SPECTROMETER FOR RAMAN AND PHOTOLUMINESCENCE SPECTROSCOPY WITH VARIABLE SPECTRAL RESOLUTION FOR MARINE APPLICATIONS

Pavić, Ivan

Doctoral thesis / Disertacija

2022

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Split, Faculty of Maritime Studies / Sveučilište u Splitu, Pomorski fakultet**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:164:875124

Rights / Prava: In copyright/Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-05-09**

UNIVERSITY OF SPLIT

dabar
DIGITALNI AKADEMSKI ARHIVI I REPOZITORIJI

FACULTY OF MARITIME STUDIES

Ivan Pavić

SPECTROMETER FOR RAMAN AND PHOTOLUMINESCENCE SPECTROSCOPY WITH VARIABLE SPECTRAL RESOLUTION FOR MARINE APPLICATIONS

DOCTORAL THESIS

Split, 2022

FACULTY OF MARITIME STUDIES

Ivan Pavić

SPECTROMETER FOR RAMAN AND PHOTOLUMINESCENCE SPECTROSCOPY WITH VARIABLE SPECTRAL RESOLUTION FOR MARINE APPLICATIONS

DOCTORAL THESIS

Supervisor: Joško Šoda, Ph.D.
Co-supervisor: Mile Ivanda, Ph.D.

Split, 2022

DATA ON EVALUATION AND DEFENCE OF THE DISSERTATION

Doctoral thesis evaluation committee:

1   prof. Danko Kezić, Ph. D., president, full professor, Faculty of Maritime Studies, University of Split
2   Petar Matić, Ph. D., member, assistant professor, Faulty of Maritime Studies, University of Split
3   Veres Miklös, Ph. D., member, senior research fellow, Wigner Research Centre for Physics, Budapest Hungary

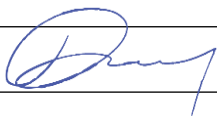Doctoral thesis defense committee:

1   prof. Danko Kezić, Ph. D., president, full professor, Faculty of Maritime Studies, University of Split
2   Petar Matić, Ph. D., member, assistant professor, Faulty of Maritime Studies, University of Split
3   Veres Miklös, Ph. D., member, senior research fellow, Wigner Research Centre for Physics, Budapest Hungary

Doctoral thesis defended on: 30.09.2022.

STATEMENT ON DOCTORAL THESIS ORIGINALITY

I declare that my doctoral thesis <u>is the original result of my work</u> and that it <u>clearly states and cites the references of contributions and papers by other authors</u>. I also declare that I have fulfilled all the conditions for initiation of the procedure of evaluation and defense of the doctoral thesis, including those related to the publishing and presentation of papers from the doctoral thesis research area.

I declare that the proposed doctoral thesis has been formatted according to the Instructions for doctoral thesis formatting.

(first and last name of doctoral candidate, signature)

The Doctoral thesis defence committee for *Ivan Pavić* certifies that this is the approved version of the

SPECTROMETER FOR RAMAN AND PHOTOLUMINESCENCE SPECTROSCOPY WITH VARIABLE SPECTRAL RESOLUTION FOR MARINE APPLICATIONS

President:	prof. Danko Kezić, Ph. D., full professor	Faulty of Maritime Studies, University of Split
Member:	Petar Matić, Ph. D., assistant professor	Faulty of Maritime Studies, University of Split
Member:	Veres Miklös, Ph. D., senior research fellow	Wigner Research Centre for Physics, Budapest Hungary

ACKNOWLEDGEMNT

ABSTRACT

The maritime industry includes all stationary and floating infrastructure needed for maritime transport and trade. Therefore, it is directly related to the marine environment. However, it is necessary to follow the regulations issued by the International Maritime Organization (IMO) to reduce the damage to the marine environment and reduce the maintenance caused by the marine environment. One of the techniques used for this purpose is Raman and photoluminescence spectroscopy (PL). Raman and PL spectroscopy are important analytical tools in materials science that provide information on the vibrational and electronic properties of molecules and crystals. This dissertation presents the results of a novel approach to Raman and PL spectroscopy that takes advantage of variable spectral resolution by using zoom optics in a monochromator in front of the detector. The results show that the spectral intervals of interest can be acquired at different zoom factors, significantly reducing the acquisition time and changing the spectral resolution for different zoom factors. The smallest spectral intervals acquired at the maximum zoom factor yield higher spectral resolution suitable for the Raman spectroscopy. In contrast, larger spectral intervals acquired with the minimum zoom factor yield the lowest spectral resolution suitable for the PL spectroscopy. The proof-of-concept was demonstrated using a zoom lens with a zoom factor of 6. Then prototype spectrometer was demonstrated using a zoom lens with a zoom factor of 18. Special electronics were developed to allow automated Raman and PL spectra acquisition. The graphical user interface (GUI) was developed for spectra acquisition. The resulted spectra obtained were compared to those obtained with a high-quality commercial spectrometer. The comparison was performed on several different materials for Raman and PL and the statistical analysis of variance (ANOVA) was used to evaluate the spectra obtained. The results show that such spectrometer could be an efficient and fast tool for the search of Raman and PL bands of unknown materials and the subsequent spectral acquisition of the spectral interval of interest with a suitable spectral resolution. Therefore, it can be used to develop new materials used in the marine industry to reduce damage caused by the harsh marine environment.

Key words: Maritime industry; Marine environment; Raman spectrometer; photoluminescence; spectral resolution; zoom lens

CONTENTS

1. INTRODUCTION

Nowadays, 80% of goods are transported by commercial maritime transport [1], which can significantly impact the marine environment. However, the marine environment affects all maritime infrastructures, including both floating and stationary objects. Therefore, in the field of maritime research, it is necessary to analyze the structural and electronic properties of materials to eliminate the damage caused by the marine environment. Scientists have developed various qualitative and quantitative analytic methods, including Raman and photoluminescence (PL) spectroscopies [2–4]. Raman effect is an inelastic photon scattering process in which a transfer of energy between the light and the system corresponds to the energy of a particular vibrational state of the material. The result of Raman scattering is a Raman spectrum, which is unique to the individual molecule or crystal and is therefore considered and studied as a "fingerprint" of the specific material [5],[6]. In recent times, since the signal of Raman scattering is very weak, a number of new techniques for Raman scattering enhancement have been developed and deployed [7–14]. When the luminescent molecule is excited by interaction with a photon of electromagnetic radiation, fluorescence or phosphorescence may occur. Furthermore, if the electromagnetic energy is released immediately, it is a fluorescence process; if the release of electromagnetic energy is delayed it is a phosphorescence process. Both fluorescence and phosphorescence are two forms of PL [15],[16].

Spectra obtained by the Raman spectroscopy are usually accompanied by a broad background in which the intrinsic PL dominates the sample [17]. In the case of a moderate or weak PL signal, it could be recorded with a Raman spectrometer as a superposition on the Raman spectrum. However, if the PL signal is stronger, the recording process consists of a lot of spectral intervals (defined by the physical size of the CCD) combined into the single superimposed PL and Raman spectrum. This process is time-consuming and could produce unwanted artefacts depending on how the spectral intervals are combined into the single PL spectrum. In the case of a very strong PL signal, it completely masks the Raman signal [18],[19]. Therefore, the Raman spectrometer could be a great tool for providing information on the PL phenomenon. Besides, when the Raman and PL spectral information [20–23] are both taken from the same material or microscopic part, they could provide complementary information on the material's vibrational and electronic properties. Also, combining Raman and PL techniques allows the detection of specific structural features of complex chemical and biological samples [24],[25]. Raman and PL spectroscopy is used in biology, pharmaceuticals,

chemistry, environmental analysis, marine, electrical engineering, geology, and many other fields [26–34]. Several attempts [35–39] have been made to build cost-effective laboratory-based and portable Raman spectrometers of low spectral resolution. Their main purpose is to record Raman spectra for fast inspection of materials. However, a middle or high-resolution Raman spectrometer is needed for the observation of the splitting of narrow spectral lines and/or small changes in the shift and width of the observed spectral lines. Furthermore, the most sensitive detectors are used in many commercial laboratory-based Raman spectrometers due to weak Raman scattering. These detectors could also be very useful in detecting PL signal, even for the cases of a moderate or weak PL signal which is below the threshold of detection of the conventional PL spectrometers.

This thesis presents a novel concept for developing a Raman spectrometer with a variable spectral resolution by implementing zoom optics into the monochromator of the spectrometer. Within this approach, it is possible to measure both the large spectral intervals and fast low resolution acquisitions on the one end and high resolution, detailed spectral images on the other end. The large spectral intervals are obtained with the smallest zoom factor (ZF) of the inserted optics. This is suitable for recording PL spectra and the fast inspection of the whole Raman spectrum since the same amount of photons is distributed over a smaller number of pixels on the CCD. For the acquisition of detailed Raman spectra, a high spectral resolution is required. This is usually achieved by reducing the slit width, increasing the number of grooves on a diffraction grating, and increasing the focal length of the focusing mirror to the detector. Contrary to the usual solutions, the degree of spectral resolution in the proposed spectrometer can vary by the ZF of the focusing zoom lens to the detector [40],[41],[42].

The dissertation is divided into eight chapters. After the introduction, Chapter 2 presents the theoretical basis of Raman and PL spectroscopy and the equipment used. Further, the chapter describes the basic setup of the current spectrometers, followed by an overview of Raman and PL spectrometry in maritime applications. Chapter 3 describes the dissertation's primary goal and research plan, which addresses the idea of developing Raman and PL spectrometers with variable spectral resolution. Preliminary research is described in Chapter 4, which presents the first prototype developed and its results. Chapter 5 describes the methodology and components used to develop a second prototype. The results obtained with the second prototype and their evaluation are described in Chapter 6. Finally, an invention applied for the patented invention is described in Chapter 7, followed by a conclusion in Chapter 8.

2. RAMAN AND PHOTOLUMINESCENCE SPECTROSCOPY

This chapter covers the basic theory of Raman spectroscopy and used instrumentation, followed by the photoluminescence (PL) spectroscopy theory. Also, a review of a Raman and PL spectroscopy for maritime application is described.

2.1. Raman spectroscopy

Raman spectroscopy is based on inelastic incoherent scattering of light, discovered in 1928 by Indian physicists Sir Chandrasekhara Venkata Raman and Kariamanickam Srinivasa Krishnan. When a beam of monochromatic light is incident on a material, a series of new spectral lines called Raman lines are observed in the scattered light in addition to the incident frequency [43]. Raman spectroscopy is a technique for observing vibrational, rotational, and other low-frequency states in a system. Raman scattering can be used for analyse surfaces, thin films, powders, all types of solutions, gases, etc. [44]. The Raman spectrum is considered as a single molecule "fingerprint" [45]. Compared to the elastic light scattering, Raman scattering has a relatively low intensity, as only 1 in 10^6-10^8 photons is scattered in this way in comparison to the elastic scattered light [46]. However, thanks to modern technology and the development of lasers and sensitive detectors, this process can now be used for spectroscopic studies of various materials [47–49]. In Raman scattering, the photon energy changes by an amount equal to the energy of the individual vibrational transitions (Figure 1). In the Stokes process, which corresponds to absorption, the scattered photon's frequency decreases due to the transfer of energies from the excited photon to the molecule. In the anti-Stokes process, which corresponds to emission, the scattered photon's frequency increases due to the absorption of energy released by the molecules during the transition to the ground state [50]. However, most photons retain their original frequency and scatter elastically in the so-called Rayleigh scattering.
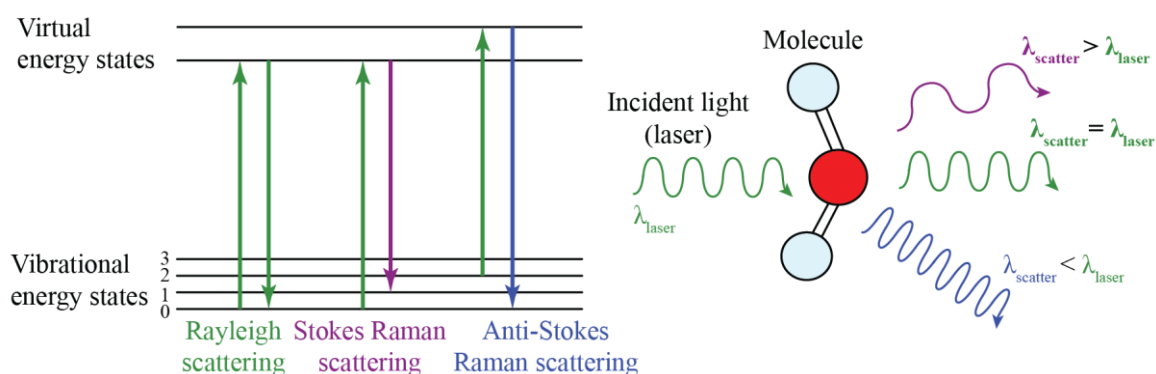
Figure 1. Schematic diagram of a Rayleigh, Stokes and anti-Stokes Raman scattering (left) and difference in wavelength for each of the scattering (right).

