below at picture 13, it can be seen the database interface:

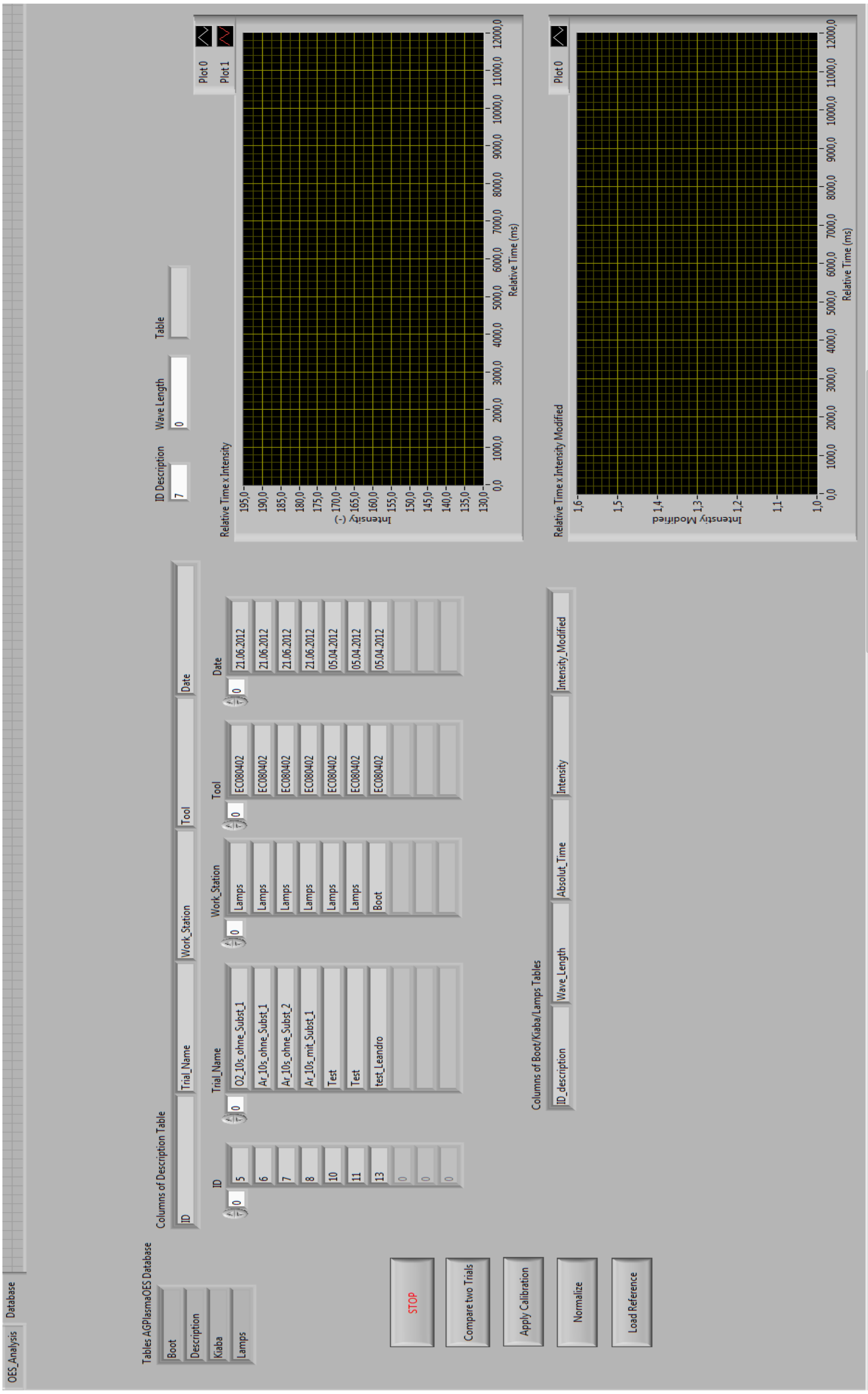

Figure 13 – Interface to manipulate the data from database

The interface shows the tables contained at the database and the Description table, and by this table it is possible to procedure with the evaluation, inserting the ID value at the ID_Description field. When that value change, the program searches into the database, the corresponding table (Lamps, Boot or Kiaba) and gets the values of the rows where ID_description is equal of the inserted value and build the three dimension matrix of this sample.

Once it is done, the three main operations can be done. To compare two trials this is not necessary.

When the user wants to compare to samples a pop up show up to select the samples and the comparison is made only graphically.

7.7: Conclusion

Despite of the fact that the operations with the database are the same as the operations with the three dimension matrix generated by the OES files, making the interfaces separately makes the program runs faster and manage with the situation of the memory storage. Also the final interface is cleaner visually.

The only problem found in this step of the project was the fact that the subVIs Insert and Update from the Database Connectivity Toolkit only accept the data modification row by row, it is not possible to send an entire array directly, so due to the fact that the amount of data is to big it takes too long to export or update the data into the database.

This problem is reversed because the queries run really fast making this option worth.

The database is the backup of the matrix analyzed in the program. It can be export some tables and graphs to excel, but it is not possible to save the whole matrix besides sending to the database, which turns this operation really important for the whole process.

Chapter 8: Results

With the software developed in this project, now the research group has a tool to evaluate low pressure plasma process. This tool works fast and properly and has a friendly interface that makes the analysis easier.

It is also possible to export the pertinent data to a SQL based database and to read the data from it and proceed with the evaluation without processing the data from the text files generated by the OES again.

