

UNIVERSIDAD AUTONOMA DE MADRID
ESCUELA POLITECNICA SUPERIOR



**Grado en Ingeniería de Tecnologías y Servicios de
Telecomunicación**

TRABAJO FIN DE GRADO
BACHELOR THESIS

*Programación del adaptador de impedancias AI4S mediante el
protocolo de comunicación Modbus*
*Programming of the tuner impedance AI4S with the Modbus
communication protocol*

**Julián Valentín Ochoa Tutor: Gilles Courret Ponente: Juan
Córcoles Ortega**

MARZO 2019

***Programación del adaptador de impedancias AI4S mediante el
protocolo de comunicación Modbus***

***Programming of the tuner impedance AI4S with the Modbus
communication protocol***

AUTOR: Julián Valentín Ochoa

TUTOR: Gilles Courret



**Dpto. de Tecnología Electrónica y de las Comunicaciones
Escuela Politécnica Superior
Universidad Autónoma de Madrid
Marzo de 2019**



Resumen

Este Trabajo de Fin de Grado consiste principalmente en la creación de una interfaz cuyo objetivo será facilitar el uso y control de un adaptador automático de impedancias (AI4S) creado por la compañía francesa Sairem.

Dicho adaptador, el cual consta de un sensor de impedancias, cuatro stubs, una guía de onda y control electrónico; se encargará por tanto de adaptar la impedancia de un generador de microondas con la impedancia de carga, la cual será variable. El objetivo final no es otro que transferir la máxima potencia posible y a su vez evitar la aparición de ondas electromagnéticas reflejadas, las cuales provocarían una menor potencia en las señales emitidas.

Mi trabajo contribuirá a una parte de un circuito creado en la Haute École d'Ingénierie et de Gestion du Canton de Vaud (HEIG-VD) en el cual la meta final es encender una bombilla en cuyo interior se encuentra un compuesto de sulfuro y argón. La luz emitida por este compuesto es de gran potencia, y su uso puede ser satisfactorio para muchas aplicaciones, tales como fuente de calibración de paneles solares u horticultura cubierta.

La presencia de este nuevo adaptador de impedancias en el circuito de microondas y su capacidad para adaptar cualquier carga, ya sea manual o automáticamente, ayudará a concluir si debido a la reflexión de las ondas electromagnéticas emitidas por el generador de microondas provocan una menor estabilidad en la luz emitida por la bombilla. Dicha estabilidad consistirá en cuánto tiempo el plasma de la bombilla se mantiene en su forma esférica.

En el informe se detallarán las razones de diseño, el proceso empleado para el desarrollo de dicha interfaz y, finalmente, el funcionamiento de esta misma, así como las conclusiones obtenidas y el trabajo futuro.

Palabras clave

Adaptador, Impedancia, Microondas, Modbus, Maestro, Esclavo, Sintonizador, Reflexión, Guía de onda, Potencia, Trama, Registro, LabVIEW, Interfaz.

Abstract

This Bachelor Thesis consists principally on the implementation of an interface whose final goal will be to make easier the use and control of an automatic tuner impedance (AI4S) produced by the French company Sairem.

The tuner mentioned, which contains, in the same equipment, an impedance sensor, four tuning stubs, a waveguide and electronic control; will be in charge of adapting the source impedance of a microwave generator to the load impedance, which will be variable. The final objective is no other but to transfer the maximum power possible and at the same time avoid the presence of reflected electromagnetic waves, which would cause less power in the transmitted signals.

My project will contribute to a certain part of an already created circuit in the Haute École d'Ingénierie et de Gestion du Canton de Vaud (HEIG-VD) in which the final aim is to light a bulb whose inside is made of a compound of sulfur and argon. The light emitted by this compound has great power, and it can be useful for many applications, such as for solar-cell calibration and indoor horticulture.

The presence of this new tuner impedance in the microwave circuit and its capability to adapt any load, either manually or automatically, will help to conclude if due to the appearance of reflected electromagnetic waves transferred by the microwave generator make a less stability in the light emitted by the sulfur plasma bulb. This stability consists on how much time the plasma from the bulb remains in its spherical shape.

The reasons for the design, the process followed during the programming for the development of the user interface and its working will be detailed during the lecture of this report, as well as the final conclusions and the future work.

Keywords

Tuner, Impedance, Microwaves, Modbus, Master, Slave, Stubs, Reflection, Waveguide, Power, Frame, Register, LabVIEW, Interface.

Agradecimientos

Cuando decidí finalizar mi carrera desde Suiza sabía que me enfrentaba a un duro desafío que con trabajo y dedicación podía ser muy satisfactorio, y así fue. Quiero agradecer especialmente a mis padres por ponerme las pilas cuando más se me olvidaba que mi principal objetivo era sacar la carrera de Telecomunicaciones adelante. Me alegro de que lo hayáis hecho con cariño y firmeza al mismo tiempo.

La siguiente persona a la que quiero agradecer es a alguien que jamás me habría imaginado hace cuatro años, ni siquiera hace 7 meses. Esther ha sido quien más me ha apoyado durante mi estancia en Suiza, quien más me ha ayudado cuando más negro lo veía todo y quien me ha hecho ver gracias a su experiencia que todo va a salir bien.

En cuanto al ámbito profesional, especial mención a Gilles Courret por aceptarme en un proyecto tan interesante que ojalá se comercialice en un futuro y por brindarme la oportunidad de terminar la carrera con él. También quiero agradecer a los ingenieros Thibault Gadeyne, de Sairem; Tomasz Kowalczyk, de National Instruments; Julien Maritz, de la HEIG-VD; y Juan Córcoles de la Universidad Autónoma de Madrid. Todos ellos me han ayudado a llevar a cabo este proyecto del que los primeros meses no sabía ni siquiera por dónde empezar por ser absolutamente todo nuevo para mí.

Para terminar, aunque no sois claves en mi carrera, sí lo sois en mi vida, así que a la hora de agradecer me salen los siguientes nombres que me hacen feliz con tan solo recordarlos: Pablete, Clarinete, Edu, Sara, Guille, las Paulas, Berta, Lau, Blanqui, Jaime, Manu, Joseca, las Beas, Iván, Antoja, Lucas, Iniesta, Darío, Pagola, Any, Gisy, Abuela, Silvi, Moni... Y muchos más. Ea.

TABLE OF CONTENTS

1	Introduction	2
1.1	Motivation	2
1.2	Goals.....	2
1.3	Organization of the report	3
2	State of the art	6
2.1	Sulfur plasma bulb	6
2.1.1	Innovation.....	7
2.2	Impedance matching	8
2.2.1	Reflection coefficient	8
2.2.2	Voltage Standing Wave Ratio	9
2.2.3	Single-stub matching	10
2.2.4	Double-stub matching	10
2.2.5	Triple-stub	11
2.3	Plasma as a load impedance	12
2.4	Modbus.....	14
3	Design.....	17
3.1	RS232 Modbus Parameters	17
3.2	User Interface	19
4	Development	24
4.1	Previous steps.....	24
4.2	Block diagram	27
4.2.1	Communication from the Master to the AI4S	27
4.2.2	Communication from the AI4S to the Master	31
5	Integration, tests and results	34
6	Conclusions and future work.....	40
6.1	Conclusions	40
6.2	Future work	40
	References	41
	Glossary.....	44

INDEX OF FIGURES

Figure 2.1 - Sulfur plasma bulb supported with the mechanical rotation system.	6
Figure 2.2 - Scheme of the microwave circuit created by the physicists Koulakis and Putterman and the mechanical engineer Thornton.....	7
Figure 2.3 - At the left, the lighted bulb in its spherical shape. At the right, the plasma has an irregular shape because of the absence of any resonance.	7
Figure 2.4 - Rectangular waveguide WR340	8
Figure 2.5.....	10
Figure 2.6.....	11
Figure 2.7 - At the left, the three-stub tuner used in the microwave circuit. The numbers represent the length of each stub in millimeters (9.2, 12.9 and 26.2 mm). At the right, the three stubs in the interior of the waveguide.....	12
Figure 2.8.....	13
Figure 2.9 - Smith Chart. The red line denotes the axis for inductive and capacitive points, while the green line represents the axis for resistive points.....	13
Figure 2.10.....	14
Figure 3.1 – Automatic four-stub tuner at the laboratory from the HEIG-VD ..	17
Figure 3.2 - Back of the device	18
Figure 3.3 - Parameters that need to be specified for the RS232 cable. The picture is taken from the AI4S user guide.	18
Figure 3.4 - Palette of Boolean controls and indicators provided by LabVIEW 2017 (32-bit version).....	19
Figure 3.5 – Palette of numeric controls and indicators from the modern folder provided by LabVIEW (32-bit version)	20
Figure 3.6 - Serial configuration part of the user interface.....	21
Figure 3.7 - Control part of the user interface.....	22
Figure 3.8 - Tip strip for the RESET button.....	22
Figure 4.1 – Interior of the tuner. At the left, the two pairs of stubs, one from each channel. At the right, four integrated rectifier sensors	24
Figure 4.2 - Plasmionique Modbus Master Library.....	25
Figure 4.3 - Function of every pin from the RS232 cable	26
Figure 4.4 - Beginning of the program	27
Figure 4.5 - Continuation of the program. From now on, the program is inside a while loop	28
Figure 4.6 - Automatic mode	29
Figure 4.7 - Manual mode.....	29
Figure 4.8 - Representation of the high and low position of the stubs, respectively. The positions in both channels are asymmetric	29
Figure 4.9 - Modification of the gain only for automatic mode.....	30
Figure 4.10 – No change in the gain for manual mode.....	30
Figure 4.11 - Lecture of the real positions of the stubs	31
Figure 4.12 - Discriminator level.....	31
Figure 4.13 - Read all faults.....	32
Figure 4.14 - Instructions to correct each of the faults	33
Figure 4.15 - End of the program.....	33
Figure 5.1 - Representation of the stubs in the neutral position (5000).....	34

Figure 5.2 – Positions 2610 and 2652 are the real lowest positions for the stubs from channel X and Y, respectively	35
Figure 5.3 – Interior of the waveguide. Stubs are in their lowest position possible	36
Figure 5.4 – Positions 6873 and 6934 are the real highest positions for the stubs from channel X and Y, respectively	36
Figure 5.5 - Interior of the waveguide. Stubs are in their highest position possible	37
Figure 5.6 – Neutral position of the stubs.....	37
Figure 5.7 - Interior of the waveguide. The stubs are in the central position	38

INDEX OF TABLES

Table 1 - Types of Objects	14
Table 2 - Format of a Modbus RTU frame	15
Table 3 - Function codes	15
Table 4	34

EQUATION INDEX

(1)	8
(2)	9
(3)	9
(4)	9
(5)	9
(6)	9

1 Introduction

1.1 Motivation

Although the main objective from my project is not to create your own matching network, I decided to carry out this project for many different reasons. First of all, a previous knowledge of microwaves was required in order to understand fully the high frequency circuit of the sulfur plasma bulb. The circuit is a very interesting project which surely will be known in the upcoming future due to its utility for the fields mentioned in the abstract. Since my main intention is to dedicate my work life on the microwave engineering field, a complete knowledge of the microwave circuit would fit perfectly on my objectives of improving my skills. A previous study and review of many microwave engineering terms had to be done, such as transmission lines, waveguides, electromagnetic waves, matching networks, the Smith Chart and many more. This study will also help me understand the main function of the tuner AI4S itself as well as its role in the high frequency circuit.

The second point to be taken into account for the realization of this project was the programming itself. A previous knowledge in LabVIEW programming was recommended, although, for the time I started the project, I was a complete beginner. Doubtlessly, a future knowledge in this program was a decisive reason to decide to perform the project. LabVIEW is a very famous program known for its use in data acquisition, industrial automation and instrument control. The knowledge of such a well-known and useful program can be very helpful for my future, since microwave engineering and LabVIEW, as it will be seen along this project, can go hand in hand.

A fact that also motivated me to go through this project was the presence of physics. During my stay at the HEIG-VD I had to refresh some basic physics knowledge, due to the fact it has been a long time since I studied physics. Important terms had to be studied to understand the functioning of the whole circuit, such as plasma vibration, resonance, light emission and many more.

1.2 Goals

Impedance matching is surely one of the main issues to take care of upon microwave engineering. Its role is present in many current cases in everyday life: the connection of a sound amplifier with its corresponding speakers, the connection of a radio transmitting equipment with the antenna system, the connection of an amplifier stage to a transistor with another of similar characteristics, and many more examples. All these mentioned examples have, among other aspects, something in common: an adjustment of impedances must be done in the equipment in order to reach a correct operation.

The goal of this project is nothing else but to acquire a correct operation of an already created microwave circuit. The impedance matching will be done through a tuner fabricated by a company called Sairem. Such tuner will substitute a three-stub tuner which can only be handled manually. To control the new tuner, which has, instead, two pairs of two stubs, I will have to program a user interface with the LabVIEW program by using the Modbus communication protocol. The employees of the physics laboratory of the HEIG- VD will use the interface in the future.

1.3 Organization of the report

The report consists of the following chapters:

- Chapter 1 includes the motivation and the goals of the project and the organization of the report.
- Chapter 2 consists of the state of art, which is divided in a description of the circuit and an explanation about impedance matching and the Modbus communication protocol.
- Chapter 3 considers a previous configuration before starting to program and the design of the user interface.
- Chapter 4 is about the whole development of the user interface, beginning with some previous studies that needed to be done.
- Chapter 5 shows the integration, tests and results of the programming carried out.
- Chapter 6 includes a brief conclusion and the future possible work.
- Chapter 7 includes the references.

2 State of the art

2.1 Sulfur plasma bulb

The programming of the impedance tuner is just a small part of a whole microwave circuit that contributes to the correct transfer of power emitted from a microwave oven. The goal of the microwave circuit is to light a bulb in whose interior contains sulfur powder mixed with argon plasma. The project is nothing else but a more sustainable alternative to the fluorescent lamp predominant nowadays. Starting with the sulfur lamp Solar-1000 [1], developed in the 1990s, many examples have been carried out over the years. A mechanical rotation system, as the one shown in Figure 2.1, was the method used in order to light the sulfur plasma bulb. The system combined in the same equipment a magnetron, a fan whose function is to cool down the magnetron, a mesh cavity that contains in its interior the sulfur plasma bulb and below the cavity a drive which makes the bulb rotate continuously so the compound inside the bulb keeps moving away from the glass of the bulb. In this former version, this last step is indispensable, otherwise the high temperature of the sulfur plasma would melt the bulb.