Raman spectrometers were developed to record Raman spectra. Therefore, the basic setup of Raman spectrometers consists of an excitation source, a sampling unit, and a detection part, as shown in Figure 2. Depending on the sampling unit, Raman spectrometers can be used for micro and macro Raman spectroscopy. The sampling unit for micro-Raman spectroscopy consists of a standard optical microscope, which is used to obtain Raman spectra from the microscopic sample analysis. In macro-Raman spectroscopy, the excitation source is aimed directly at the sample, and the scattered light is collected with a detection unit. Monochromatic light is used as the excitation source i.e., a laser. The detection unit is a spectrometer with a Czerny-Turner configuration consisting of a diffraction grating, two collimating mirrors, and a detector (see the right side of Figure 2 where C1 and C2 represent collimating mirrors).
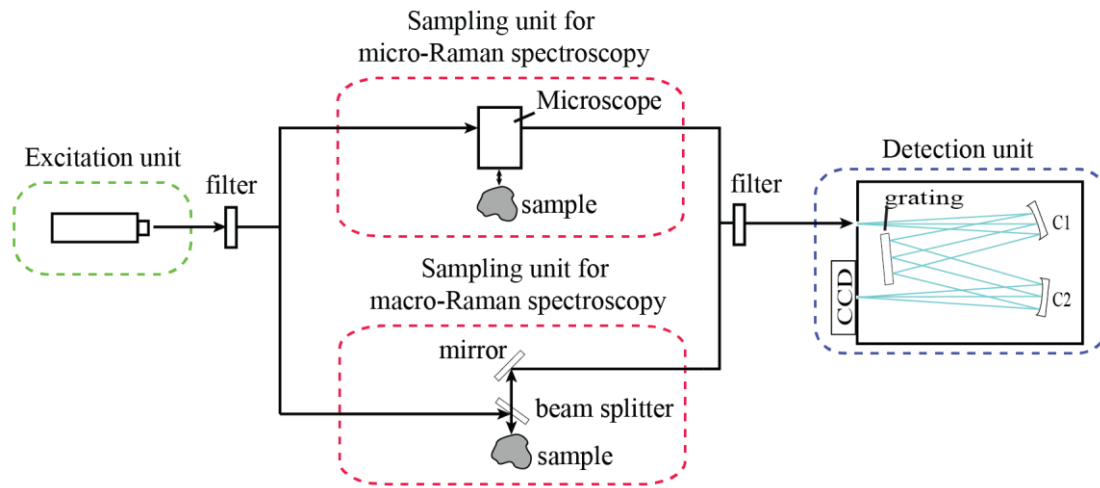
Figure 2. Schematic diagram of a fundamental Raman spectrometer design. Note, that two different systems are used for micro (mid-top) Raman spectroscopy and macro (mid-bottom) Raman spectroscopy.

In addition, the components of a Raman spectrometer can be coupled using optical fibers, which increases the instrument's flexibility, especially for *in situ* monitoring. Since Raman scattering is a weak phenomenon and can be masked by the fluorescence of a sample, lasers with a different wavelength are used as excitation sources. For example, some samples (especially "organic" or "biological") are more fluorescent, so using a laser with a wavelength of 523 nm can promote fluorescence. To avoid this, lasers in the red region at 633 nm or in the near-infrared (NIR) at 785 nm can better detect Raman scattering [41]. Therefore, the choice of a laser is crucial because it directly affects the quality of the results, i.e., the quality of the spectrum and the spatial resolution. However, many different lasers are used in Raman spectrometers (Table 1).

Table 1. Different lasers and their technology used in Raman spectrometers [41].

Wavelength [nm]	Spectrum Range	Technology
229	Ultra Violet (UV)	Gas laser (Ar)
244	UV	Solid-state (tripled Nd: YAG)
257	UV	Gas laser (He: Cd)
266	UV	Solid-state (quadrupled Nd: YAG)
325	UV	Gas laser (doubled Ar)
355	UV	Gas laser (doubled Ar)
364	UV	Gas laser (doubled Ar)
405	Violet	Diode laser
442	Blue	Gas laser (He: Cd)
457	Blue	Gas laser (Ar) and Solid-state laser
473	Blue	Solid-state laser
488	Blue	Gas laser (Ar) and Solid-state laser
514	Green	Gas laser (Ar) and Solid-state laser
532	Green	Solid-state (frequency-doubled Nd: YAG)
594	Orange	Solid-state laser
633	Red	Gas laser (He: Ne) and diode laser
638-640	Red	Diode laser
660	Red	Solid-state laser
671	Red	Solid-state laser
691	Red	Diode laser
785	Near Infra-Red (IR)	Diode laser
830	Near IR	Diode laser
980	IR	Diode laser
1064	IR	Solid-state (Nd: YAG)

The sampling unit for micro-Raman spectroscopy is an optical microscope. It can be a normal microscope where the focus on the sample is achieved by moving the sample up and down. Open microscope without a base so that larger samples can be measured. Inverted microscope, where the objectives are under the sample, and infinity-corrected microscope, where the image of the sample is focused at infinity so that additional optics can be added to the microscope tube, increasing the flexibility of measurements for different samples [51].

The filters used in Raman spectrometers are located at a scanning unit's input beam (laser) and output beam (Raman path). Two main types of filters are used, bandpass filters (BPF), so that the sample can only receive radiation from the input laser beam and all other incoming light is rejected, i.e., for a laser with a wavelength of 532 nm, the BPF filter only allows light with a wavelength of 532 nm to pass. Since the Raman signal is very weak and masked by Rayleigh scattering, filters for the output beam can be edge or long pass filters (LPF) and notch or band stop filters (BPF). Edge filters block all wavelengths up to a certain point,

which can be useful when measuring only Stokes or anti-Stokes, while notch filters eliminate only the laser wavelength so that both Stokes and anti-Stokes can be measured.

The detection unit consists of a monochromator with a Czerny-Turner configuration [52]. Since the term monochromator refers to a device that selects only one spectral interval, measurement of the entire spectrum can be achieved by rotating the diffraction grating. The grating is the main component of the monochromator and it is a dispersive element consisting of grooves, that direct diffracted light into a different optical path. At the end of the monochromator is the detector. The detector can be a charge-coupled device (CCD), an electron-multiplying CCD (EMCCD), or a complementary metal-oxide-semiconductor (CMOS) connected to a personal computer for acquisition and analysis of Raman spectra. The result obtained by a Raman spectrometer is a spectrum of a recorded sample where the x-axis shows Raman shift in cm^{-1} and the y-axis represent intensity of recorded photons in arbitrary units. Figure 3 shows the Raman spectra of silicone (Si) recorded with the HORIBA Jobin Yvon T64000 spectrometer. Nowadays, there are many types of commercially available Raman spectrometers. Some of them are the Reinshaw inViaTM confocal Raman microscope, the B&WTEK i-Raman portable spectrometer, the Cora 100 handheld spectrometer from Anton Paar, and many others. The Raman LIDAR is also a remote sensing instrument for atmospheric aerosol analysis. It uses a laser as a light source that is directed into the atmosphere and absorbed by molecules and aerosols, resulting in backscattered light collected by a receiving telescope [53].
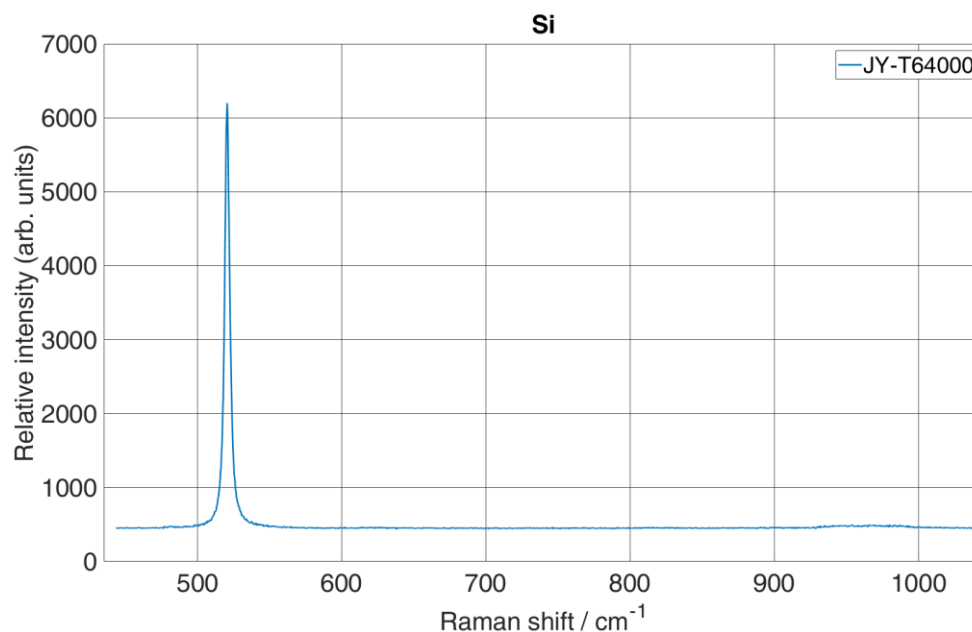
Figure 3. Raman spectra of silicon recorded with a HORIBA Jobin Yvon T64000 spectrometer.