Also the database is located into a server which has a backup, so it is a safer way to store the data instead of text file at the personal computer.

All the tasks demanded from getting the data in its original form and transform it were accomplished with success.

The measurements, got with the OES and processed by this program, were showed and exported with precision.

Now analyzing step by step of the project showed at the figure 1:

Getting the data into text files and extract the needed information from it and generating a three dimension matrix with the important information, plus an array with the sample information's, works effectively for any size of file and faster enough. All the needed data now are in a LabView format and ready to be manipulated.

The export operation also works In a satisfactory manner, for the Microsoft Excel is easier because when the user exports a sheet for each wave length selected is created and then only the table or graph related with that wave length is exported, in other words, only one line at the three dimension matrix is exported for each sheet.

Contrary to it, when exporting to the database it has to be passed each single element of the matrix separately taking a lot of time, but only had to make this operation once.

Plot the data in graphs only affects one row (in case of plotting the graph Intensity vs Time) or column (graph Intensity vs Wave Length) of the 3D matrix at the time, which makes the operation runs fast and without any problems.

The three possible operations with data to help the evaluation process were easily programmed and do their job correctly.

When asked the final user the opinion of the program, the answer was that the data were organized and showed properly, the main operations were working and returning the expected values and it was really easy to understand and work with the interface.

Chapter 9: Conclusion and Future Prospects

The integration between all the plant parameters with the measurements got with the OES could not be made due the fact that there was not enough time to conclude this step, and this is planned to happen later as a continuity of this project.

Those plants parameters are stored into the database server in another database, the program would get those data and co-relate with the samples and with this information the user could do the evaluation as he wants.

Integration with this program with the programs used to generate the plasma, also written in LabView, was also expected to happen, but due to time limitation it was not possible. This step would be interesting because after the plasma generated and the measurement taken with the OES, the person who operates the machine could analyze the data at the same interface without concerning whit nothing.

Now, how the program is, the person responsible by the measurement has to save the files into a proper folder and has to load those data later in a different computer and interface.

It would be important because already happen case of people forgetting where the files were saved; also sometimes the person who does the measurement and generates the plasma is a different person from who will evaluate the data.

With the evaluate program integrate with the starter program the person could run the data and already export to the database and then the person who will proceed with the evaluation could select the data.

Despite of it, the results obtained with the program in this version met all the design requirements and the user can do his evaluation without any problem, with a friendly interface.

Also comparing with the previous *Matlab* program, this version works faster, and has the possibility to store and read the data into a database also has the load calibration function; such thing does not have in the *Matlab* version. So in comparison the program developed in this project despite of unfinished has more functions and options to work than the previous version, so that is an improvement of the evaluation program.

To the research group this program was important because with it is now possible to verify the uniformity and the composition of the plasma generated, in a non invasive way, to evaluate if the plasma that will be used for barrier coating or for changing surface properties, is being generated following the patterns necessaries to these applications.

In this project it was applied the knowledge got during the graduation, such as database, programming skills, development of a strategy to built the program, use final user’s need to improve the program and so on.

It was an important project not only for applying many engineering skills, but to have an experience working into a real research laboratory environment and improve team work qualities and adaptability skills.

References:

[Gri93] GRILL, A., "*Cold Plasma in Materials Fabrication: From fundamentals to applications*", New York, NY, 1993.

[MAN00] Manual LABVIEW NATIONAL INSTRUMENTS

[Heg07] HEGENBART, A.C., "*Analyse nd Überwachung von Plasmaprozessen für die Kunstsoffverarbeitung mittels optischer Emissionsspektroskopie (OES)"*, 1ªEdição, Alemanha, 2007.

[DFK08] D'AGOSTINHO, R., FLAVIA, P., KAWAY, Y., IKEGAMI, H., SATO, N., AREFI-KHONSARI, F., "*Advanced Plasma Technology*", Ed. Wiley-VCH, Weinheim, Germany, 2008.

[SWS97] STEVENSON, J. O.; WARD P. P.; SMITH, M. L.; MARKLE, R. J.; "*A Plasma Process Monitor/Control System".* IN: CONFERENCE PROCEEDINGS OF SURFACE ANALYSES, 1997, Albuquerque, New Mexico. Chapter of the American Vacuum Society.

[LL05] LIEBERMAN, M. A.; LICHTENBERG, A. J. "*Principles of Plasma Discharges and Materials Processing".* .Second Edition. Hoboken, New Jersey: Wiley-Interscience, 2005.

[MAN01] LabVIEW Database Connectivity Toolkit Manual

[URL00] Institufür Kunststoffverarbeitung [Internet]. Aachen (Germany): RWTH University.Last Access in: July 05th, 2012. Available in: http://www.ikv-aachen.de/en/das-ikv/

[URL01] Institu für Kunststoffverarbeitung [Internet]. Aachen (Germany): RWTH University. Barrier Coating of Polymers: [Around 2 pages]. Last Access in: July 05th, 2012. Available in: http://www.ikv-aachen.de/fileadmin/ikv-uploads/Forschungsschwerpunkte/FAWT/FAWT-EN-Barrier-Coating.pdf

[URL02] Institu für Kunststoffverarbeitung [Internet]. Aachen (Germany): RWTH University. Functionalisation of Plastic Surface : [Around 2 pages]. Last Access in: July 05th, 2012. Available in: http://www.ikv-aachen.de/fileadmin/ikv-uploads/Forschungsschwerpunkte/FAWT/FAWT-EN-Functionalisation.pdf