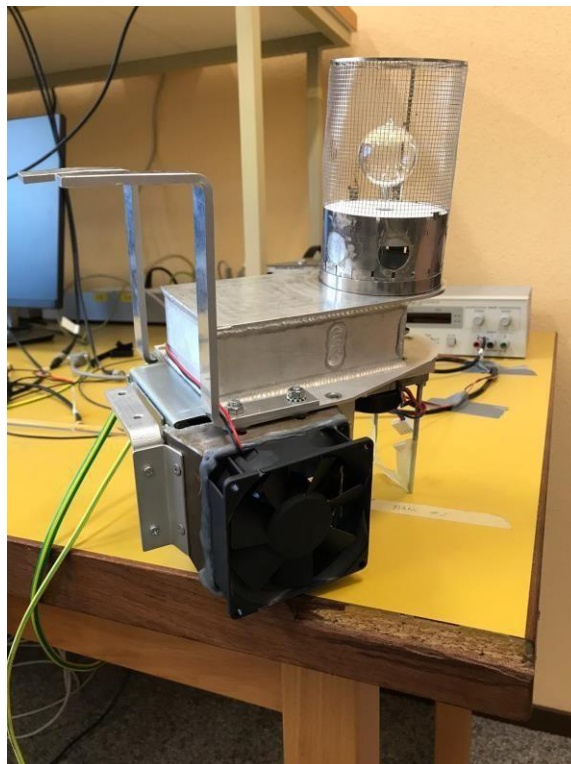


Figure 2.1 - Sulfur plasma bulb supported with the mechanical rotation system.

The procedure is the following: the magnetron produces 2.45 GHz microwaves [2] [14] which are guided by the waveguide from the top to the mesh cavity. Once inside the bulb, the role of the sulfur is to absorb those microwaves, while the function of the argon is to ignite the sulfur. The ignition of the argon provokes the process of fusion and evaporation on the sulfur, which finally emits an efficient and high-powered light [14].

2.1.1 Innovation

Due to recent researches [3], it has been discovered the presence of acoustic modes inside the bulb that favor the idea of replacing the mechanical rotation system by a one which controls the stabilization of the plasma by acoustic modes. These modes let the hottest region of the plasma stay in the center of the bulb, avoiding its melting. This is truly a great advantage since the user will not have to worry whether the bulb is melting or not, letting the user to increase the temperature of the plasma. Hence, an improvement on the efficiency of the lamp might be achieved.

A waveguide circuit might be an alternative to the mechanical rotation system. For example, the waveguide circuit developed by John P. Koulakis, Alexander L. F. Thornton and Seth Putterman [4] has achieved some satisfactory results. The circuit works as it follows: starting with a magnetron at 2.45 GHz, the circuit is followed by an isolator, a directional coupler, useful to measure both forward and reflected power flowing in one direction; a three-stub tuner and a sliding short, both used for guaranteeing an optimum transfer of the maximum power possible to the load; and the mesh cavity where the sulfur plasma bulb is. The circuit is schematized in the following image. The circuit used during my stay at the HEIG-VD has a similar form as the one showed, although the isolator was not mounted.

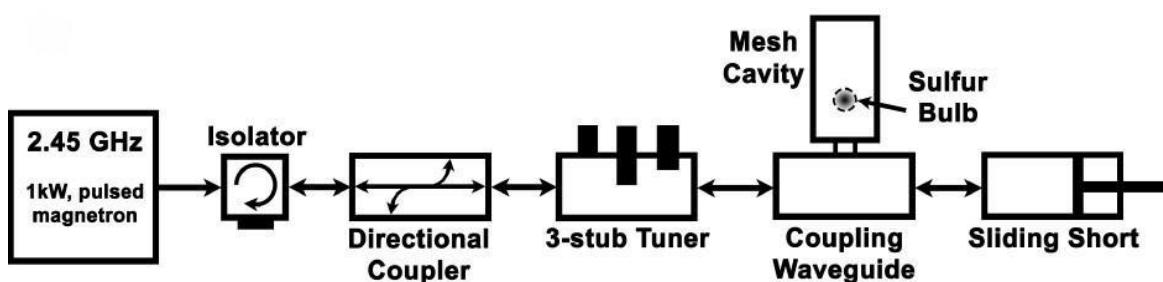


Figure 2.2 - Scheme of the microwave circuit created by the physicists Koulakis and Putterman and the mechanical engineer Thornton.

Unfortunately, for the moment, the stabilization of the plasma in the center of the bulb does not maintain its spherical resonance for no longer than one second, approximately. That is why the four-stub tuner in question has replaced the three-stub tuner, due to the fact that this new tuner has the capability of being a motorized and automatic tuner.

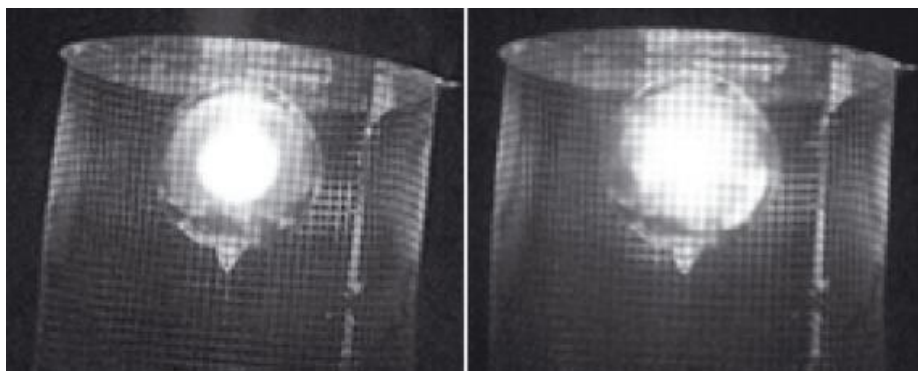


Figure 2.3 - At the left, the lighted bulb in its spherical shape. At the right, the plasma has an irregular shape because of the absence of any resonance.

2.2 Impedance matching

Impedance tuning is indispensable for microwave circuits. Its importance is decisive for two main reasons. Firstly, every time the load of a certain circuit is matched with the generator's source, the maximum power will be delivered to the load and, therefore, the loss of power will be minimized. If, for example, a power generator has an incident power of 6 kW (kilowatts) but, however, half of this power is reflected back, only 3 kW will actually be delivered to the aim of a certain circuit. Thus, this consequence leads us to the next important reason of impedance matching: an improvement in the signal-to-noise ratio (SNR) of the system. Antennas, amplifiers and many more receiver components will benefit from the adjustment of impedances with an improvement in the quality of their signals. In our case, the sulfur plasma bulb would take benefit of it by improving in its stability.

Firstly, I will briefly discuss about the reflection coefficient and the VSWR (Voltage Standing Wave Ratio). To continue, I will introduce the single-stub tuning, the simplest of the matching techniques, and, after that, I will direct our vision to the double-stub tuning, which is, in a certain way, the case of this project. Finally, I will end with a brief description of the triple-stub matching technique.

2.2.1 Reflection coefficient

The reflection coefficient is an impedance ratio which allows us to determine the current situation of an electromagnetic wave transmitted in a transmission line, in our case a waveguide as the one shown in Figure 2.4.

$$\Gamma = \frac{V_{\text{reflected}}}{V_{\text{forward}}} = \frac{Z_2 - Z_1}{Z_2 + Z_1} \quad (1)$$

The previous equation (1) defines the reflection coefficient. Its value varies from 0 to 1, where zero stands for full transmission of the wave, while 1 means full reflection of the wave. Z_1 and Z_2 refer to the characteristic impedance of the waveguide and the load impedance [5], respectively.



Figure 2.4 - Rectangular waveguide WR340

2.2.2 Voltage Standing Wave Ratio

A different way of determining an impedance mismatch is the Voltage Standing Wave Ratio (from now on, the VSWR). A standing wave is the superposition of the transmitted and reflected wave, that is, two waves going in opposite directions in the interior of a transmission line, such as a waveguide.

Thus, the voltage standing wave refers to the sum of the amplitudes of the incident wave (V_i) and the reflected wave (V_r). Since the VSWR is about a ratio, to quantify this value we need to divide the maximum and minimum possible values of the standing wave, which leads us to the following definition of the VSWR [6]:

$$VSWR = \frac{V_{max}}{V_{min}} = \frac{V_i + V_r}{V_i - V_r} \quad (2)$$

The VSWR can also be defined as:

$$VSWR = \frac{1 + |\Gamma|}{1 - |\Gamma|} \quad (3)$$

The last equation (3) can also be written as:

$$|\Gamma| = \frac{VSWR - 1}{VSWR + 1} \quad (4)$$

A value of $VSWR = 1$ denotes full transmission along the waveguide, thus, no reflection, while a value of ∞ means full reflection. In case it exists reflected power, we define return loss as the power which was not delivered from the generator to the load. The return loss is defined as:

$$RL \text{ (dB)} = -20 \log_{10} |\Gamma| \quad (5)$$

Since the power is proportional to the square of the voltage, the reflected power can also be expressed as a percentage of the total wave emitted by making use of the next equation, which shows how to calculate the reflected power ratio.

$$\text{Reflected Power (\%)} = 100 * |\Gamma|^2 \quad (6)$$

During the Integration, tests and results section of the report, it will be explained the maximum VSWR allowed for our tuner impedance.

2.2.3 Single-stub matching

The method used inside the AI4S is the tuning with stubs. Such tuning is always more convenient for microwave circuits rather than lumped elements, which are simpler and suitable for circuits where the frequency is low enough (up to about 1 GHz) and/or where the size of the circuits is quite small [7].

In single-stub tuning, the two parameters to be taken into account are d , the distance from the stub to the load, and l , the length of the stub, as it can be shown in Figure 2.5. The length of the stub will determine the desired value of the reactance (or susceptance) in order to reach the impedance adjustment.

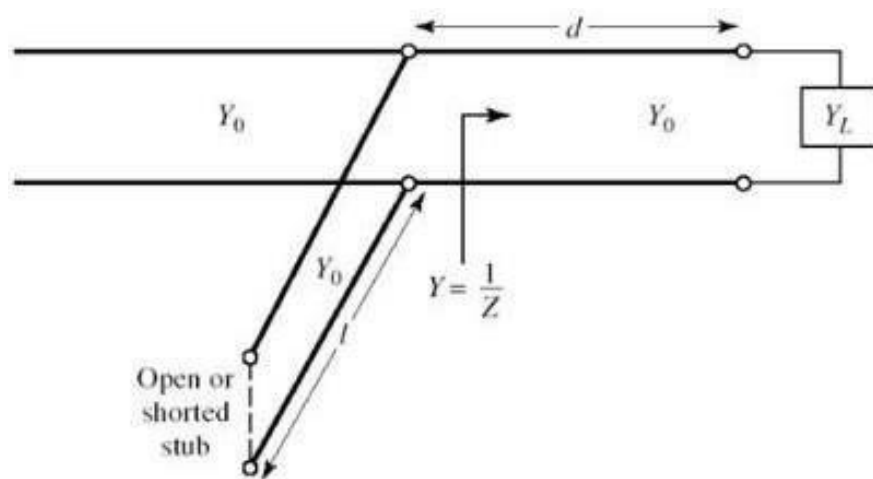


Figure 2.5

As it can be seen in the previous figure, the stub can be ended either in an open or short-circuited way. In our case, for waveguides, it is more preferable to use short-circuited stubs, since the length of the opened one may be too large (in terms of electrical length), provoking a radiation which would lead to a non-purely-reactive impedance of the stub.

2.2.4 Double-stub matching

The single-stub tuning is an impedance matching technique than can match any load impedance (on the condition that it does not have a zero-real part) to a source impedance and a transmission line. The main disadvantage of this technique is its requirement of a variable length of line between the load and the stub (d). This drawback affects to those who desire to create an adjustable tuner, as the four-stub tuner of the project. To deal with this inconvenient, two tuning stubs in fixed positions can resolve it.

Double-stub tuners are frequently fabricated in waveguides or coaxial lines, where the adjustable stubs are connected in parallel to the main transmission line. Its main disadvantage is its inability to match every impedance. The range of impedances that are

impossible to be matched is known as the forbidden area. The forbidden range of impedances unable to be matched can be reduced by diminishing the distance between the two stubs. Despite this drawback, it lets the load to be in an arbitrary distance away from the stubs, in contrast to the single-stub tuning.

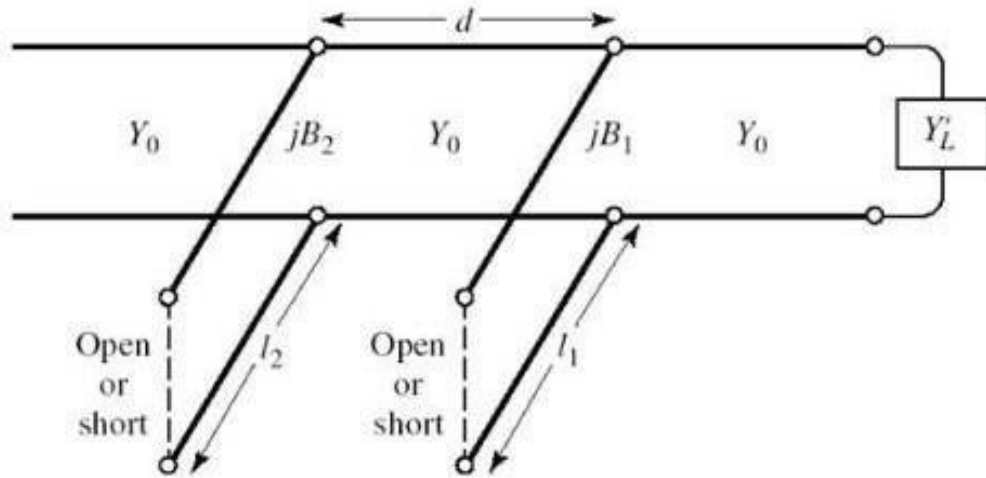


Figure 2.6

Figure 2.6 shows the main idea of the double-stub tuning. As it can be seen, the two parameters which need to be calculated are the lengths of both stubs (l_1 and l_2), while the distance between them (d) usually is chosen to be either $\lambda/8$ or $3\lambda/8$.