2.2. Photoluminescence (PL) spectroscopy

Luminescence phenomena may occur when light is directed onto the sample [54]. Luminescence is an emission of light from an irradiated sample after absorption of a certain form of energy. Depending on the excitation source, there are three forms of luminescence: fluorescence, phosphorescence and chemiluminescence. Both fluorescence and phosphorescence are forms of photoluminescence (PL) where the emission of light is caused by the energy of a photon excitation, while chemiluminescence is caused by the energy of a chemical reaction [55],[56]. Figure 4 shows a diagram of a PL process. When a molecule's electron absorbs a photon's energy, it becomes excited and transitions from the ground state (S_0) to a higher electronic excited state. Since higher excited electronic states are intrinsically unstable, the excited electrons relax back to their ground state. During the relaxation process, energy is released by the emission of light. The difference between fluorescence and phosphorescence is the time in which the electromagnetic energy is released. Further, if the electromagnetic energy is released immediately, it is a fluorescence process, and if the electromagnetic energy is released with a time delay, it is a phosphorescence process [15],[16].
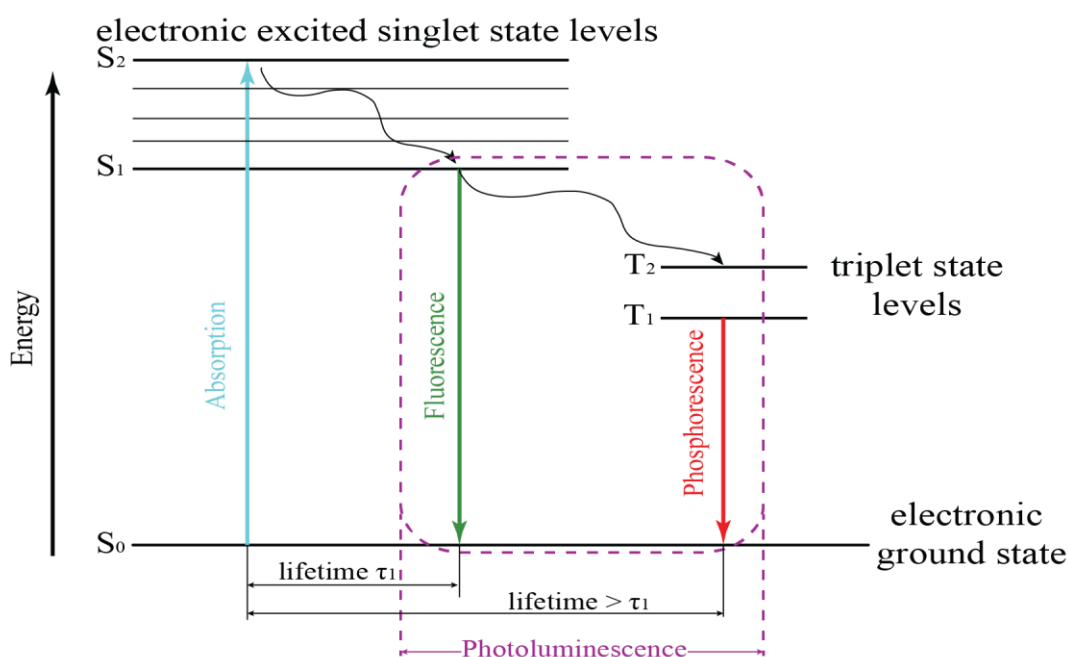
Figure 4. Simplified Jablonski diagram for photoluminescence phenomena. Note that fluorescence is emission of light when an electron is relaxed to a ground state from a singlet electronic excited state level, and phosphorescence from a triplet state level.

A common application of PL is characterizing a semiconductor's optoelectronic properties. This is done by measuring PL spectrum with a PL spectrometer or with a Raman spectrometer. The difference is that Raman spectrometers have a higher spectral resolution and the acquisition of a PL spectrum, which is spectrally much broader, takes a longer time. When recording PL spectra, the output of a PL spectrometer or Raman spectrometer is a spectrum where the x-axis represents the wavelength in nanometres, and the y-axis represents the intensity in arbitrary units. Figure 5 shows PL spectra of a Schott 550 glass recorded with a HORIBA Jobin Yvon T64000 spectrometer.
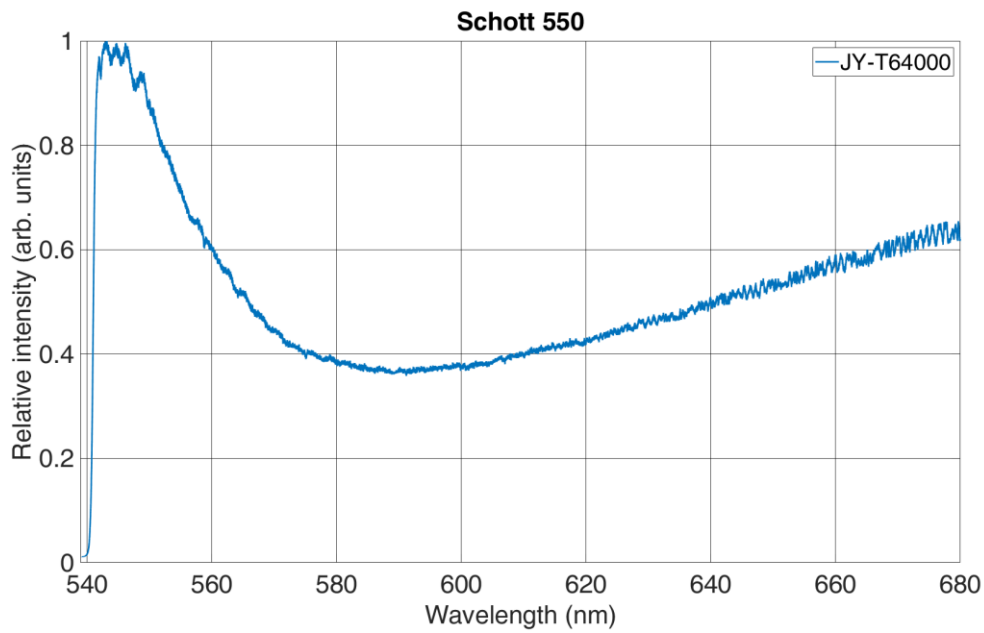
Figure 5. Raman spectra of a Schott 550 glass recorded with a HORIBA Jobin Yvon T64000 spectrometer.

2.3. Raman and PL for maritime applications

In this chapter a review of Raman and PL spectroscopy application in maritime will be discussed.

2.3.1. Microplastic

Since plastics have entered the consumer realm, a major global pollution problem has begun. It is estimated that each person accounts for 35 kg of the total annual production of plastics [57]. Part of this waste is not disposed of or recycled, resulting in pollution, i.e., in the oceans, seas, freshwater, etc. Raman spectroscopy is a great analytical tool for detecting microplastic (MP) smaller than 50 μm [58]. The Raman spectra of detected polymers are compared with the commonly known composition's Raman spectra, shown in Table 2.

Table 2. Most recurring polymers in environment [59].

Polymer	Abbreviation	Application
Expanded polystyrene	EPS	Bait boxes, floats, cups
Low density polyethylene	LDPE	Plastic bags, bottles, gear, cages
High density polyethylene	HDPE	Plastic bags, bottles, gear, cages
Polypropylene	PP	Rope, bottle caps
Polypropylene terephthalate	PET	Bottles, ger
Styrene butadiene rubber	SBR	Car tyre
Polyamide	PA	Gears, fish farm nets, rope
Polystyrene	PS	Containers, packaging
Polymethyl methacrylate	PMMA	Insulation, packaging
Polyvinylchloride	PVC	Film, pipe, containers
Polycarbonate	PC	Textiles, leisure boats
Polyurethane	PU	Insulation, floats
Alkyd	ALK	Paints, packaging
Polyester	PES	Textiles
Acrylonitrile butadiene styrene	ABS	Consumer products, auto parts
Polytetrafluorethylene	PTFE	Personal care products

Zettler et al. [57] collected and analysed plastic marine debris at several sites in North Atlantic. Plastics were collected in a rectangular net with 333 μm mesh size towed by the vessel SSV Corwith Cramer at the surface. The authors used PeakSeeker Pro Raman spectrometry system to identify the resin composition of the plastic fragment. Most of the collected fragments were positively identified as HDPE and PP, as shown in Figure 6.
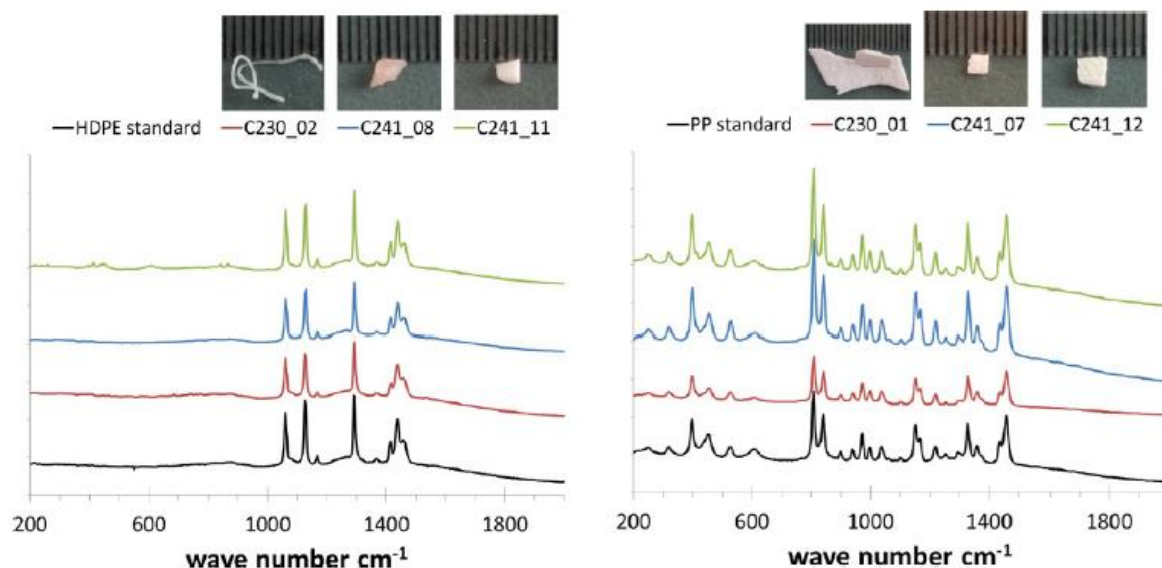
Figure 6. Raman spectra of collected plastics compared with spectra of known standard (black line) [57].

Lenz et al. [58] investigated the comparison of MP identification with visual microscopy and a Raman micro spectroscopy. Samples were collected using a closed filter unit with a mesh size of 10 μm, and a preceding mesh size of 300 μm from a trans-Atlantic cruise on the research vessel R/V Dana. The authors used a DXR Raman spectrometer coupled to a microscope and a 455 nm excitation laser to identify MP. First, a total of 452 fibres and 827 particles with size of <50 μm, 50-100 μm, and >100 μm were visually identified with the microscope. Then the Raman spectrometer was used. The results showed that 75% of the fibers and 68% of particles were confirmed using the Raman spectra. The success rate for visual plastic identification was 83% for particles with a size of >100 μm, 67% for the group between 50 and 100 μm, and 63% for particles with a size <50 μm. Pure visual identification resulted in a misidentification for both particles (32%) and fibers (25%). The size of a particle plays an important role in the identification, and the success rate in identifying particles with a size of <50 μm is higher using Raman spectroscopy.

Zhao et al. [60] investigated the accumulation of MP suspended in the subsurface water of three urban estuaries in southeastern China. Samples were collected using a Teflon pump at a water depth of 30 cm and poured through a 330 μm steel sieve. Randomly selected plastic particles were identified by micro-Raman spectroscopy and classified as PP, PE, PVC, and PTFE polymer types.

Naidu et al. [61] study the presence of MP in benthic fauna from the coastal waters of India. The sediment was collected and analysed using DXR Raman microscope with an excitation wavelength of 532 nm, and 98% of samples matched with polymer type PS.

Choy et al. [62] collected and investigated the distribution of MP in the deep-sea water column of Monterey Bay using a remotely operated vehicle (ROV). Reinshaw InVia confocal microscope along with a Raman spectrometer was used to analyse the collected particles. The authors reported the presence of the polymers PET and PA, indicating the existence of MP in the deep-sea water column.

Peng et al. [63] investigated MP contaminants in the deepest part of the world ocean. Water samples and sediment were collected from 2500-11000 m and 5500-11000 m depths, respectively. Identification was made using an optical microscope and a Raman spectrometer, revealing the MP abundance of several polymers, e.g., PVC, PA, ABS, PP, PE, PS, PET, PES, and PU. The authors suggested that "missing" MP in the ocean may have been transferred to the deep ocean.

Catarina et al. [64] provided a review for MP identification using micro-Raman spectroscopy, where they outlined the advantages and limitations of MP identification. Therefore, micro-Raman spectroscopy is a method of choice for detecting MP particles less than 20 μm. However, it stands out since it is a none-destructive method. To overcome limitations well-known for Raman spectroscopy, such as weak Raman signal and fluorescence interference, the authors suggested appropriate cleaning protocols, applying baseline algorithms, and using more efficient detectors.

MP pollution is found in water bodies all over the world. Research by many authors in collecting such data has shown that visual detection of MP alone is insufficient, but other analytical methods must be applied and discovered to detect certain sizes of MP, including Raman spectroscopy. Reports on the abundance of MP Raman spectroscopy is shown in Table 3.

Table 3. Reports of MP detection using Raman spectroscopy.

Authors	Sample type	Study Area
Zettler et al. [57]	Seawater	North Atlantic Ocean
Lenz et al. [58]	Seawater	Trans-Atlantic
Zhao et al. [60]	Seawater	Estuaries in SE China
Naidu et al. [61]	Sediment	Coastal waters of India
Choy et al. [62]	Seawater	Monterey Bay
Peng et al. [63]	Seawater	Mariana Trench
Saeed et al. [65]	Seawater	Kuwait
Pan et al. [66]	Seawater	NW Pacific Ocean
Wang et al. [67]	Seawater	Mid-west Pacific Ocean
Fytianos et al. [68]	Seawater	Mediterranean Sea
Li et al. [69]	Seawater	Yellow Sea China
Clunies-Ross et al. [70]	Sediment	Coastal location of New Zealand
Esiukova et al. [71]	Sediment	Baltic Sea
Mehdinia et al. [72]	Sediment	Caspian Sea
Chen et al. [73]	Sediment	Western Pacific Ocean

2.3.2. Marine corrosion

Corrosion is a problem in general for any structure made of metal, specifically for the marine environment and maritime activities. It directly affects pollution and causes damage to structures, equipment, machinery, port installations, and ships. Both marine and maritime infrastructure consist of stationary and mobile structures. Stationary structures are port terminals, cargo handling terminals, shipyards, offshore platforms, underwater pipelines, etc. At the same time, mobile structures consist of vessels of various sizes and categories, such as general cargo ships, passenger ships, tankers, and others. The basic material of all these structures is steel, which is directly affected by corrosion caused by the electrochemical process during the interaction of the metal surface with water [74]. The most frequently found corrosion products in the rust layers of steel are shown in Table 4.

Table 4. List of corrosion compositions found in rust layers.

Name	Type	Formula
Hematite	Oxide	$\alpha - Fe_2O_3$
Maghemite	Oxide	$\gamma - Fe_2O_3$
Magnetite	Oxide	Fe_3O_4
Ferrihydrite	Oxide	$Fe_5HO_8 4H_2O$
Goethite	Oxyhydroxide	$\alpha - FeOOH$
Akageneite	Oxyhydroxide	$\beta - FeOOH$
Lepidocrocite	Oxyhydroxide	$\gamma - FeOOH$
Feroxyhyte	Oxyhydroxide	$\delta - FeOOH$

According to International Maritime Organization (IMO), one of the conventional steels used for ship plates is A36. Sun et al. [75] developed new steel for the bottom plate of cargo oil tanks, whose composition differs from A36 by lower content of carbon (C) and phosphor (P), and higher content of alloying elements. The authors used micro-Raman spectroscopy to gain insight into the corrosion mechanism of the newly developed steel. JY-HR800 Confocal Microprobe Raman Spectrometer with an excitation wavelength of 532 nm was used. Raman spectra revealed that the corrosion product of the newly developed steel consists of goethite and magnetite, which is a stable and protective rust layer that further improves the corrosion resistance.

Li et al. [76] investigated the corrosion behaviour of low-alloy steel for cargo oil tanks under upper deck conditions, especially with the influence of elemental sulphur. For the test, the authors used E36, one of the conventional ship plate steel, and in-door apparatus was fabricated to simulate upper deck conditions. The test was conducted for 98 days. Raman spectroscopy was used to determine the corrosion composition. The final results revealed a spectral peak for elemental sulphur in the inner rust layer, which promotes cracking, fracturing, and peeling of the rust layer.

In contrast to corrosion caused by the interaction of metal surface and water, corrosion can be caused by the influence of the marine atmosphere. Rust phases caused by marine atmospheric corrosion can be further identified and classified by Raman spectroscopy to study corrosion morphology [77],[78].

Fuente et al. [79] investigated the use of different surface-sensitive techniques to characterize rust surfaces formed on mild steel exposed to marine atmospheres. The authors sought to elucidate where akageneite is located in the rust layer, a corrosion composition that forms when steel is exposed to marine atmospheres. Two cold-rolled mild steel panels were used and exposed to open air at two different sites with different marine salinity levels. One of the techniques used was micro-Raman spectroscopy. Raman spectra were obtained using a Raman module of the JEOL model JSM-6610LV spectrometer with an excitation source of 785 nm and an adjustable power between 0 and 30 mW. The results indicate two phases (lepidocrocite and goethite) are located at the outermost surface, while the phases akageneite and magnetite are formed in the innermost part of the rust layer, close to the steel base.

Husain et al. [80] studied corrosion resistance for AISI 316 stainless steel in marine environment application with a new developed coating as a combination of polyvinylidene

fluoride and polymethylmethacrylate. The authors used the inVia Reinshaw Raman microscope with an excitation source of 514.5 nm to differentiate stainless steel and the surface coated with polyvinylidene fluoride, polymethylmethacrylate and surface coated with a combination of both. Results showed characteristic peaks of both polymers in the combined coating. The authors reported outstanding performances of combined coating for anticorrosion efficiency, in contrast to using only one of the polymers as a coating.

Corrosion is the biggest problem in the maritime industry as it affects metal, which is a basic material for almost all mobile and stationary objects. To increase its durability and maintain its properties, new alloys and new coatings are being developed for corrosion protection. However, the newly developed alloys and coatings need to be investigated for their corrosion mechanisms. Therefore, Raman spectroscopy is one of the techniques that can provide information about the corrosion mechanism.

2.3.3. Antifouling

Biofouling, in maritime terms, is the accumulation of microorganisms and other organisms on the surfaces of all stationary and floating objects [81]. These organisms can cause damage to equipment, including pumps, valves, and piping systems. In addition, they can affect vessel performance by increasing hull resistance, leading to higher fuel consumption and engine stress to maintain desired speeds [82–84]. Therefore, chemicals with antifouling agents are used to prevent the growth of marine organisms on the hull of ships or stationary objects. This ultimately leads to unnecessary maintenance costs and damage to the stationary and floating objects [85],[86].

Yong et al. [87] developed a nanopaint for marine antifouling applications by incorporating zinc oxide (ZnO) into the alkyd resin matrix. Raman spectroscopy was used to study the bonding interaction, resulting in a final product with ZnO nanoparticles mixed into the alkyd resin matrix. The paint thus developed can be used in the maritime industry as a coating for ship hulls, etc.

Feng et al. [88] investigated the prevention of fouling using a pulsed high voltage electric field (HPEF) mediated by a carbon fiber coating. The Renishaw invia Reflex Raman spectrometer was used to analyze the coating with and without HPEF treatment. The Raman spectroscopy results show that the HPEF treatment did not change the molecular structure of the coating. In

addition, the HPEF method, which is mainly used in food sterilization and hospital wastewater treatment, could also be used as an antifouling method for materials in contact with seawater.

Long et al. [89] investigated the surface wettability of modified Si nanowires in terms of their antifouling performance. The authors used a plasma CVD system to coat Si nanowires with diamond and other carbon materials. Raman spectroscopy was used to characterize the microstructure of the modified nanowires. The Raman system was Horiba, LabRAM HR Evolution with an excitation laser of 324 nm. The Raman spectra confirmed diamond formation leading to a superhydrophobic surface consisting of diamond-coated Si nanowires required for antifouling application.

Researchers are developing new antifouling materials to meet International Maritime Organisation (IMO) [90] requirements for antifouling methods. However, the materials used to synthesise antifouling coatings must be characterised. Therefore, one of the techniques researchers use for characterisation is Raman spectroscopy [91–96].

2.3.4. Marine diesel engine

Marine diesel engine operates on a bunker fuel containing high amounts of sulphur [97]. Therefore, IMO address the regulation of particle emissions from the fuel combustion in the maritime industry [98]. Phenomena that occur during the combustion process is carbonaceous particulate matter (PM) known as soot. Since soot is dependent on combustion characteristics, its structure can be analysed using the Raman spectroscopy technique to provide information about carbon particles.

Lee et al. [98] studied diesel soot from 2-stroke and 4-stroke marine diesel engines under fixed load. The authors used two different bunker fuels (A and C), with the 2-stroke engine running on bunker A and the 4-stroke engine running on bunker C. The soot was collected directly from the exhaust pipe of the engines. Raman spectroscopy was used to characterise defects in the graphene structure based on the ratio of the intensity of the D/G band. The results showed the presence of graphitic and amorphous structures, i.e., D and G bands, with a higher degree of graphitization in the D/G ratio in 4-stroke engines than in 2-stroke engines, due to the different exhaust temperatures. Thus, the nanostructure of the soot is affected by the different engines and the different fuel burned.

Choi et al. [99] studied the characterization of PM from marine diesel engines for the Korean shipbuilding industry. The authors collected soot from a T/S Hanbada diesel engine burning Bunker B fuel at various cruise speeds. Raman spectroscopy was used to characterise the ratio of black carbon (BC) and organic carbon (OC) in the soot. The results showed that the ratio of BC to OC increased with increasing cruise speed and thus nitrogen oxides (NOx) emission.

Baek et al. [100] investigated the use of marine diesel engine combustion products as anode material for lithium-ion batteries. The authors collected soot from the exhaust pipe of a diesel engine, which was characterised by Raman spectrometry. The Raman spectrometer used was a Lambda Ray P-1 with an excitation laser of 532 nm. The results show that the soot transformed to a highly crystalline graphite, which facilitate electrochemical performance and is suitable as an anode material.

2.3.5. Photoluminescence

A spectrofluorometer with excitation wavelengths from 200 nm to 480 nm and emission wavelengths from 260 nm to 700 nm was used to detect oil-contaminated seawater. Seawater samples were collected during the summer at five locations in the Southern Baltic Sea. The authors defined the fluorometric index as a tool for oil contamination. Only one oil was used for the tests (Petrobaltic crude oil). The results show the proposed method's independent universality, so the study can be extended to a larger area and different seasons [101]. Using PL spectroscopy, authors [102] proposed a future study to detect the distinction of organic and non-organic objects in the water surface. Since lubricating oil is an essential component of marine engines, the PL method can be used to characterise oil properties because monocyclic and polycyclic compounds are present that cause the PL phenomenon in oil. In [103], changes in oil PL were represented by synchronous spectra, considering the operating time of the lubricating oil in a marine engine. Bukin et al. [104] developed laser-induced fluorescence spectrometer for remote-operated vehicles (ROV) to record the concentration of chlorophyll A and organic matter of natural origin.

3. THESIS SCIENTIFIC CONTRIBUTION AND RESEARCH PLAN

The development of a Raman spectrometer with a built-in zoom lens as a second collimating mirror is expected to enable a Raman spectrometer with variable spectral resolution. Further, such implementation will allow a Raman spectrometer to record photoluminescence spectra at the lowest zoom factor and record Raman spectra at the highest zoom factor in the same measurement. Therefore, based on the research object and taking into account the current research field state, and available literature, the working hypothesis is:

- Using zoom optics, a variable spectral resolution spectrometer system can be built to record Raman and photoluminescence spectra.

From the defined hypothesis, the main dissertation scientific contribution is:

- To approve proof-of-concept of variable spectral resolution by using zoom optics.