2.2.5 Triple-stub

The double-stub matching technique does not cover a certain range of impedances. Although the range can be reduced by diminishing the distance between the two stubs, there will always exist a certain limit of impedances impossible to be matched. The presence of a third stub allows covering the whole range of impedances by playing between the pair of consecutive stubs. It is recommendable not to move the three of them at the same time [12]. This means, according to the next image, to combine stubs A and B or stubs B and C, while the other stub remains static.

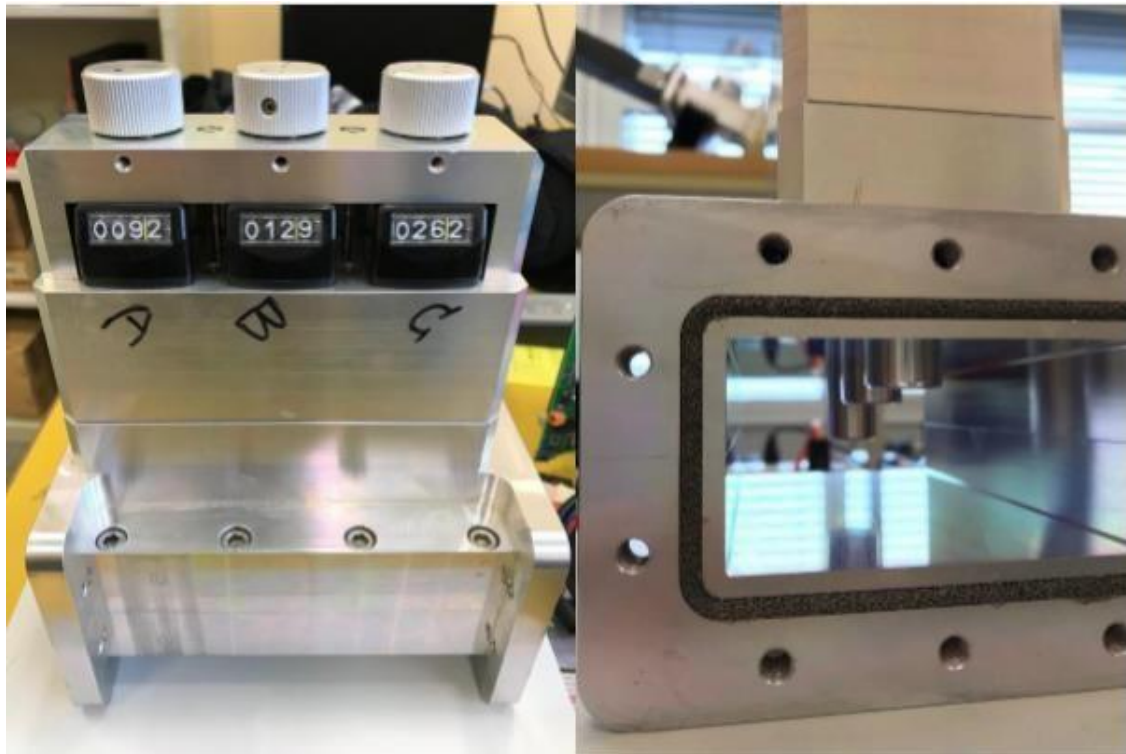


Figure 2.7 - At the left, the three-stub tuner used in the microwave circuit. The numbers represent the length of each stub in millimeters (9.2, 12.9 and 26.2 mm). At the right, the three stubs in the interior of the waveguide.

2.3 Plasma as a load impedance

Most RF (Radiofrequency) microwave generators have a source impedance of 50 Ohms, so the load impedance (that is, the plasma impedance in our case) will have to be adapted by the microwave tuner. That means the matching network, which will be placed between the power generator and the plasma impedance, will have to be designed in a certain way that the load impedance finally is converted into a 50 Ohms impedance.

As it has been mentioned before, the sulfur plasma impedance is a variable impedance. Its variance depends on many different factors: the ignited plasma, which provokes a change in its temperature, the appearance of gas and the mesh cavity itself where the bulb is located are some factors that may contribute to the change of the impedance of the plasma [13]. All these different factors result on the possible appearance of the three different types of impedances possible: inductive, capacitive and resistive.

That is the main reason why the impedance tuner needs to be working continuously in order to cover a concrete range of impedances while the process keeps running. This range of impedances can be estimated by a feature characteristic on the plasma impedances. This sort of impedances is usually made of a resistive part (R_1) and a capacitive part, which is, in turn, divided into the cold (C_c) and plasma (C_p) capacitances. The cold capacitance of the plasma is the one that lets the designer of a matching network estimate how wide the range of impedances that will need to be covered will it be.

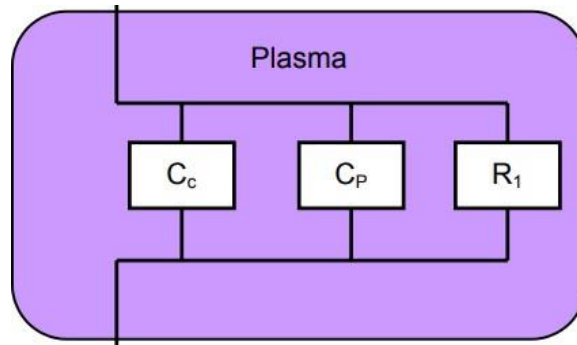


Figure 2.8

The microwave source is usually designed in a way that a purely real impedance (thus, with no reactive components), such as 50 Ohms, will need to be matched. Nevertheless, as it has been said, there is a strong possibility that inductive and capacitive components will appear on the load impedance. Thus, the matching network will have to be designed with reactive components which will store energy and “cancel” the reactive part of the load impedance. That is why stubs have almost always a purely imaginative characteristic impedance which will annulate the imaginary part of the load impedance in order to tune it successfully with the one of the microwave source generator.

The tool used theoretically to design a matching network is the Smith Chart. One can measure with no power supply the impedance of the cold capacitance and, after some few steps which need to be done in the Smith Chart, we can achieve the matching network for that specific cold load impedance. The role of the four stubs from the AI4S tuner is very similar. A change in the positions of each stub will induce a change in the impedance of the load. Stubs from channel X will act on the resistive part of the Smith Chart, while stubs from channel Y will act on the reactive part of it. The automatic mode control of the device will guarantee an optimum power transfer from the microwave oven to the plasma load.

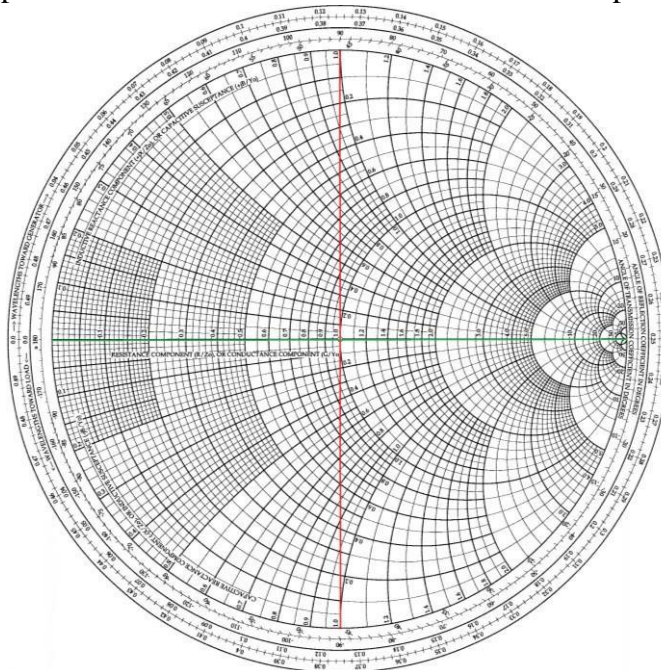


Figure 2.9 - Smith Chart. The red line denotes the axis for inductive and capacitive points, while the green line represents the axis for resistive points.

2.4 Modbus

Modbus is generally used for SCADA style network communication between devices, where SCADA stands for Supervisory Control And Data Acquisition. Modbus follows a master/slave architecture where the master sends a request to the slave and waits for a response, as seen in Figure 2.10. This architecture allows the master to have complete control over the whole flow of information.

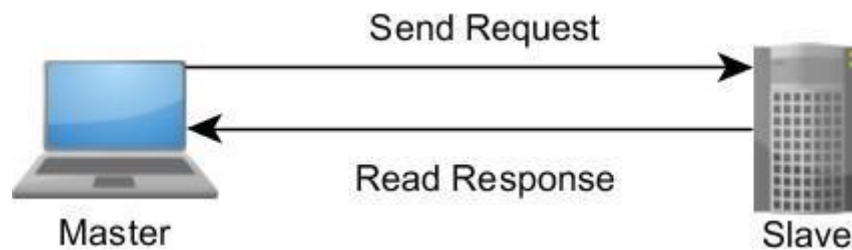


Figure 2.10

The mentioned request is a data set in layers. The first layer is the Application Data Unit, which often is what it is considered as the type of Modbus communication. There are three types of Modbus communication: ASCII (American Standard Code for Information Interchange), TCP/IP (Transmission Control Protocol/ Internet Protocol) and RTU (Remote Terminal Unit). All of them depend on the desired physical network (that would be Serial, Ethernet, etc.), the quantity of devices in the network and the type of Modbus communication that the devices, either master and/or slave, can support. The protocol that had to be used during this project was the Modbus RTU (Remote Terminal Unit), since it is the appropriate for serial transmission mode.

Although Modbus ASCII is also convenient for serial transmission mode, the main advantage of RTU mode is its greater density of characters, which results in a better performance for the same velocity. Modbus RTU makes use of a binary representation of data for the communication protocol. The RTU format follows the data with a Cyclic Redundancy Checksum (CRC) as an error checking mechanism to guarantee the reliability of the data.

Modbus only admits two types of data: Booleans (size of 1 bit) and unsigned integers (size of 16 bits). Within these two types of data, it distinguishes between four different types of objects. The master and the slave have different type of access to those objects. The next table explains clearer the different kind of objects Modbus can handle with:

Table 1 - Types of Objects

Type of Object	Data Type	Master Access	Slave Access
Coils	Boolean	Read/Write	Read/Write
Discrete Inputs	Boolean	Read	Read/Write
Holding Registers	Unsigned Integer	Read/Write	Read/Write
Input Registers	Unsigned Integer	Read	Read/Write

For the case it concerns us, only coils and registers will be necessary and, since our laptop will be the master, only the master access is necessary to take into account. We will then wait for the response of the slave from the master.

The format of a Modbus RTU frame is as it follows:

Table 2 - Format of a Modbus RTU frame

	Start	Address	Function	Data	CRC	End
Bits	28	8	8	N*8	16	28

Between each frame, at least 28 bits must be respected, which is known in this communication protocol as time of silence. The frame begins with 8 bits which correspond to the address and, right after that, the following 8 bits correspond to the function code. Function codes determine how the master accesses and modifies the data. When the master sends a request to the slave in order to run a function code, the slave uses the parameters of the function to run that well-defined behavior. The main functions in Modbus RTU are the following ones:

Table 3 - Function codes

Function	Function code (Hexadecimal)
Read Discrete Inputs	0x02
Read Coils	0x01
Write Single Coil	0x05
Write Multiple Coils	0x15
Read Input Register	0x04
Read Holding Registers	0x03
Write Single Register	0x06
Write Multiple Registers	0x16

The main functions the master will implement during this project will be “Write Single Register” and “Read Holding Registers”, among other functions.

Continuing with the format of the Modbus frame, the field “Data” must be filled, of course, with the data plus a length which will depend on the type of message. Finally, the last 16 bits correspond to the Cyclic Redundance Checksum (CRC) mentioned before.

3 Design

3.1 RS232 Modbus Parameters

The tuner has some characteristics detailed in the datasheet provided by Sairem which have to be respected for a correct functioning of the device. Specially, what needs to be correctly established are the serial communication parameters. As mentioned, the protocol that had to be used is the Modbus RTU (Remote Terminal Unit), since it is the appropriate for serial transmission mode. As it can be seen in Figure 3.2, the AI4S must be connected on the RS232 MODBUS connector using a DB-9 to DB-9 “straight” cable, where straight means that pin 1 is connected to pin 1 of the other side, pin 2 to pin 2, and so on.



Figure 3.1 – Automatic four-stub tuner at the laboratory from the HEIG-VD

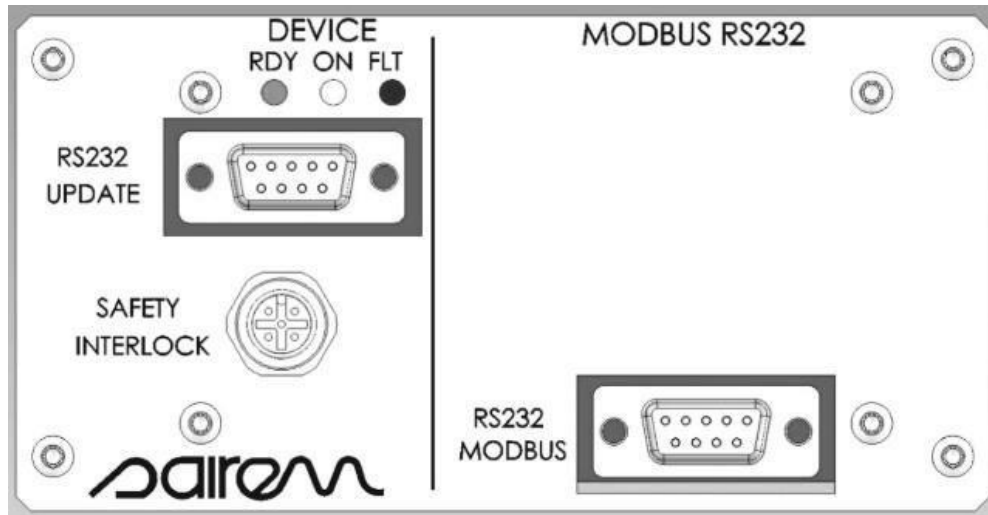


Figure 3.2 - Back of the device

To initialize the serial communication, it is necessary to specify the format of the RS232 cable, which is detailed in the user manual of the device. The format must be 115200-8-N-1. The first number defines the baud rate, which means the rate (or how fast) at which the information will be transferred through the communication channel. There are many typical values for the baud rate, and 115200 bps (bits per second) is probably the greatest one.