- To construct a Raman and PL spectrometer with a microscope using zoom optics to realize the concept of variable spectral resolution.

- To develop the electronics and corresponding software for signal acquisition, control, and regulation of the newly installed optics' optical gratings, focus, and zoom factors.

- To test this spectrometer in recording Raman and PL spectra.

Also, the additional contribution to the dissertation is:

- Fast acquisition of PL spectra using minimum zoom factor and acquisition of high-resolution Raman spectra using maximum zoom factor.

Finally, upon building the proposed system, the dissertation will derive:

- Realization of the Raman spectrometer for Raman and PL spectra for flexible material measurement.

- Patent for an invention.

4. EXPERIMENTAL SETUP

This chapter presents a macro-Raman spectrometer prototype that implements zoom optics to achieve variable spectral resolution suitable for recording both Raman and PL spectra. The prototype of the instrumental setup and its application in recording Raman and PL spectra are described and compared with the complementary information acquired on the same samples by the reference Raman spectrometer to validate the proposed concept. The proposed hardware for recording Raman and PL spectra, the samples used for measurements, and the results for validating the variable spectral resolution will be developed and described.

4.1. Instrumental setup for the proof-of-concept spectrometer

The schematic diagram of the proposed Raman spectrometer with the variable spectral resolution is shown in Figure 7, together with the implemented additional zoom lens component.
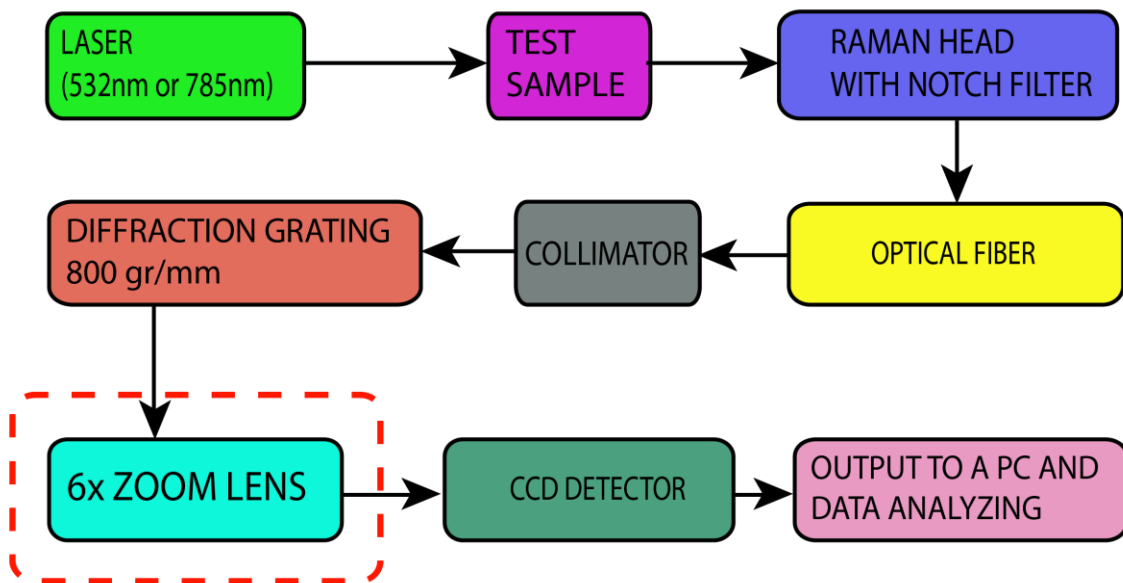
Figure 7. Schematic diagram of the proposed Raman spectrometer with additional zoom lens component (red dashed rectangle) based on a typical Raman spectrometer configuration.

In the proposed Raman spectrometer, contrary to the classical approach, the second collimating mirror that focuses Raman or PL signal to the CCD detector is replaced with a zoom lens, as shown in Figure 7. For such a case, a 6x zoom Fujinon-TV lens, f/=1.6, with a variable focus f_{co2} = 14 - 84 mm, was used. The working prototype assembly is shown in Figure 7a. It is based on commercial, off-the-shelf, and custom-built components. The excitation laser of wavelength λ = 532 nm was a DPSS laser (CNI, model DHL-W 532-500 mW). It was coupled

to the off-the-shelf Raman head with a notch filter (that blocks Rayleigh scattered light) via a multimode optical fiber (ThorLabs, 0.10 NA, d = 25μm). An additional 785-nm laser (CNI, model FC-D-785-450 mW) was used for the wavelength calibration. The same multimode fiber was used for the incoming laser excitation of the sample as well as the signal carrier to the custom-built spectrometer. The optical fiber's exit is placed in the focus of the first collimating lens (f_{co1} = 150 mm), which provides a parallel light beam of the signal after the exit from the optical fiber. This parallel light beam falls onto the diffraction grating with 800 grooves/mm. After grating, the dispersed beam is focused onto the CCD using a zoom lens. In this case, the CCD detector was Starlight Xpress, model MX716, with 752 x 580 pixels of size 8.6 x 8.3 μm.

To validate the principles of the proposed spectrometer, a high-grade laboratory Raman spectrometer (Horiba, JY-T64000) shown in Figure 8b, was used to compare the recorded spectra [105]. The referenced spectrometer can operate as a single, a double or a triple monochromator spectrometer with the double foremonochromator usable as a subtractive filter, or as a double additive dispersive system for the ultimate resolution. The single monochromator mode has a spectral resolution of 0.55 cm^{-1}. It contains liquid-nitrogen-cooled CCD (Cryogenic, Symphony I, 1024 x 256 pixels, pixel size 26 x 26 μm), 640 mm focal length monochromator, and a working spectral range from 400 to 1100 nm. To ensure the same conditions for the incoming Raman scattered photons in both spectrometers, the optical cable was detached from its entrance and attached to the entrance slit of the third (single) monochromator of the JY T64000 spectrometer. This way, the same Raman and PL signal was analysed on both spectrometers.
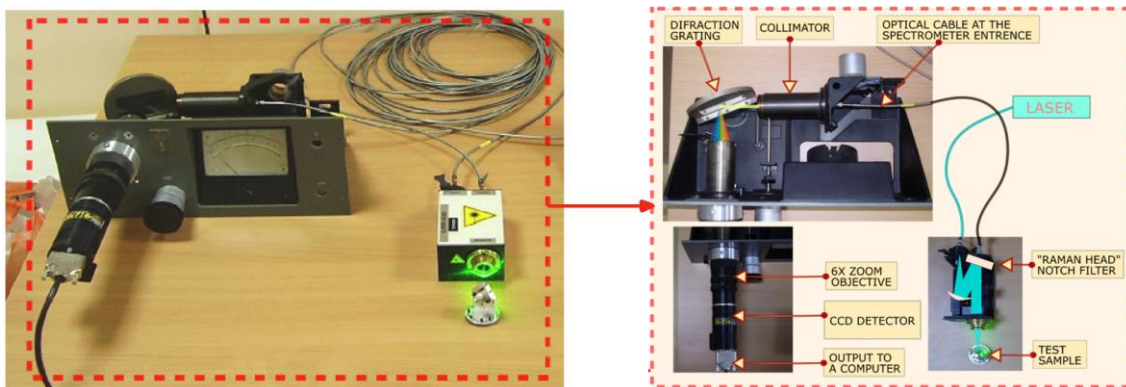
(**a**)

(**b**)

Figure 8. Assembled prototype of the proposed Raman spectrometer (a) and the reference Raman spectrometer JY-T64000 (b).

4.2. The samples analysed for comparison

Three samples with a different spectral resolution requirements were used to test the prototype's spectral zooming performance for Raman and PL spectra recording. The first two samples are powder forms of nanosized zirconium oxide (ZrO_2) and titanium oxide (TiO_2) crystals. These samples were used due to their strong and known Raman bands located at 335, 475, and 620 cm^{-1} for ZrO_2 and 400, 520, and 640 cm^{-1} for TiO_2. The third sample is a powder of alumina oxide (Al_2O_3) which has a strong and broad PL signal in the spectral interval of 500 - 850 nm. The samples were placed in separate transparent cuvettes for the measurements for both Raman spectrometers.

4.3. Raman and PL spectra

To record Raman and PL spectra, the wavelength interval on the CCD region of the proposed Raman spectrometer was first calibrated with two lasers (λ_{green} = 532 nm, λ_{red} = 785 nm), converting each pixel to the corresponding wavelength value. A 300 – 1200 nm spectral range was obtained at the minimum ZF, and 490 - 616 nm at the maximum ZF. After the calibration, a 532-nm excitation laser was used for all samples. The output power was calibrated so that the laser power incident on the sample surface was 5 mW, measured with a portable laser power meter (Coherent Laser Check Power Meter). The exposure time was set at 1 s for Raman measurements and 3 s for PL measurements for both the prototype and the reference spectrometer. The wavelength interval for the Raman measurement was set according to the ZrO_2 and TiO_2 spectral lines which are located between 150 and 750 cm^{-1}.

For all Raman spectra acquired with the proposed systems, the data were processed using Matlab R2018. It needs to be pointed out that by taking the moving average value of all pixels in the vertical axis and then normalizing it in the acquired spectra, the noise (such as hot pixels and salt-and-pepper noise) was eliminated. This resulted in one-dimensional spectra plotted as intensity vs. wave number. Equations (1) and (2) show the proposed system's spectral resolution analytically. Equation (1) is a Gaussian function used to fit Raman bands, and equation (2) is a Full Width at Half Maximum (FWHM) [106][107] for the Stokes lines of the ZrO_2 Raman spectra at a minimum and maximum ZF. The variable parameters were peak intensity A, the peak center position b, and the standard deviation of the Gaussian fit σ.

$$y = A e^{\frac{(x-b)^2}{2\sigma^2}} \tag{1}$$

$$FWHM = 2\sigma\sqrt{2ln2} \tag{2}$$

Assuming a Gaussian line profile [108], a reasonable approximation for the FWHM of the Raman lines is provided by the relationship:

$$FWHM = \sqrt{(\Delta\tilde{v}_{slit})^2 + (\Delta\tilde{v}_{resolution})^2 + (\Delta\tilde{v}_{line})^2} \tag{3}$$

where:

$\Delta\tilde{v}_{slit}$ is a true Raman bandwidth

$\Delta\tilde{v}_{resolution}$ is a spectral resolution of the spectrometer

$\Delta\tilde{v}_{line}$ is a Raman line width

Raman spectra of ZrO_2 were recorded with a different ZF. Then TiO_2 and ZrO_2 spectra were recorded with the maximum ZF and compared with Raman spectra of the same samples obtained with the JY-T64000 spectrometer. Next, the spectral resolution for the different ZF is analytically calculated. Finally, a comparison of the PL spectrum of Al_2O_3 obtained with the proposed system and JY-T64000 was performed.