The numbers 8 and 1 refer to the bits being sent to the device. The master acts on the slave by sending requests in form of functions. As it will be detailed later, some of the functions that can be asked by the master are to read multiple bytes or write a single bit, among other requests. Those requests have, each of them, a hexadecimal function code of two digits (or 1 byte or 8 bits). Thus, the master has to send 8 bits, plus two more bits, located at the beginning and at the end of each packet sent. The first bit is called as start bit while the last bit is known as stop bit. The last one can be modified to be either one or two bits, so that is why it needs to be specified how many stop bits are there going to be used.

A way used to check whether an error occurs or not is to use the technique of parity. This technique consists on the number of 1's that are in the byte sent. Depending on whether this number is even or odd, the user can decide to establish the parity as even or odd in order to check for possible errors occurred during transmission. Unfortunately, this technique might slow down the data transmission, so probably that is the reason for which the parity in our case needs to be specified as None (N). Due to this decision, the flow control (or handshaking), which assures that both master and slave are ready to communicate between them, must also be specified as None.

RS232 MODBUS parameters				
Baud rate	Data bits	Parity	Stop Bits	Handshaking
115200	8	None	1	None

Figure 3.3 - Parameters that need to be specified for the RS232 cable. The picture is taken from the AI4S user guide.

3.2 User Interface

To build a LabVIEW user interface, the program provides us with controls and indicators which have to be placed in the front panel of the program. Controls allow us to manipulate the data, which lets the user, for example, decide the temperature in a Celsius to Fahrenheit program or to stop the program whenever the user desires to. On the other hand, indicators refer to every value the slave returns to the master. They can indicate these values in form of numeric data, graphics, LEDs, etc. LabVIEW has available a great variety of different controls and indicators with multiple designs depending on the interests of the programmer.

According to the AI4S user manual, the computer (the master) needs to direct the programming through one road or another. It needs to specify whether the mode control of each channel is automatic or manual, allow the user to stop the request sent for each channel, create a RESET request in case some faults occur during the communication and finally allow the modification of the channel gain, which can be changed by a simple numeric control. All these specifications can be established in many ways. In my case, I decided to create buttons (Booleans) in order to be familiar and more interactive for the user. Figure 3.4 shows the different kinds of buttons that are available in LabVIEW program.

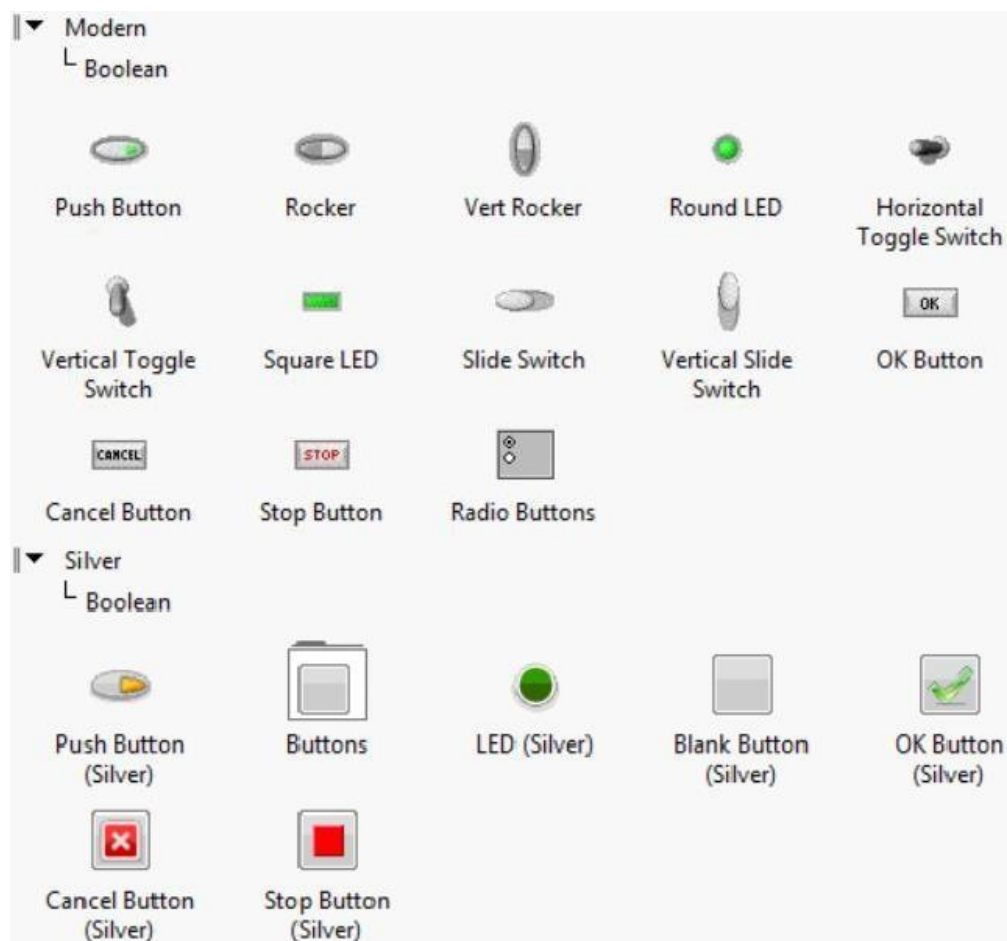


Figure 3.4 - Palette of Boolean controls and indicators provided by LabVIEW 2017 (32-bit version)

For design reasons, I decided to use the silver blank button for all buttons related with channels X and Y. For the RESET button, a silver push button was selected in order to make the user take into account the importance of this button in comparison with the rest of them. Modern round LEDs were used for all the different faults that may occur during the communication between the master and the slave, such as over currents, thermal faults or internal communication faults. Finally, the silver stop button was the one selected in order to finish with the communication anytime the user desires to.

Regarding the stubs, the user needs to set a desired position for them. According to the guide, this position can be established from 0 to 10.000, so, to let decide which value is desired for the position of the stubs from each channel, I made use of two vertical pointer slides. Since the possibility of establishing a very concrete position on the stubs exists, I decided to attach to both slides a digital display to let the user write the exact position set point because the pointer is not as accurate as the numeric display.

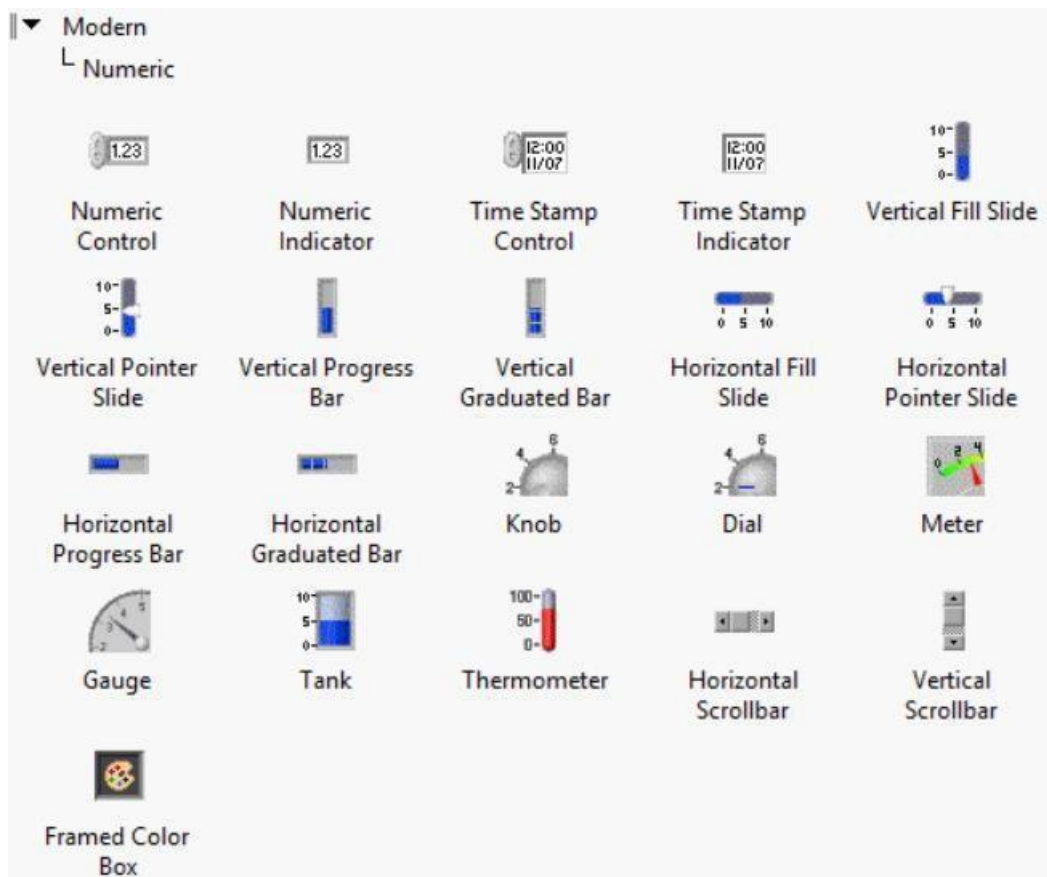


Figure 3.5 – Palette of numeric controls and indicators from the modern folder provided by LabVIEW (32-bit version)

Regarding the communication from the slave to the master, the last one needs to read the response to its requests. To do so, I made use of different indicators. As mentioned before, round LEDs were used to declare any fault and, attached to each LED, there is a string indicator in order to detail what error has happened during the communication.

Furthermore, for each channel I placed two more indicators; specifically, one vertical

progress bar and one horizontal graduated bar. The role of the first one is to indicate the real position of the stubs, which will be close to the position specified by the user in the set point slide. In order to have an accurate value of the real position, I attached to both bars a numeric indicator. Lastly, the horizontal graduated bar, also with an additional digital display, will be used for the discriminator readout, which will be explained later.

The final design of the interface is showed in Figure 3.6 and Figure 3.7. The user interface is made up of two sections. The first section is the serial configuration, while the second section goes with everything related to the control of the tuner impedance. As it can be seen in the first image, for design reasons, I included a picture of the tuner and the logo of the HEIG-VD, as well as a brief description of the tuner. Besides, I used numeric string controls in order to define the values of the parameters needed for a correct Modbus communication.

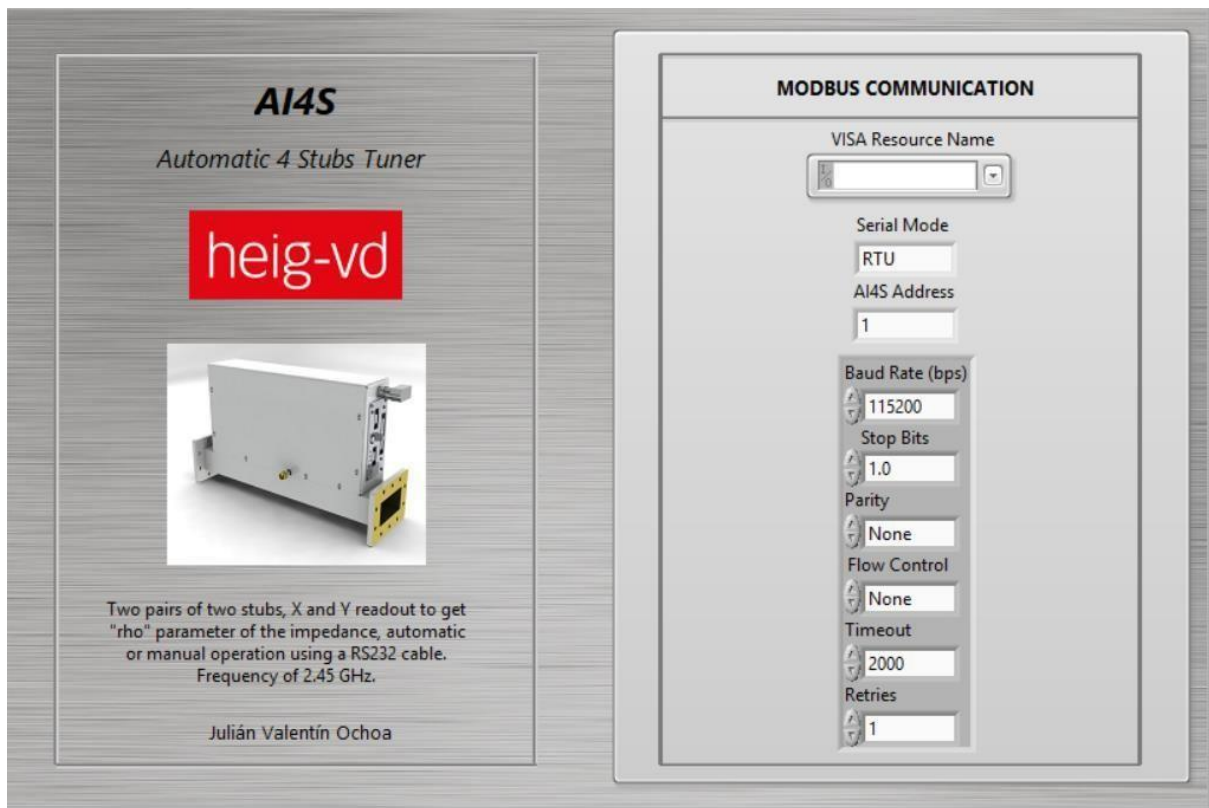


Figure 3.6 - Serial configuration part of the user interface

In both parts of the user interface, the default values are the ideal ones. In what respects to the default values of the serial configuration parameters, they should not be changed otherwise the Modbus communication will not work properly. Only the timeout and the retries values can be modified. Regarding the second part of the user interface, the stubs have been positioned in the neutral position (5.000) during the first execution, but, of course, this value can be modified whenever the user needs to.