4.3.1. Raman spectra

The Raman spectra of ZrO_2 acquired with the prototype Raman system with the different ZF are shown in Figure 9.
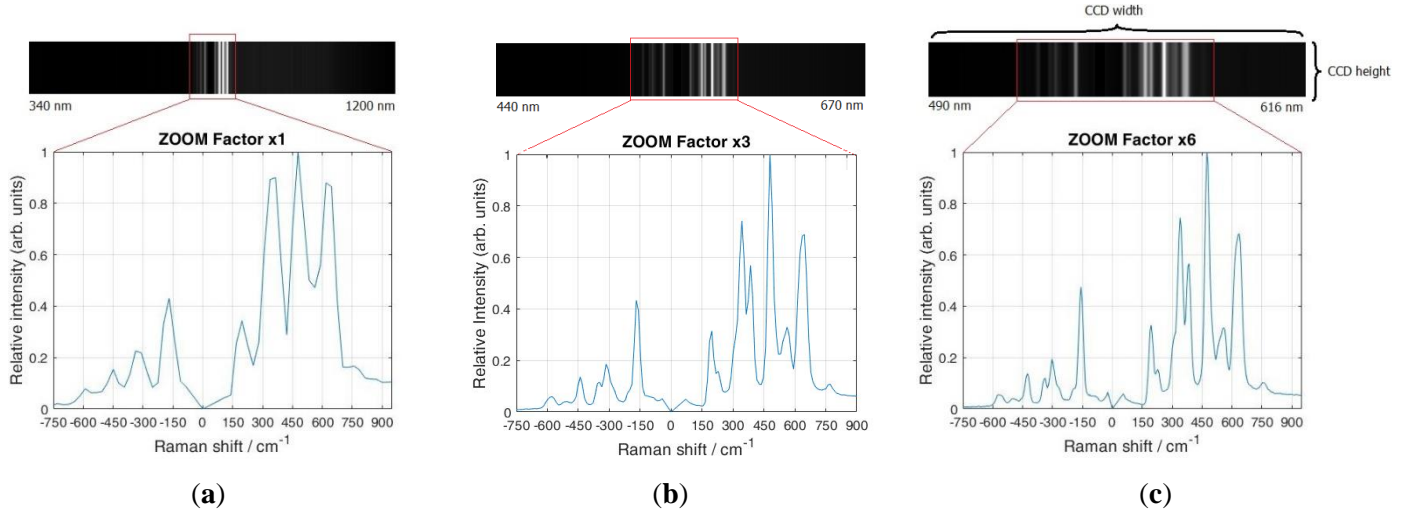

Figure 9. Acquired image over the CCD region and a Raman spectrum of a ZrO_2 with different zoom factors, (a) for a ZF of 1x, (b) for 3x, and (c) maximum ZF of 6x.

It has to be pointed out that on the top of each figure are recorded spectral lines of the used sample, and on the bottom, a spectre is shown as intensity vs. Raman shift. Figure 9a shows the spectral interval over the entire CCD region at a ZF of 1x range, i.e., from 340 to 1200 nm. In this case, only a few spectral lines appear around the expected location of the 532 nm laser. The spectrum shows a few anti-Stokes lines and three strong Stokes lines at 337.2, 477.6, and 618.1 cm^{-1}. However, more spectral lines appear when the ZF is increased three times, as shown in Figure 9b. The spectral interval over the CCD range is now from 440 to 670 nm. Also, it can be observed that the obtained spectra have additional "hidden" Stokes lines at wavelengths 337.2 and 477.6 cm^{-1}, which is due to the increased spectral resolution. Furthermore, if the ZF is set to the maximum value of 6x (Figure 9c), it can be observed that the spectral interval becomes even narrower across the CCD region, measuring only 490 - 616 nm. It has to be pointed out that the spectral lines at this ZF are more distinct than at a lower ZF. Therefore, by increasing the ZF, the total spectral interval over the CCD region reduces, and the spectral resolution increases, which is evident from the appearance of new lines accompanied by the line narrowing. Figure 10 shows the Raman spectra of TiO_2 and ZrO_2 compared to the reference Raman spectra.
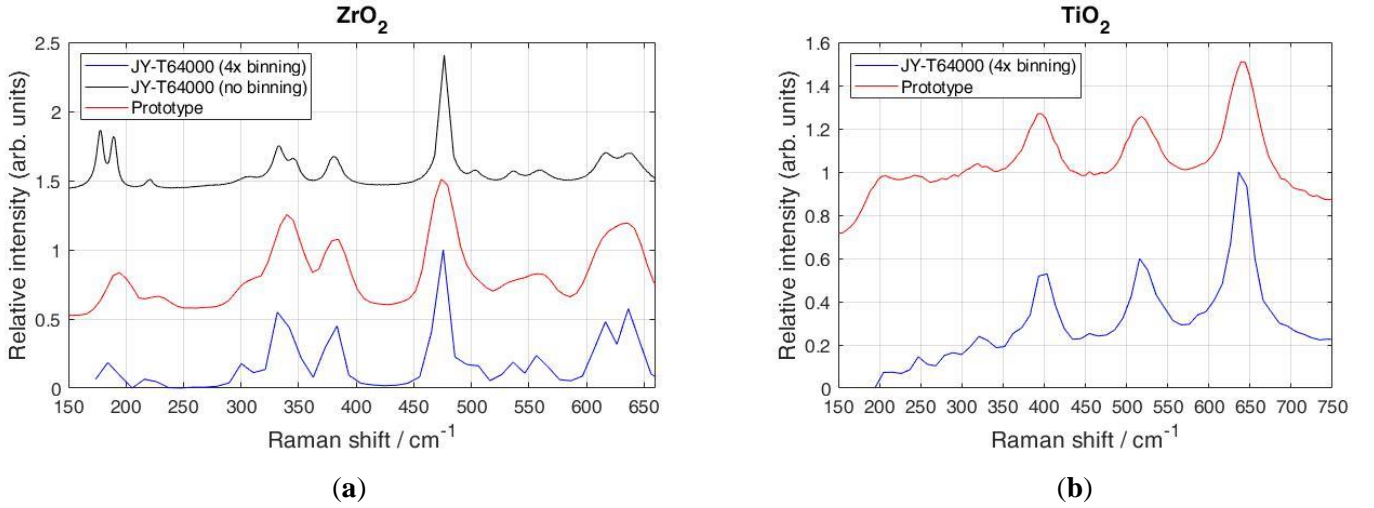
(**a**)   (**b**)

Figure 10. Comparison of Raman spectra of ZrO_2 (a) and TiO_2 (b) using the proposed spectrometer with a reference spectrometer. Note that the reference spectrometer spectra without binning at (a) does not match the spectral dispersion of the prototype spectrometer, therefore, binning is used.

The ZF was set to the maximum 6x magnification, and the spectral interval from 537 to 557 nm was recorded. From Figure 10, it can be seen that both spectrometers (prototype and the reference model, respectively) show similar spectra. However, in Figure 10a on the left-hand side, the laboratory-grade spectrometer shows additional spectral lines at 552 and 630 cm^{-1} for ZrO_2. This is due to the reference spectrometer's high spectral resolution achieved with the long focal length and a minimum noise generated by deep-cooled CCD. However, for the TiO_2 sample seen in Figure 10b, the Raman spectra match the reference spectrometer's spectra, which is due to the strong Stokes peaks of a sample where additional "hidden" peaks don't exist.

4.3.2. Spectral resolution

The following equation gives the relationship between the reciprocal linear dispersion D^{-1}, the focal length f, the order of the diffraction n, and the distance between blazes d.

$$D^{-1} = \frac{d}{n*f} \tag{4}$$

By substituting the value of the focal length f = 84 mm for the maximal ZF, the order of the diffraction n = 1, and the distance between rulings d = 1/800 mm, in the above equation reciprocal linear dispersion (D^{-1}) for the first-order diffraction is 14.9 nm/mm. Per-pixel width of this value is 0.128 nm/pixel. The spectral dispersion at the position of the Raman band at 483.5 cm^{-1}, as shown in Figure 9c (546 nm wavelength of light) is 4.28 cm^{-1}/pixel. The same calculation for the reference Raman spectrometer gives a dispersion of 0.756 cm^{-1}/pixel, which is a 5.67 times smaller value. For this reason, to arrive at a similar spectral resolution, a binning factor of 4 times was used during recording by the reference Raman spectrometer.

The input slit also limits the spectral resolution. The input slit width of the proposed spectrometer was fixed and determined by the diameter of the coupled optical fiber core, which is 25 μm. At the maximal ZF of 6x, the total magnification of the monochromator is fco2/fco1 = 0.56, which gives us the image size of 14 μm of the fiber core at the CCD. From Nyquist limit, which determines whether the CCD sensor can differentiate between two neighbouring spectral lines, i.e., 3 times image core size on CCD. Therefore, for the proposed system, the CCD sensor can differentiate between two adjacent spectral lines when their peaks are separated at 42 μm on the CCD, which is 21 cm^{-1} for the wavelength of 546 nm. The difference of the peak positions observed as two humps (overlapping lines) in the spectral interval 520 - 580 cm^{-1} (shown in Figure 10a) is 28.2 cm^{-1}, which is very close to the expected value.

Further, the obtained resolution by measuring the linewidth for the band at 483.5 cm^{-1} has been tested. By fitting Gaussian profile (Figure 11a,b), the linewidth at the minimal and maximal ZF was measured, resulting in 81.88 cm^{-1} at the minimum ZF and 27.62 cm^{-1} at the maximum ZF. By increasing the ZF, new Raman bands also appear due to the increased resolution.

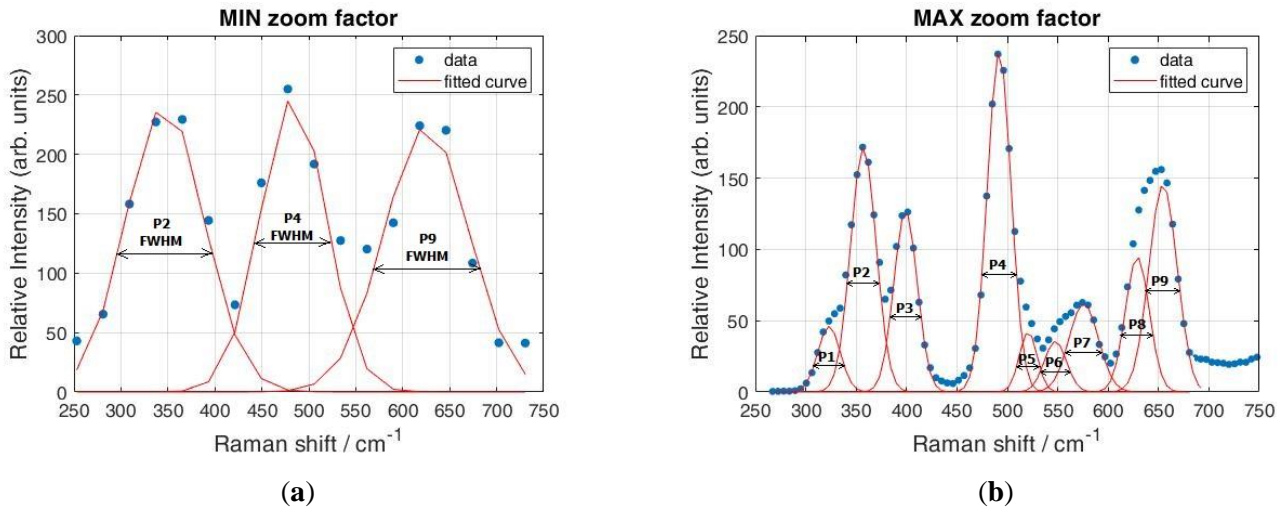(**a**)   (**b**)

Figure 11. Calculation of the spectral resolution of the proposed system with a ZrO2 Raman spectrum using Gaussian fit for each Stokes peak at different ZF. Note the difference in the number of peaks and fitted curves at the same spectral interval for minimum ZF (a) and maximum ZF (b).

Table 5 shows the Gaussian fitting parameters for each peak at the minimum and maximum ZF, together with their FWHM. It is evident from the linewidth of the characteristic peak at 483.5 cm^{-1} that spectral resolution increased significantly, i.e., the spectral resolution calculated by the FWHM became more than three times larger.