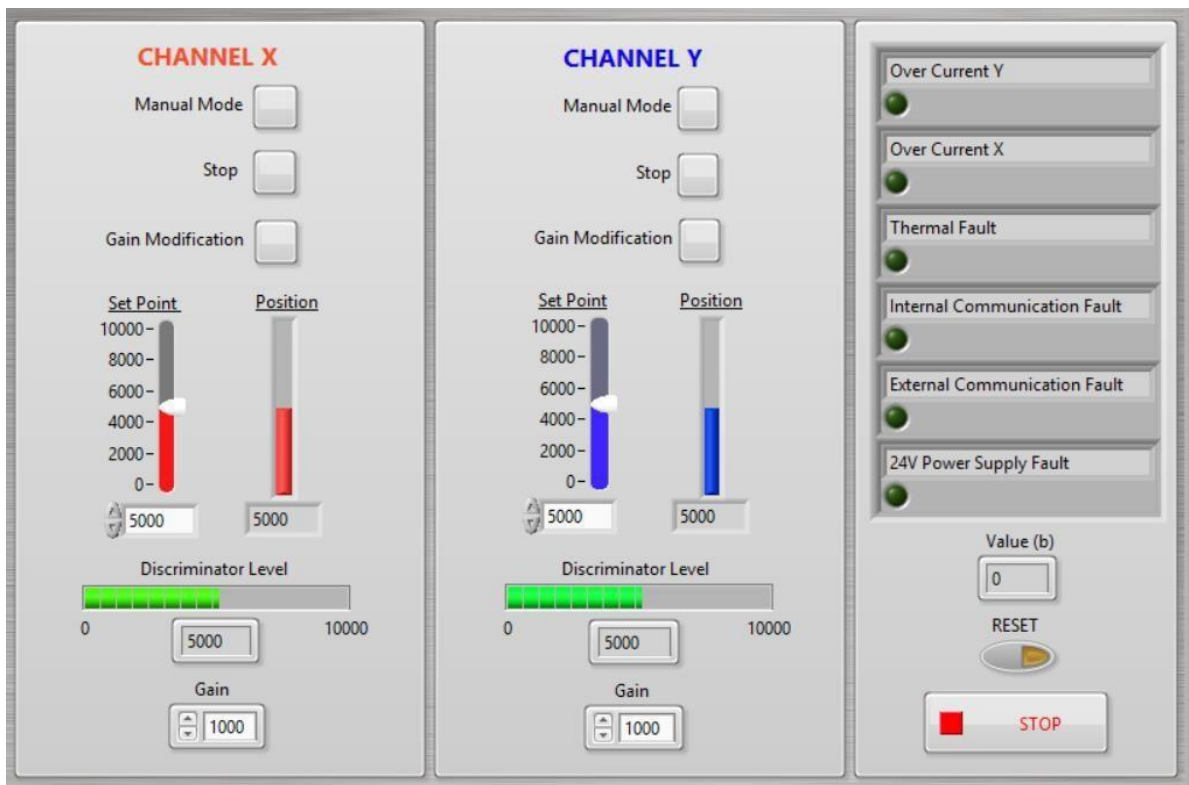


Figure 3.7 - Control part of the user interface

Lastly, during my design I had to think about who is going to use this interface. That means I must create it in a way that future users can use the application without any difficulties. For any future questions these users might have, I included some additional information on every control and indicator to provide enough information. This information can be seen every time the users move their cursor over them. The next Figure is an example for the RESET button.

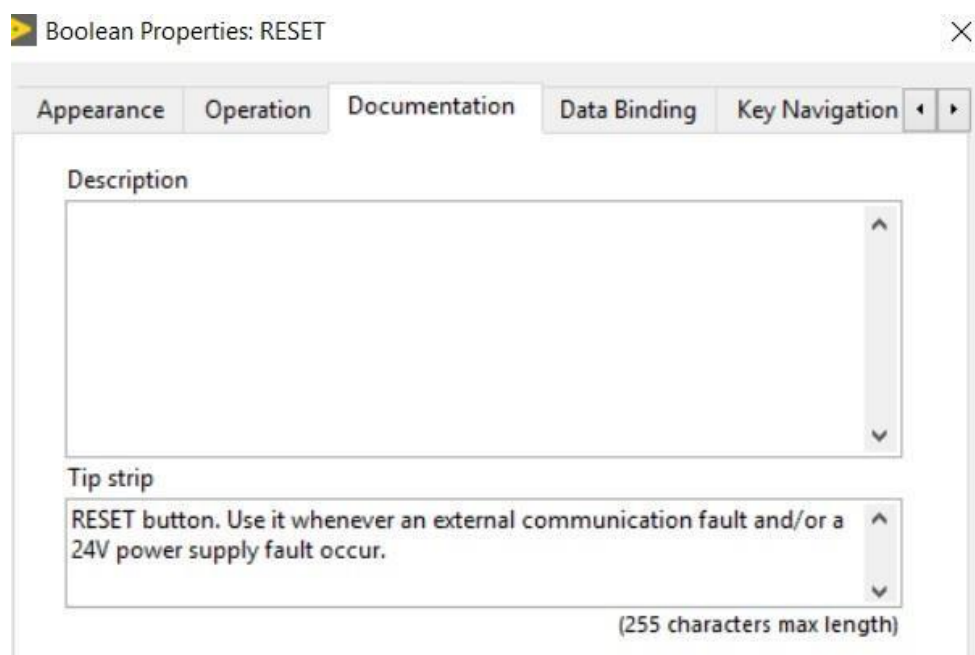


Figure 3.8 - Tip strip for the RESET button

4 Development

4.1 Previous steps

The initials stand for “Adaptateur d’impédance à 4 stubs” (in French), which, in English, means Automatic 4 Stubs Impedance Tuner. With it, the datasheet and the operation manual were included. After a detailed lecture of the manual, I had to figure out what did I need exactly in order to start programming it.

The automatic matching tuner contains, in the same equipment, an impedance sensor, tuning stubs and electronic control. The four stubs are grouped by pairs acting on X and Y-axis of the impedance diagram (Smith Diagram). The variation of the positions of the stubs from channels X and Y will induce a variation of the impedance of the load in question. To communicate with the tuner, the Modbus communication protocol allows controlling it over RS232 serial line. The master, which could be either a PLC (Programmable Logic Controller) or a computer, will send continuously requests to the slave, which will be the AI4S. In our case, due to the lack of a PLC, we decided that the master would be a Lenovo laptop provided by the university.

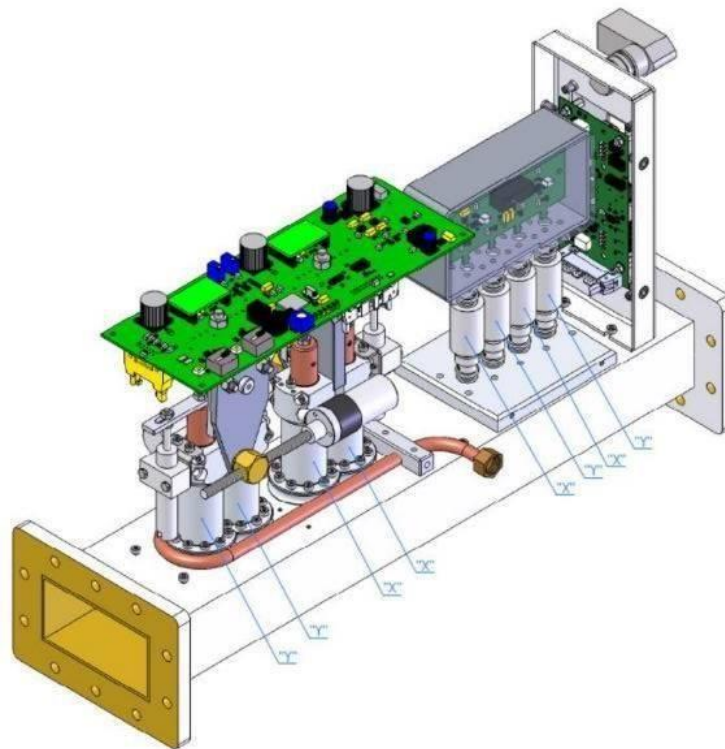


Figure 4.1 – Interior of the tuner. At the left, the two pairs of stubs, one from each channel. At the right, four integrated rectifier sensors

To achieve a Modbus communication, it was necessary to download a program that supports this kind of protocol. Two options were available: LabVIEW or Matlab. Because of previous knowledge of Matlab program during my career, I decided to learn about LabVIEW because of the reasons exposed in the first section of the report.

Thanks to the well-organized National Instruments website, it was not necessary to spend too much time at checking what features I exactly needed for establishing a Modbus communication in LabVIEW. Firstly, LabVIEW 2017 program (32-bit version) was downloaded. After that, some modules suitable for Modbus protocol were needed. Specifically, the Full Development system and the Datalogging and Supervisory Control and Real Time modules. Lastly, it was necessary to download a Modbus library. In concrete, the two libraries I downloaded were NI's Modbus V1.2.1 and the Plasmionique Modbus Master libraries, created by an advanced plasma-based technology company. This last library is a replacement for the first one officially approved by National Instruments and the one used along this project. Both libraries included some Modbus Examples in which I could base on according to my interests.

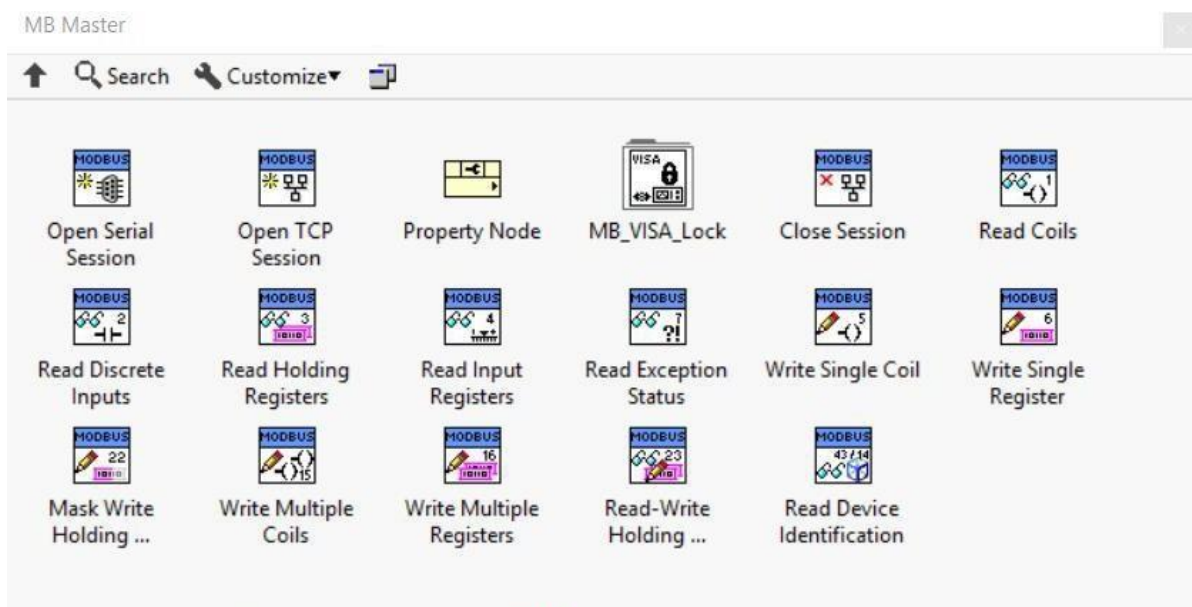


Figure 4.2 - Plasmionique Modbus Master Library

Once everything was ready, firstly I had to get familiar with the LabVIEW program and the Modbus communication protocol. For that, National Instruments provides a series of introductory videos [12] in which they explain the most used items through many examples. LabVIEW is famous for its ease of use due to a graphical programming. It is equally valid either for professional programmers as it is for beginners. Lack of knowledge is not a problem for beginners that desire to do a complex program. Programs made in LabVIEW are called Virtual Instruments (VIs) and they are divided into two parts, the front panel and the block diagram. The first one plays the role of the interface while the second one is the program itself. The process of understanding was based on the programming of different VIs that included some of the most important functions, such as different kinds of loops, arrays, data types, charts, graphs and many more.

After a deep learning about LabVIEW and Modbus protocol, it was time for me to start with a simple program that could let me communicate with the device. The university provided me a RS232-USB converter (because the laptop lacks a RS232 port) to make the communication between the AI4S and my laptop possible. Once finished my simple program and ran the program on the device, an error appeared. I typed the error ("Error 6101, Timeout Error") on the National Instruments discussion forums to check if any other programmer had

the same error as me. Unfortunately, this error turned out to be a very general error, which means many different reasons could have caused it.

After trying many advices from the discussion forums, none of the possible reasons resulted to be satisfactory. My next step was to call for a NI service request. Engineer Tomasz Kowalczyk answered my call. His advice was to start, together, from the beginning. That meant checking, firstly, if the cable being used (RS232-USB converter) was working properly. After that, we would be moving forward on the program itself.

To check the behavior of the cable, the engineer told me to do a Loopback Test. Once done, I would call him again. To do so, I downloaded the PuTTY program in order to check if pins 2 and 3 (transmission and receiver pins) communicated between them. I used a simple wire to connect both pins and, meanwhile, typed on the PuTTY program any word on its terminal to see if communication was being done properly. Unfortunately, nothing was typed, which meant the cable did not work. The next step was to ask to the university for a new RS232-USB converter whose driver was compatible with Windows 10 version. I had to wait for an entire week to finally receive a new converter and continue working.

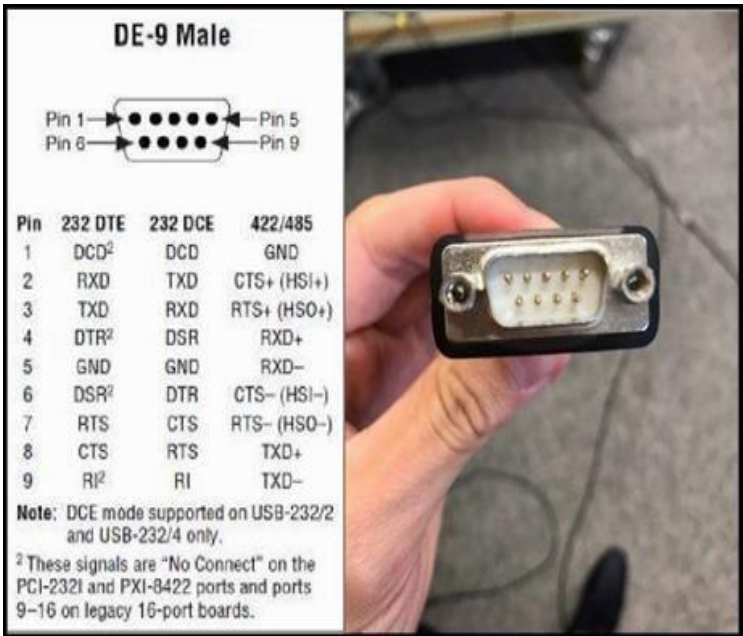


Figure 4.3 - Function of every pin from the RS232 cable

Finally, after one week, the new converter arrived. Therefore, I checked whether communication between both pins worked, and, indeed, it did. However, Error 6101 kept appearing. Tomasz advice was to keep changing small details from my program and compare with some of the Modbus libraries examples what I did have different.

No solution was found after some days, so I decided to contact Sairem company (company that created the AI4S) to check if they could help me find a solution to my problem. The engineer Thibault Gadeyne contacted me by email and sent me a VI example of how it communicates with a magnetron, also created by Sairem. This example was truly a huge help and an inflection point during the process of creation of my project. Although it was completely different to my device, it let me know where the programming should be directed.

4.2 Block diagram

Thanks to the AI4S user manual, I was ready to start programming the device. In order to explain in a detailed way, I will go through the program itself, step by step, to finally end with the final overview of the interface.

4.2.1 Communication from the Master to the AI4S

Firstly, since the AI4S must be connected on the RS232 Modbus connector, some parameters must be written on the port settings, which will be port COM 5 of my laptop. Figure 4.4 shows the beginning of the program. “VISA Resource Name” block will specify the resource (port) the user wants to open, while the “Parameters” block includes the parameters that need to be common between the master and the slave for a serial communication. Those parameters are, as mentioned in the previous section, the baud rate, the number of stop bits and the type of parity and flow control. Moreover, within this block are included two more parameters: timeout and retries. The first one needs to be, at least, 2 seconds (2000 ms). Each command or request from the master to the slave (or vice versa) must be sent every 2 seconds so no external communication fault will appear.

The Plasmionique Modbus Master Library, with the objective of trying more than once a Modbus transaction, created the second parameter called “Retries” for anytime an error occurred during the communication between the master and the slave. However, as it has been mentioned in the Design section, no error-check technique has to be set (parity and handshaking), so the role of this parameter is actually useless. I left this value be by default, which is one more retry.

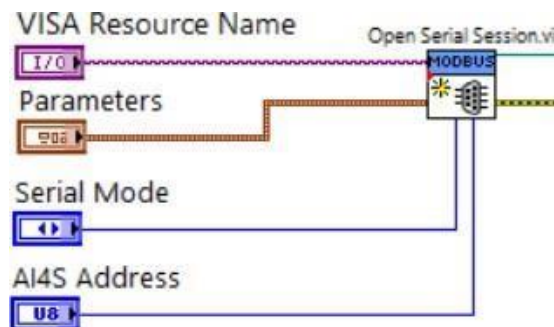


Figure 4.4 - Beginning of the program

Following the program, the next step is to send requests to the device itself. Firstly, it is necessary to write a single register (Function code 0x06) to the device. The role of this register will be indispensable in many ways. At first, thanks to it, it will let the user decide which mode control, either automatic or manual mode, will the tuner implement in channels X and/or Y. Manual mode will let decide the user the position of the stubs from one channel or both at the same time, while the automatic mode will automatically adapt the load to the source depending on the conditions.

Secondly, the user will be able to stop the request sent by the master on channels X and/or Y. Thirdly, a RESET request will be allowed in case some faults occur during the program execution. The role of the RESET button is quite important, and it will be explained later its significance.

Finally, the user will be able to change the gain of each channel by pressing the “Gain Modification X (or Y)” button. The function of this button will also be explained later.

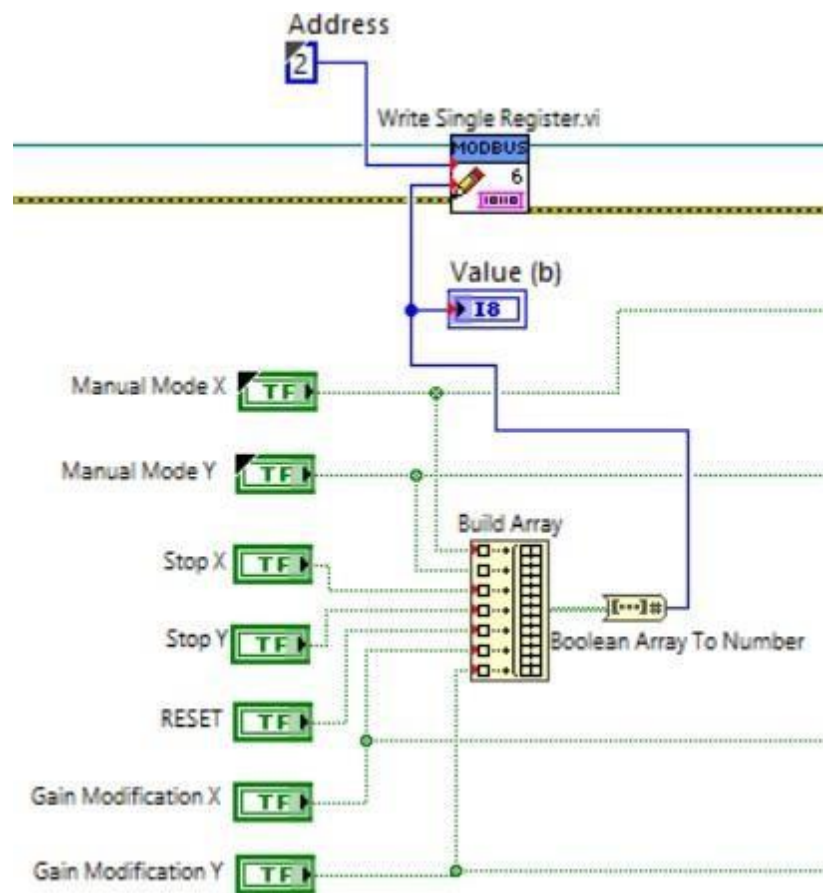


Figure 4.5 - Continuation of the program. From now on, the program is inside a while loop

As it can be seen in the previous Figure, the Boolean values are built in an array, to finally convert it into a number (Value) which will show the user which buttons have been pressed. Therefore, if “Manual Mode X” and “RESET” buttons have been pressed, the controls part of the user interface will return the “0010001” value. The “Value” indicator does not play any important role during the programming of the device; it is just a way I decided to show how the master sends requests to the slave in a binary form. To communicate with the device, the starting address for this register is 2 [16, page 22].

Depending on which mode control has been selected, the flow of the programming will be directed through one road or another. Figure 4.6 and Figure 4.7 show the programming of each mode. A case structure has been used for it. Depending on whether Manual Mode (for channel X and/or Y) has been pressed or not (True or False) by the user, the aspect of the program is as it follows:

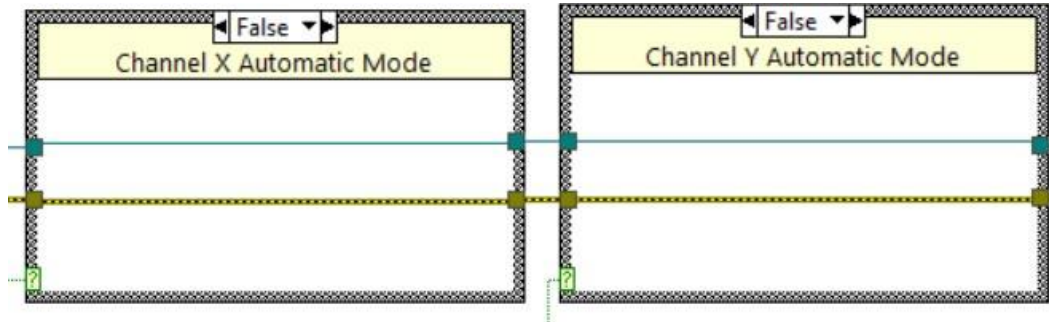


Figure 4.6 - Automatic mode

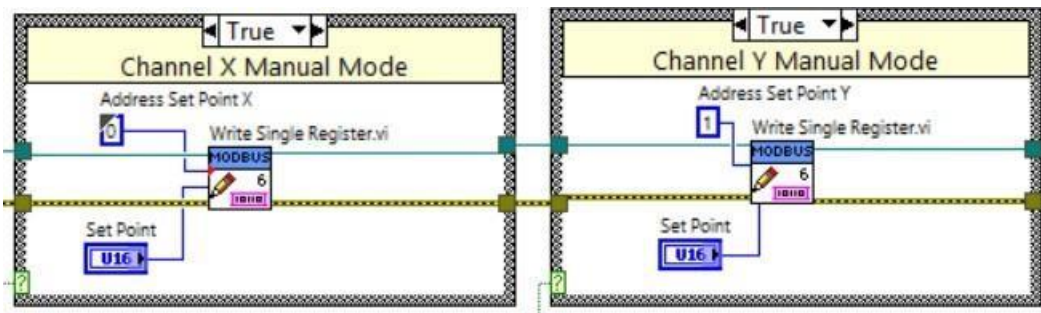


Figure 4.7 - Manual mode

The manual mode will let the user set a desired position to the stubs, being 0 the lowest position set point, 5.000 the middle position set point and 10.000 the highest position set point. For it, a single register is written in the device with starting addresses 0 and 1 for channels X and Y, respectively. In the automatic mode, the positions of the stubs are automatically set depending on the needs of the load.

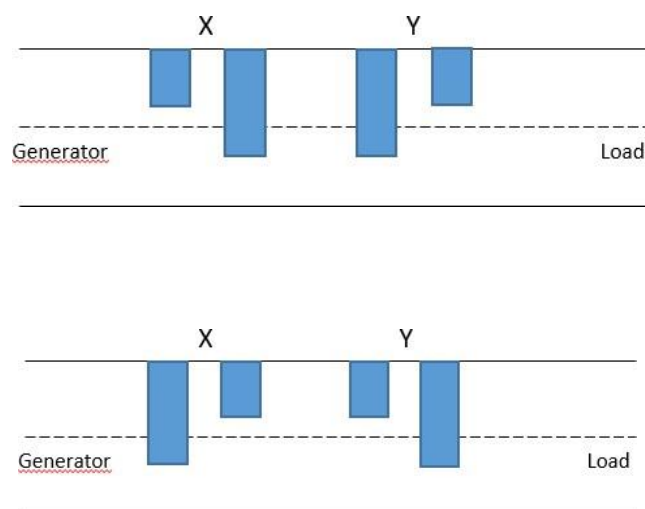


Figure 4.8 - Representation of the high and low position of the stubs, respectively. The positions in both channels are asymmetric

The next step is to let decide the user whether change the gain of each channel or not. To do so, as in the last step, a case structure is used to make this decision possible. Moreover, a single register is written in the device with starting address 9 for both channels. The role of the gain is to change the speed of the automatic matching, so it is not necessary to change its default value (1.000) for manual operation. In case the user wants to change the gain for the automatic mode, its maximum factory value is 2.000.

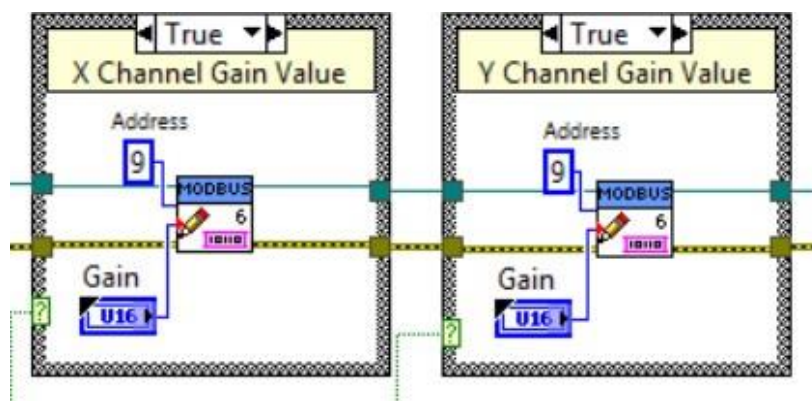


Figure 4.9 - Modification of the gain only for automatic mode

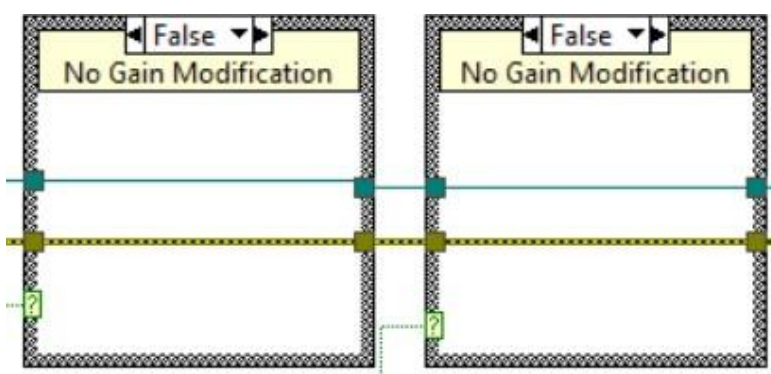


Figure 4.10 – No change in the gain for manual mode

4.2.2 Communication from the AI4S to the Master

We have sent all the requests the master can do, so now it is time for the tuner to explain us its actual situation. During the manual mode, we set a certain position to the stubs, so the tuner must show us the real position of the stubs. For that, we need to read the number of holding register (Function code 0x03). Figure 4.10 shows the way to ask the device to provide us the position readout of the stubs from channels X and Y by communicating through addresses 100 and 101, respectively.