Table 5. Gaussian fit parameters for obtained spectral peaks

Curves	MIN ZF				MAX ZF			
	A	b[cm^{-1}]	σ	FWHM	A	b[cm^{-1}]	σ	FWHM
P1	n/a	n/a			46.1	306.34	11.14	26.22
P2	241.9	346.8	41.43	97.57	171.77	340.44	12.63	29.74
P3	n/a	n/a			127.1	382.25	11.59	27.29
P4	248.5	483.5	34.77	81.88	239.1	475.14	11.73	27.62
P5	n/a	n/a			42.2	503.97	8.76	20.63
P6	n/a	n/a			35.6	530.81	11.02	25.97
P7	n/a	n/a			61.97	558.99	14.61	34.41
P8	n/a	n/a			94.9	612.17	11.82	27.84
P9	223.8	625.45	45.13	106.28	145.5	637.68	13.97	32.9

Using the values $\Delta\tilde{v}_{slit}$ = 7.0 cm^{-1}, $\Delta\tilde{v}_{line}$ = 19.0 cm^{-1} (measured by the reference spectrometer without binning factor) and fitted FWHM = 27.62 cm^{-1}, by Eq. (3), it was found $\Delta\tilde{v}_{resolution}$ = 18.78 cm^{-1}, which also agrees with previously estimated values. By using Eq. (3) and FWHM = 81.88 cm^{-1} at the lowest ZF of the 483.5 cm^{-1} line, the estimated resolution in this case is $\Delta\tilde{v}_{resolution}$ = 78.42 cm^{-1}, which is 4.17 times worse resolution in comparison with the ZF = 6. In general, most spectrometers are not routinely used at the limit of their resolution because of a number of different aberrations, so the slight differences from the expected value of 6 times change in resolution when applying ZF = 6 can occur.

4.3.3. Photoluminescence spectra

The PL spectra of an Al$_2$O$_3$ shown in Figure 12 were recorded using the proposed system and the reference Raman spectrometer with an excitation laser of 532 nm. The exposure time was set to 3 s for both spectrometers. ZF was set to 3x magnification so that the spectral interval from 450 to 850 nm is recorded in a single shot. However, the total acquisition time for the same spectral interval using the reference spectrometer was around 30 times longer because the signal of each spectral interval of 20 nm width (on average) is collected separately (scanning multichannel mode) and then combined into one single spectrum.
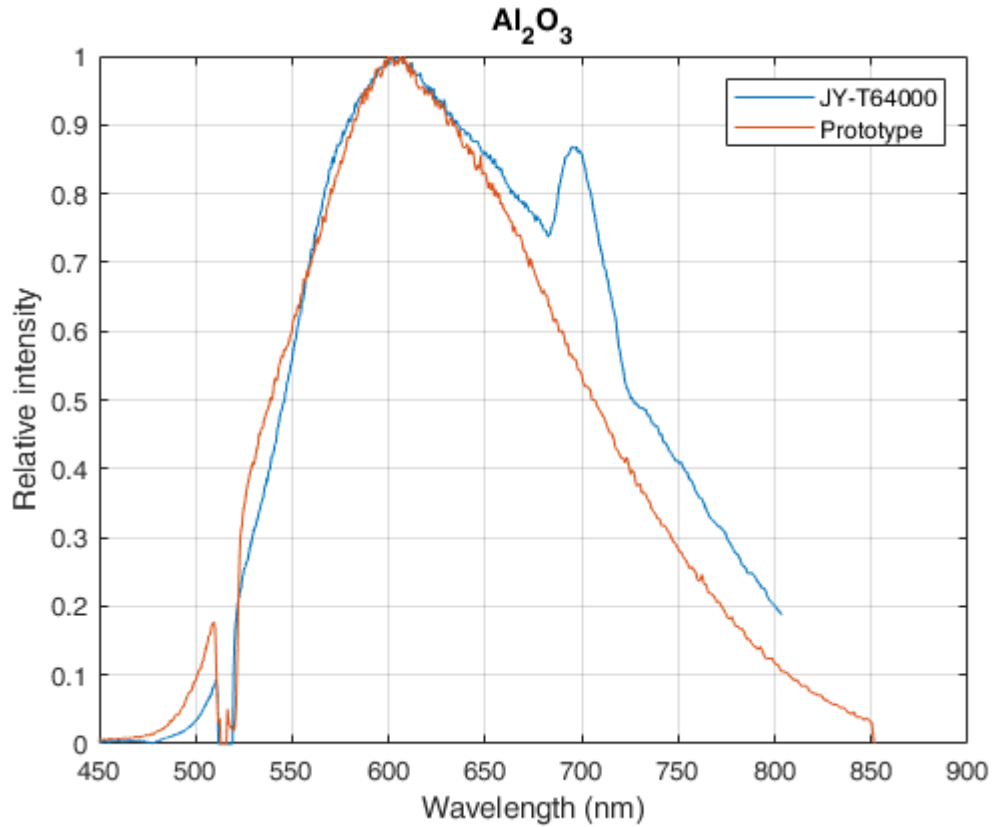
Figure 12. PL spectra of Al_2O_3 recorded with the proposed Raman spectrometer and the reference commercial JY-T64000 spectrometer.

Such a procedure could result in artifacts like the one observed in the PL spectrum at wavelength of 690 nm shown in Figure 12, which does not exist in the PL spectrum of the Al_2O_3. The spectral stitching could explain this artefact in the reference spectrum, where the recording consists of 20-30 small sections of spectral intervals. These small sections of spectra are then merged, resulting in a spectrum with a large spectral interval. However, these small sections tend to carry variable slope backgrounds, which change during the recording. So, the combining procedure has the wrong information on the PL intensity data at the beginning and the end of each section, leading to the artificial peaks that occur after combining into a single spectrum.

5. METHODOLOGY AND MATERIALS FOR THE SPECTROMETER CONSTRUCTION

This chapter describes the methodology used for designing a micro-Raman spectrometer prototype with a variable spectral resolution. Furthermore, the design of additional electronic hardware, firmware, and software to control the spectrometer and acquire spectra is described. Finally, the system was assembled, and the instrumental setup for acquiring Raman and PL spectra and the samples used for the measurements are described.

5.1. Raman spectrometer components

Every Raman spectrometer consists of three major parts, an excitation source, a sampling unit, and a detection unit. Component selection methodology was used to find all suitable components for the designed prototype to perform its operation. Furthermore, additional components were added for the prototype and are listed in Table 6.

Table 6. List of all necessary mechanical and optical components used for the prototype.

Major parts	Description	Model
Excitation source	A laser with λ_{green} = 532 nm was used for all sample measurement	DPSS, DHL-W532-500 mW
Sampling unit	A standard optical microscope was used as a sampling unit with an 80x and 0.7 NA microscope objective.	AMDSP XUM102
Monochromator unit	The spectrometer was used for the analysis of scattered light.	LSMO-6 Spectrum Acquisition System

Additional parts	Description	Model
Bandpass filter (BPF)	Filter used for elimination of surrounding light at the entrance of microscope.	Thor Labs FL532-1
Notch filter (NF)	Two filters were used to eliminate Rayleigh scattering at the exit of the microscope.	Thor Labs NF533-17
Focusing Lens	Lens used to focus incoming scattered light from microscope to spectrometer.	Solagon 1:1,2/25mm
Zoom lens	The zoom lens acts as a second collimating mirror, thus enabling variable spectral resolution.	Fujinon Lynx II A18x8.5, 1:1.7/8.5-153mm
CCD detector	They are sued for the acquisition of dispersed light from the spectrometer.	Trius Pro 694 Monochromatic
Shutter	Used for the acquisition of dark frame.	Thor Labs SHB05

5.1.1. Excitation source

The excitation source for the spectrometer is a monochromatic light, i.e., a laser. Therefore, the type of laser directly influences the quality of a spectrum and the spatial resolution. Since there are numerous types of laser wavelengths, it is necessary to select one or more that can detect the Raman and PL signal compared to the background noise, i.e., an appropriate S/N ratio. For the prototype, a laser with a wavelength of $\lambda_{green} = 532$ nm was selected because this laser energy is sufficient for both electronic and vibrational transitions of the used samples. However, another laser with a wavelength of $\lambda_{red} = 785$ nm is used to calibrate the wavelength interval on the detector area.

5.1.2. Microscope

For microscopic analysis of the samples, the prototype was extended to include the AMDSP XUM102 standard optical microscope shown in Figure 13, which enables micro-Raman spectroscopy. The excitation source was placed at the entrance of the microscope. Inside the microscope two beam splitters were replaced. The first, which split the incident laser beam onto the sample, was a 30:70 beam splitter in which 30% of the incident light energy was transmitted to the sample through the microscope objective. The laser beam was focused on the sample with an 80x - 0.7NA microscope objective used for all sample measurements. The light scattered from the samples was collected back through the microscope objective and transmitted through a second beam splitter at the exit of the microscope. Since collecting as many Raman scattered photons as possible is necessary, the second beam splitter was a 70:30, where 70% of the scattered collected light is transmitted to the spectrometer.

Figure 13. Standard optical microscope AMDSP XUM102 used for microscopic analysis of the samples with the prototype.

5.1.3. Filters

Some unwanted scatter mask the Raman signal during the Raman measurement, so it is necessary to eliminate them all. The first one is the effect of natural light in the environment of the prototype and the second one is Rayleigh scattering. Further, to suppress the ambient light, a bandpass filter of wavelength $\lambda_{BPF} = 532$ nm is used at the entrance of the microscope so that only the light from the laser can enter the microscope. Also, at the microscope's output, two notch filters or bandstop filters of wavelength $\lambda_{NF} = 532$ nm are used to eliminate Rayleigh scattering, which masks Raman scattering.

5.1.4. Focusing lens

Due to the natural divergence of the light and microscope's distance between the light and spectrometer, it is necessary to focus scattered light at the spectrometer's slit input. For this purpose, a focusing lens shown in Figure 14 with aperture of f/1.2 and focal length of f = 25 mm was used.
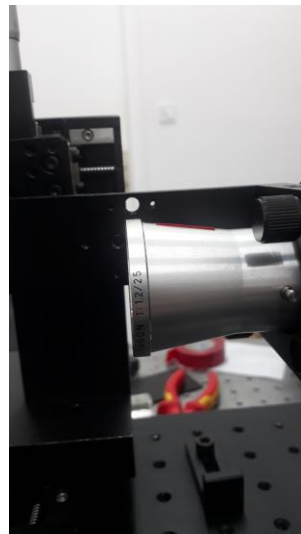
Figure 14. Off the shelf lens used for focusing the transmitted beam from the microscope to an entrance slit of the spectrometer unit.

5.1.5. Monochromator

The LSMO-6 monochromator was used for the proposed spectrometer. Its purpose is to analyse the scattered light and make the spectrum. A dispersive monochromator consists of a dispersive element i.e. a grating, and two concave collimating mirrors. The grating is mounted on a rotary platform connected to a screw mechanism by the arm. This mechanism converts a

linear motion into a rotary motion. The grating can be rotated with the described mechanism, and it scans different wavelength ranges. At the entrance to the spectrometer is a slit with adjustable width, which determines the spectral resolution. The wider the slit, the more scattered light can enter, and the resolution is lower; the narrower the slit, the less scattered light can enter, and the resolution is higher. For the prototype, the width of the entrance slit was set to 35µm, as shown in Figure 15a.