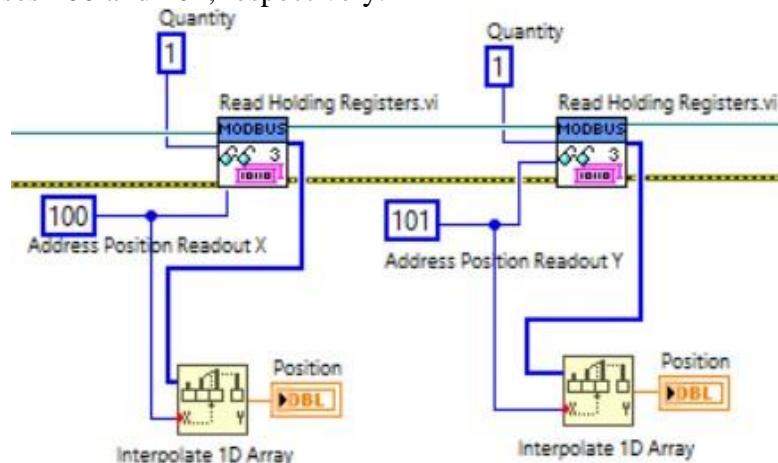


Figure 4.11 - Lecture of the real positions of the stubs

As I mentioned before, the user can set the position set point from zero up to 10.000, but, as it will be shown later, unfortunately the AI4S will not return such low and high values. If the user sets 10.000 position, the tuner will show on screen, approximately, a position of 8.000, being it the real highest limit for the stubs. Moreover, if the user writes zero as the desired position for the stubs, the tuner will adjust the stubs until the 2.000 position. This means 2.000 is the lowest limit established for the stubs.

Once read the real position of the stubs, we need to understand its meaning regarding the circuit itself. That is the role of the discriminator of each channel. Through addresses 102 and 103, channels X and Y, respectively, will show how far are they from adaptation.

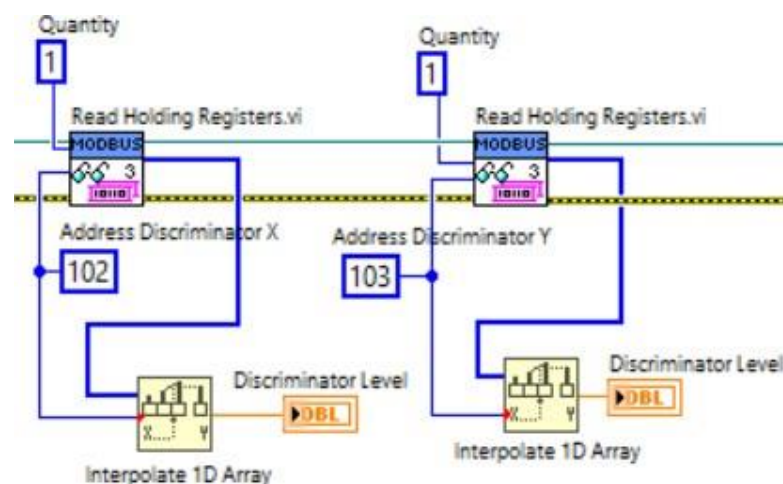


Figure 4.12 - Discriminator level

The discriminator readout can go from 0 to 10.000, being 5.000 the middle point and whose value will mean either the adaptation is properly done, or no radiofrequency has been detected during the execution of the program. Although it is not possible to deduce real parameters from these values, the discriminator value can be considered to be as the reflection coefficient. Hence, 5.000 value for the discriminator can be understood as zero for the reflection coefficient, that is, a perfect match.

In order to check if everything is going as expected either inside and/or outside the device, it is necessary to read any fault that may happen during the communication. For that, starting by address 104, a byte is read, and each bit will show a different fault in case any problem occurs.

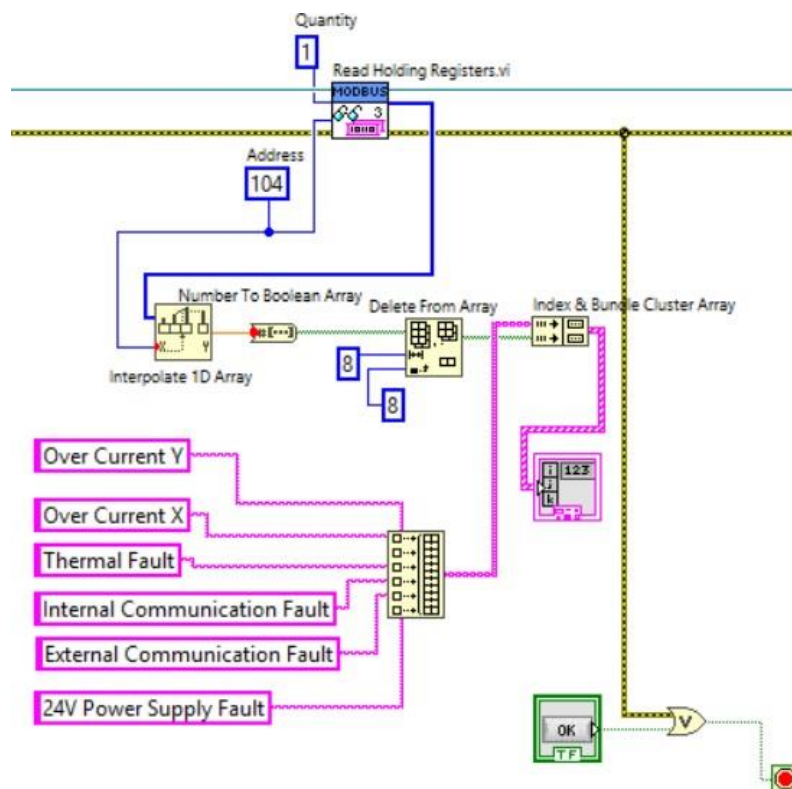


Figure 4.13 - Read all faults

Therefore, bit 0 of the message, for example, will indicate if an over current happens in channel X, while bit 1 will indicate whether there is being an over current on channel Y or not. If an over current is detected, the motor of the tuner will be stopped for 5 seconds and restarted automatically, otherwise the device might be in risk of damage because of an excessive generation of heat. The next bit talks in terms of temperature, so, in case the light of the fault is turned on, the device will auto restart only when the temperature returns low.

Bits 3 and 4 show any fault according to communication. The internal fault can be fixed by switching OFF and ON to clear the default, while the external fault, which will mean the inexistence of communication on the network during more than 1 second, must be fixed by sending a RESET request, as well as for the 24V power supply fault. The instructions to correct each of the faults are also written next to the interface, as it is shown in the next Figure.

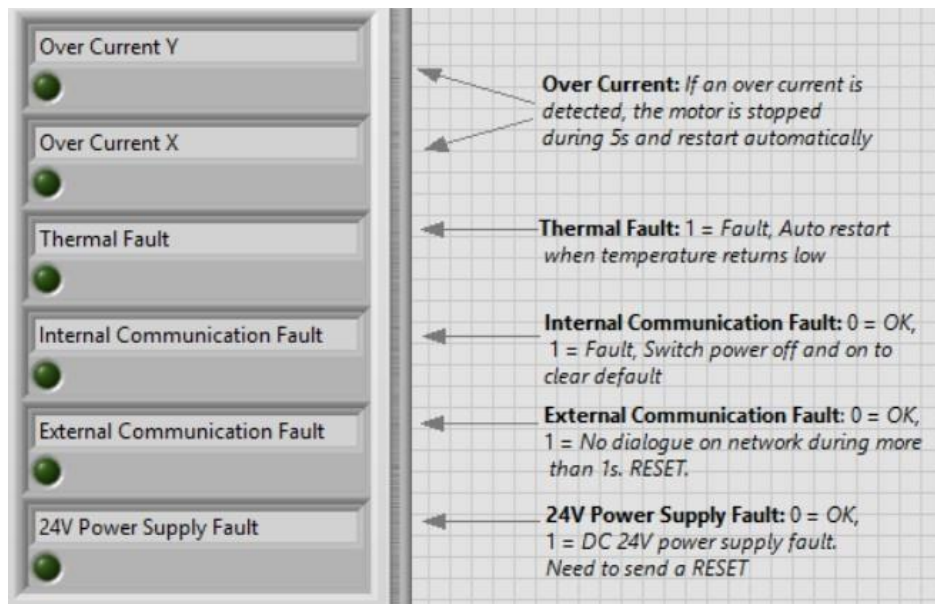


Figure 4.14 - Instructions to correct each of the faults

Finally, the while loop will only be finished if the user presses the stop button or an error occurs during the programming. These two conditions are the inputs of a logical OR which will decide whether the execution must be stopped or not.

Lastly, on the condition that an error occurred, or the user pressed the stop button, the Modbus communication between the device and the master needs to be closed, as well as the serial port. The next Figure represents the last part of the programming, being the first block the one that closes the Modbus session and the second block returns a description of the error that may have happened during the execution of the program.

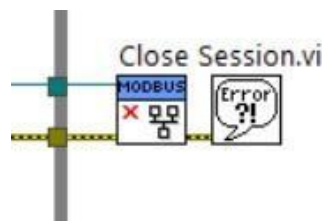


Figure 4.15 - End of the program

5 Integration, tests and results

Before starting the communication between the computer and the tuner, some previous start-up precautions need to be explained. Firstly, it is necessary to connect the tuner to a 24V power supply. Once connected to the 24V, the AI4S checks if there is communication through the RS232 gateway. In case there is no communication, the AI4S is automatically started in the automatic control mode and it will start to adapt the load at about 10 seconds after the power up.

It is important to take into account the type of load that it is going to be adapted. If this load is difficult to be tuned, such as the load of the sulfur plasma bulb, which is variable depending on its temperature, it is mandatory to first set the stubs of the AI4S in the neutral position (5.000). This process needs to be done before turning on the microwaves created by the microwave oven.



Figure 5.1 - Representation of the stubs in the neutral position (5000)

After setting the neutral position, the user can turn the microwaves on. The next step is to check whether the reflected power is higher than the allowed level by the tuner. According to the user manual, depending on the maximum power of the AI4S, the maximum reflected power allowed will be different. The maximum power depends on the generator being used by the user. In our case, the magnetron used is a 1kW power microwave generator [2], so the maximum VSWR is 4.2 (provided by the user manual) at 1kW. The reflected power will be measured by the next component of the circuit, the directional coupler, although, given the maximum VSWR, we can calculate useful values. In the next table appear the results for the maximum reflection coefficient, reflected power and return loss by using equations (2), (3), (4), (5) and (6) from the Section 2 of the report. Once the reflected power is lower than the maximum allowed level the AI4S can be started.

Table 4

Input Power (Microwave generator)	1kW
Max. VSWR	4.2
Max. Reflection coefficient	0.615
Return Loss (dB)	4.22 dB
Reflected Power (%)	37.8%

To test the programming of the tuner, firstly we needed to take it out from the circuit in order to have free control over it and be capable of check the positions of the stubs. Once

connected both the laptop and the tuner, I ran the program. In the absence of RF, the user can expect that the discriminator level from both channels will be 5.000 or a very similar value. The first step was to set a desired position, in the manual mode, to the stubs from channel Y. This channel responded in the way it was expected, but, unfortunately, the channel X did not answer to any request sent. Even if I did not set any desired point to the stubs from channel X (manual mode button not pressed), the return value for the position was 4.278, which should be, instead, 5.000 since the stubs, once the tuner is powered up, move to the neutral position. In addition, the programming for both channels is very similar, being different only the address.

I immediately informed about this issue to Professor Gilles Courret, who checked that, indeed, the programming was properly done. We decided to contact Sairem for any information about this problem and, after some instructions made by Thibault Gadeyne to assure the programming made by me was correct, he concluded the device was not working well, probably the error came from the electronic control part of the device. After proving the problem of the device was manufacturing defect, we packed the tuner and shipped it for revision to Sairem.

In my last days in Switzerland, the device arrived at the university, which let me check at least if my programming was working as expected. The tuner was properly repaired and ready to be used again. The next images show, in order, the real lowest, highest and neutral position of the stubs. Thankfully, the programming done was correct.

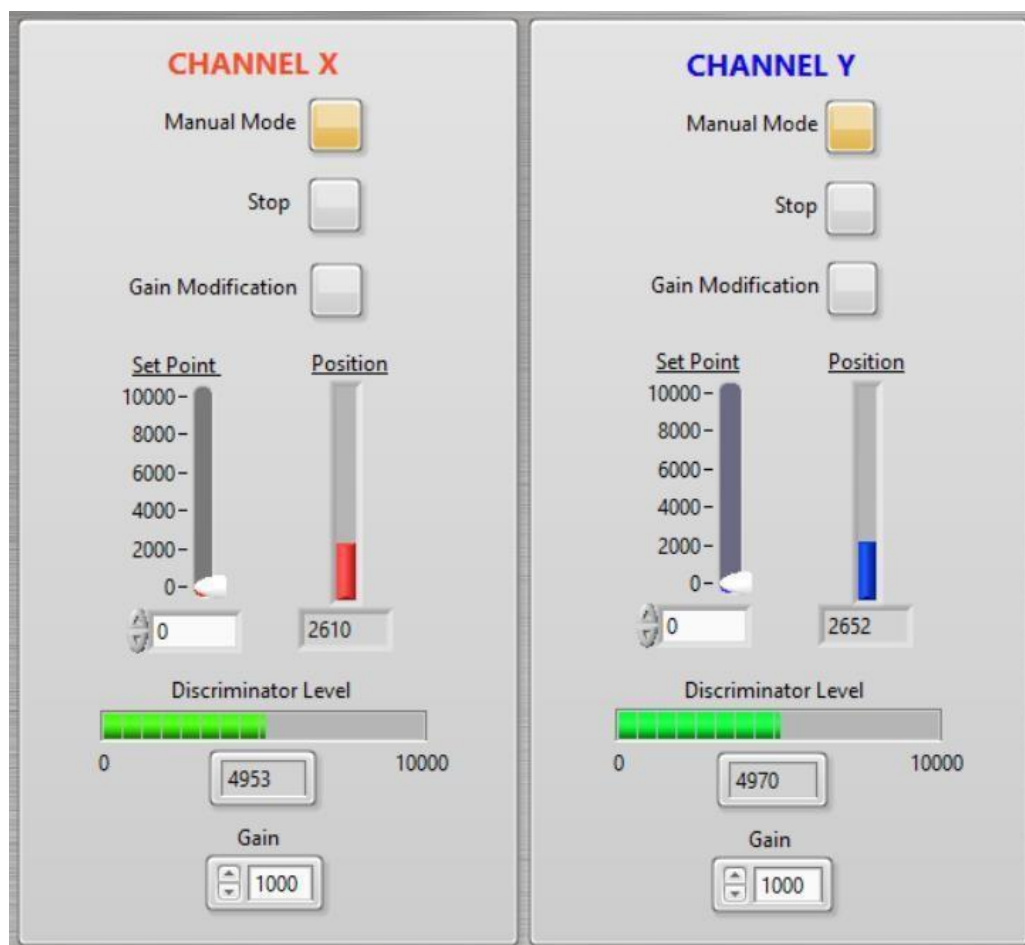


Figure 5.2 – Positions 2610 and 2652 are the real lowest positions for the stubs from channel X and Y, respectively



Figure 5.3 – Interior of the waveguide. Stubs are in their lowest position possible

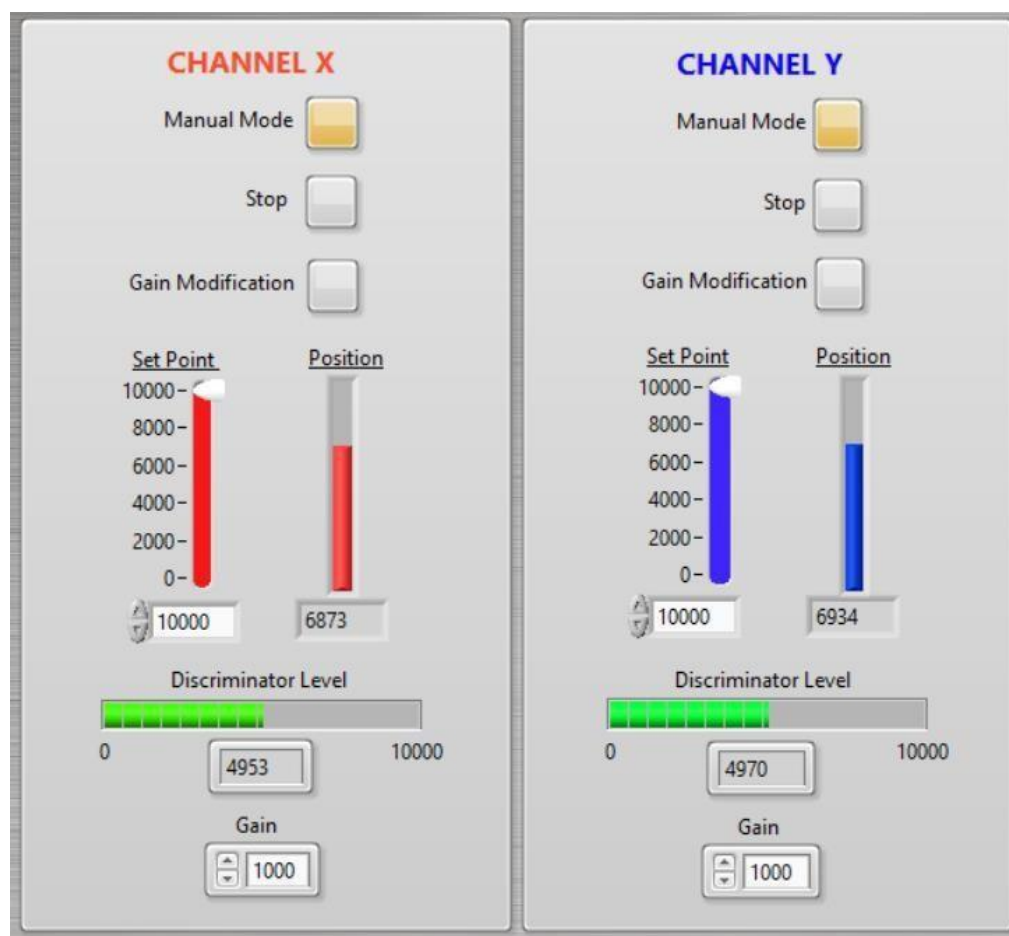


Figure 5.4 – Positions 6873 and 6934 are the real highest positions for the stubs from channel X and Y, respectively



Figure 5.5 - Interior of the waveguide. Stubs are in their highest position possible

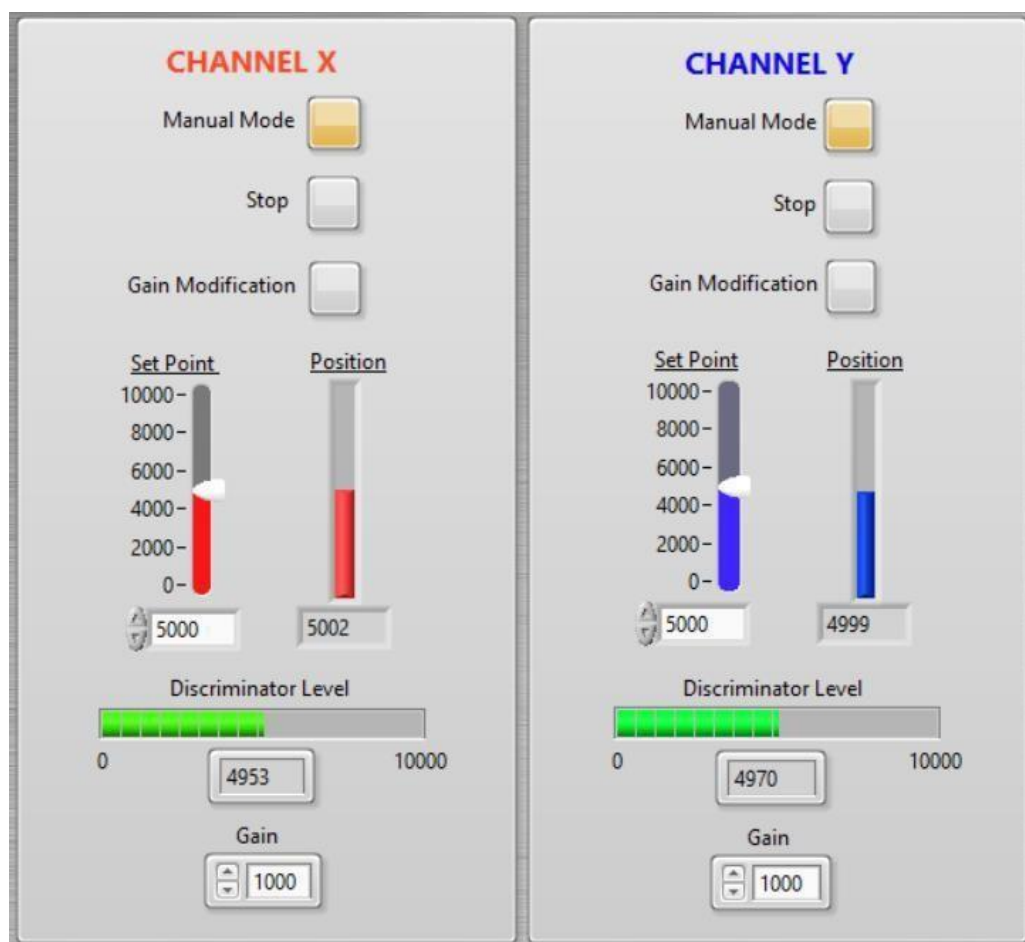


Figure 5.6 – Neutral position of the stubs



Figure 5.7 - Interior of the waveguide. The stubs are in the central position

6 Conclusions and future work

6.1 Conclusions

Unfortunately, due to the manufacturing defect of the device, I was not able to make any useful conclusions in the microwave circuit. I could not demonstrate if the new tuner with four-stubs is more suitable than the manual three-stub tuner in terms of a longer duration of the spherical shape of the lightning of the sulfur plasma bulb. However, the main conclusion of my project is the achievement of a well-implemented user interface to control the device in a proper way.

6.2 Future work

I encourage future users to play with the user interface with the presence of microwaves travelling through the waveguide of the tuner, as I had no time to check it by myself. Since the user interface works, it is time to use the tuner for its real purpose. Once the user has gotten familiar with the functioning of the interface, it is necessary to read carefully the previous precautions written in the Integration, tests and results section of the report to avoid any possible problem on the device (e.g. reflected power higher than the allowed level). Also, during the communication with the tuner, it is probable that some of the mentioned faults in the Development section turn on, so it is important to read carefully the instructions for a correct operation.

After that, with the high frequency circuit fully assembled, the user should start with the four stubs in the neutral position (5.000) and, at first, in the automatic mode control. The user can check what fits better for the duration of the lightning, whether automatic or manual mode control. In case of a manual mode control, with the light of the bulb on, the user should try different combinations on the position of the four stubs and check with which combination the duration of the lightning lasts longer in the spherical shape, keeping an eye on the possible melting of the bulb. I also recommend future users to make some graphics to compare between the position of the stubs and the discriminator level value (reflection coefficient) in order to conclude with which positions the discriminator is closer to 5.000.

Once concluded which of the two tuners is more preferable to be used in the microwave circuit, the next step might be to use an infrared camera which allows to detect the current temperature of the sulfur plasma bulb at any time. Although during my stay at the HEIG-VD I used the infrared camera Ti450 PRO from Fluke company and controlled it with the SmartView software, it would be more suitable to use an infrared camera, such as the TiX560, which allows to control it by LabVIEW. The possibility of controlling it by LabVIEW can be very beneficial for the progress of the sulfur plasma bulb project. I recommend future users to include a third section in the user interface created by me in which the user could check at any time the minimum and maximum temperatures registered by the camera. LabVIEW includes graphics and charts indicators that permit to show any image taken by the infrared camera. This work would surely help to achieve a better control on the circuit.

References

- [1] G. Courret, L. Calame, M. Croci, P. Egolf, A. Meyer, “Environmental friendly high efficient light source”, Haute École d’Ingénierie et de Gestion du Canton de Vaud (HEIG-VD), Sept 2006
- [2] Muegge GMBH, “Magnetron 2M244-M16, for industrial use, S-band (2450 MHz)”, Datasheet, Industrial Microwave + Plasma Systems
- [3] G. Courret, P. Nikkola, S. Wasterlain, O. Gudozhnik, M. Girardin, J. Braun, S. Gavin, M. Croci, P. W. Egolf, “On the plasma confinement by acoustic resonance”, Haute École Spécialisée de Suisse Occidentale (HES-SO), 2017
- [4] J. P. Koulakis, Alexander L., F. Thornton, S. Putterman, “Magnetron Coupling to Sulfur Plasma Bulb”, IEEE Transactions on Plasma Science, Vol. 45, pp. 1-2, November 2017
- [5] J. Córcoles-Ortega, “Tema 1. Conceptos básicos de electrónica de Electrónica de Comunicaciones”, Electrónica de Comunicaciones, Escuela Politécnica Superior, Universidad Autónoma de Madrid, 2017
- [6] David M. Pozar, “Microwave Engineering”, 3rd Edition, 2005, pp. 49-70
- [7] David M. Pozar, “Microwave Engineering”, 3rd Edition, 2005, pp. 228-240
- [8] “Introduction à Modbus”, National Instruments, Août 4, 2014, <http://www.ni.com/white-paper/7675/fr/#top>
- [9] “Serial Communication”, Sparkfun, Colorado, United States, <https://learn.sparkfun.com/tutorials/serial-communication/all>
- [10] G. Courret, L. Calame, J. Croisier, M. Croci, “Sustentation of a thermal plasma by acoustic resonance as a light source of high efficiency”, Haute École Spécialisée de Suisse Occidentale (HES-SO), Sept 2008
- [11] National Instruments, 2010, 30 July, “Getting Started with LabVIEW”, <https://www.youtube.com/watch?v=ZHNiKyYzrPE&list=PLB968815D7BB78F9C>
- [12] Sairem, “Manual 3-Stub Tuner”, Datasheet, Neyron, France, <https://www.sairem.com/wp-content/uploads/2017/10/AI3SMWR340D24X25MRR2PEBEN.pdf>

- [13] N. Vaguela, "Why is impedance matching important", Huettinger Electronic Inc, Farmington, United States
- [14] B. J. Park, J. S. Choi, J. G. Ryu, "Microwave Discharge Characteristics of Plasma Lighting System", Digital Appliance Research Laboratory, Wave Application Group, Seoul, Korea, pp. 197-198
- [15] B. P. Turner, M. G. Ury, Y. Leng, W. G. Love, "Sulfur Lamps – Progress in their development", Journal of the illuminating Engineering Society, Winter 1997
- [16] Sairem, "AI4S Operation Manual, Automatic 4 stubs tuner", Datasheet, Neyron, France

Glossary

RTU	Remote Terminal Unit
λ	Longitud de onda
AI4S	Automatic 4-Stub Tuner
HEIG-VD	Haute École d'Ingénierie du Canton de Vaud
SCADA	Supervisory Control And Data Acquisition
ASCII	American Standard Code for Information Interchange
TCP/IP	Transmission Control Protocol/ Internet Protocol
CRC	Cyclic Redundancy Checksum
RS232	Recommended Standard 232
LED	Light-Emitting Diode
NI	National Instruments
PLC	Programmable Logic Controller
VI	Virtual Instrument
USB	Universal Serial Bus
COM	Communication Port
VISA	Virtual Instrument Software Architecture
RF	Radiofrequency
VSWR	Voltage Standing Wave Ratio