(**a**)   (**b**)
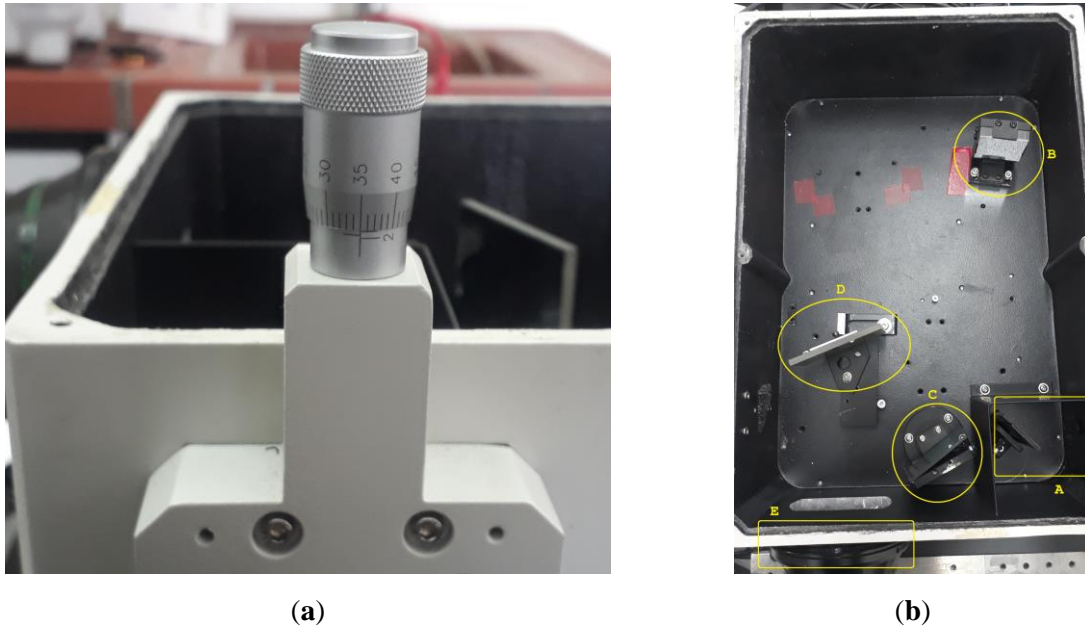
Figure 15. Variable entrance slit width with micrometer set at 35 µm (a), and the inside of spectrometer unit where the incoming beam is reflected (b).

Furthermore, the following changes were made inside the spectrometer (see Figure 15b marked yellow) as follows. First, the grating C was replaced with a 1100 g/mm grating (the more grooves, the better the spectral resolution), then the flat mirror D is added so that the dispersed light is directed from the grating to the output of the spectrometer. The second concave collimating mirror was replaced with a Fujinon zoom lens E and it is attached to the output of the spectrometer. Entrance mirror A and the first collimating mirror B were adjusted so the scattered light is parallel through spectrometer.

5.1.6. Zoom lens

To obtain a spectrometer with variable spectral resolution suitable for recording both Raman and PL spectra, the second collimating lens of the spectrometer was replaced by a zoom lens. The Fujinon Lynx II A18x8.5, 1:1.7/8.5-153 mm (Figure 16) was used for the prototype. This zoom lens was chosen because of its high zoom factor of 18x and the barlow lens of 2x,

allowing a longer focal length between the CCD detector and dispersed light from the spectrometer. Therefore, different spectral intervals can be recorded by changing the zoom factor, i.e., the lower the zoom factor, the wider the spectral interval, and conversely, the higher the zoom factor, the narrower the spectral interval. This feature within a prototype allows recording of Raman spectra with a higher zoom factor and fast recording of a PL spectrum with a lower zoom factor.

Figure 16. The Fujinon Lynx II A18x8.5, 1:1.7/8.5-153 mm attached to a spectrometer exit.

5.1.7. Detector (CCD detector)

It is necessary to use an optical sensor capable of collecting light to detect the scattered light from the spectrometer. Such detectors can be charge-coupled devices (CCD), metal oxide semiconductors with active pixel sensors (CMOS) or either electron-multiplying CCDs (EMCCD). For the proposed spectrometer, a Trius Pro 694 monochromatic CCD was used (Figure 17) mounted on the Fujinon zoom lens. This detector was chosen for its ability to capture high-resolution images with 2750 x 2200 pixels. The size of the pixels is 4.54 μm, giving an active area of 12.5 x 10 mm and a maximum binning of 16 x 16. A 16-bit analogue-to-digital converter to processes the information for each pixel and internal Peltier cooling for noise reduction.

Figure 17. Optical sensor Trius PRO-694 monochromatic CCD mounted on the Fujinon zoom lens.

5.1.8. Shutter

During Raman or PL measurement, the information about the scattered light projected onto the CCD surface must be as noise-free as possible. This noise is a combination of unwanted ambient light and thermal and electrical effects within the camera. Thermal and electrical effects cause the camera to detect small signals. However, these signals do not contain light information but provide information about the state of the sensor, i.e., the detection of hot pixels and the noise pattern generated in the camera. The image taken when the incident laser beam is blocked is defined as a dark frame. To achieve blocking of the incident laser beam, a shutter was placed in front of the laser as shown in Figure 18. The shutter used for the prototype is the Thor Labs SHB05, which has an implemented controller so that it can be easily controlled by software.
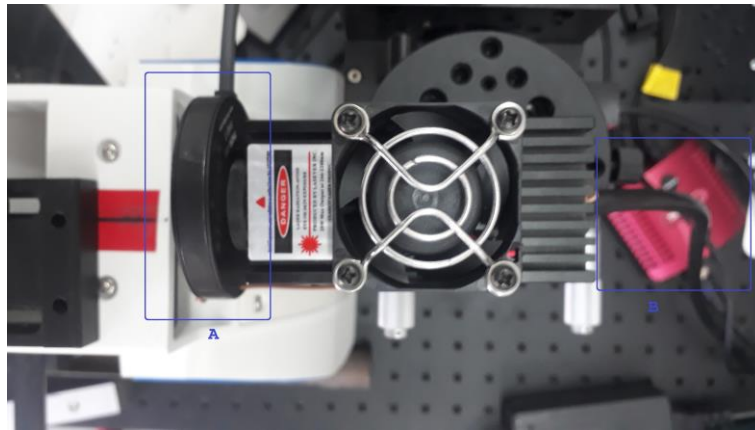
Figure 18. Thor Labs SHB05 shutter mounted in front of laser (blue rectangle A) and its controller (blue rectangle B).

5.2. Electronic hardware printed circuit board (PCB)

Additional electronic hardware is implemented to achieve automation to all the mechanical and optical components required to operate the prototype. Since no standard electronic hardware can be used for the prototype, and all commercial spectrometer manufacturers have their electronic hardware and software, it was necessary to design and build accompanying electronic hardware. This will allow automation to control the angle of the grating, zoom factor of the zoom lens, image acquisition, and shutter control. The selection of the electronic components and the development of the printed circuit board is described. The electronic hardware was developed after selecting all the necessary mechanical and optical components. The information gathered through the selected components for the proposed spectrometer

provided clear insight into what and how it could be controlled electronically. It was found that the electronic board must have the following:

- Communication between the PCB board and the computer with USB.

- Driver for the stepper motor that controls the grating angle of the spectrometer.

- Pulse width modulation (PWM) driver to control the servo motors on the zoom lens.

- I/O pins for controlling the shutter.

- Additional general-purpose I/O pins.

- Sufficient power supply for stepper and servo motors.

5.2.1. Electronic components

The design process for an electronic schematic begins with creating a block diagram that lists all the components needed for the prototype PCB. Figure 19 shows a block diagram with the essential parts of the board and additional parts that are not needed for basic functionality but are added as optional features that can be used for further improvements and upgrades.
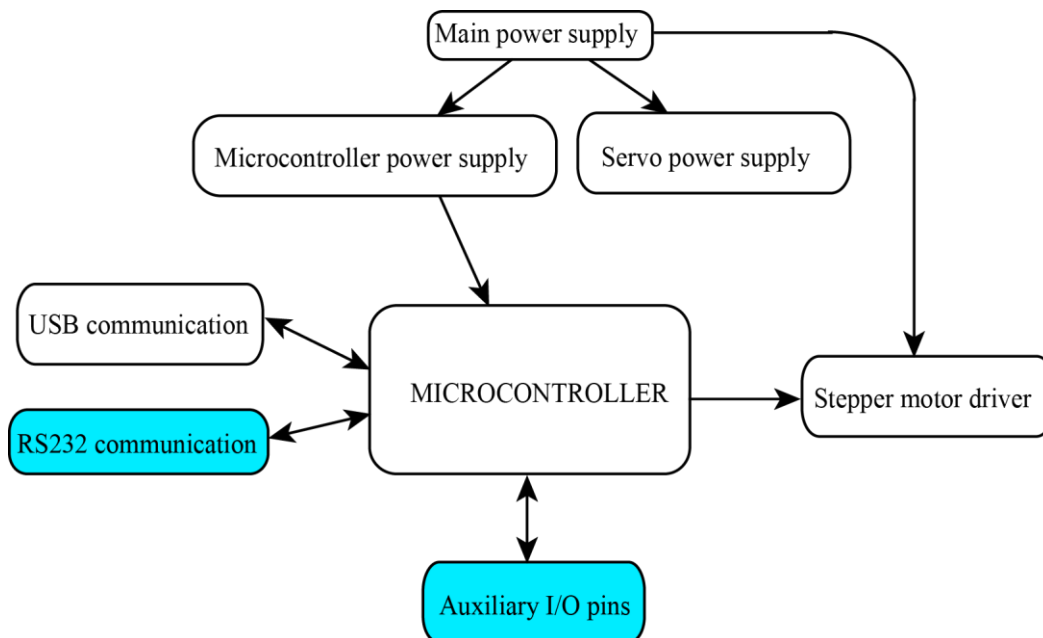

Figure 19. Block diagram of proposed embedded PCB for prototype.

Based on the block design, all active and passive electronic components had to be selected to meet the requirements of the prototype. Therefore, the microcontroller (MCU) is selected first. From the large number of MCUs, the selection was reduced by looking for an MCU that implement at least one USB interface, one PWM interface, enough I/O pins and memory for the complexity of the firmware. Therefore, the PIC18F45K50 MCU was selected. Information about the MCU capabilities and its peripherals are listed in Appendix A. However, there are two types of these MCUs, one that operates at 3.3V and another that operates at 5V. For the prototype, an MCU with 5V as the operating voltage was chosen because the stepper motor driver requires logic inputs of 5V. The stepper motor that controls the angle of the grating under the spectrometer is a DC 5-wire unipolar stepper motor, i.e., two coils of the motor are connected to the center tap wire, and four contacts $(A, \bar{A}, B, \bar{B})$ are used to drive the motor. However, since the rotation of the grating depends on the motor steps, it is necessary to control the stepper motor with a high resolution, resulting in a small spectral motion. Therefore, a driver with implemented micro stepping function must be chosen. For the prototype, a unipolar stepper motor driver SLA7078WPRT was selected. This driver has a function to control a unipolar stepper motor and has a controllable micro stepping function that allows it to operate in full, half, quarter, eighth, and sixteenth steps. The detailed features of this driver are listed in Appendix B. The zoom factor of the zoom lens is controlled by a standard servo motor DC (HDKJ D3015), which has a maximum rotation angle of 300° and is connected to a zoom lens. Therefore, the board's power supply must consist of three separate voltage levels, i.e., 5V for the MCU, 6V for the servo motor, and 12V for the stepper motor. To achieve this, the Mean Well LRS-35-12 AC-DC Single Output Power Supply was used for the main power supply. It provides up to 3A of current at 12V, which is sufficient for the board and motors. From the main power supply, 12V is distributed through an integrated circuit (IC), a fixed 5V regulator MC7805 that supplies 5V to the MCU, and an adjustable positive low dropout regulator IC LM1084 that can be configured as a 6V regulator with a maximum output current of 5A for the servo motors. The driver for the stepper motor and the stepper motor is directly connected to a 12V main power supply. Finally, according to the circuit diagrams described in the next chapter, all other passive and active components are selected according to the circuit diagrams.

5.2.2. Electronic circuit diagram

After all of the components for the electronic circuit board have been selected, an electronic circuit diagram is designed. A circuit design starts in the schematic capture phase using KiCad electronic Computer-Aided Design (eCAD). The first schematic for power distribution is created.
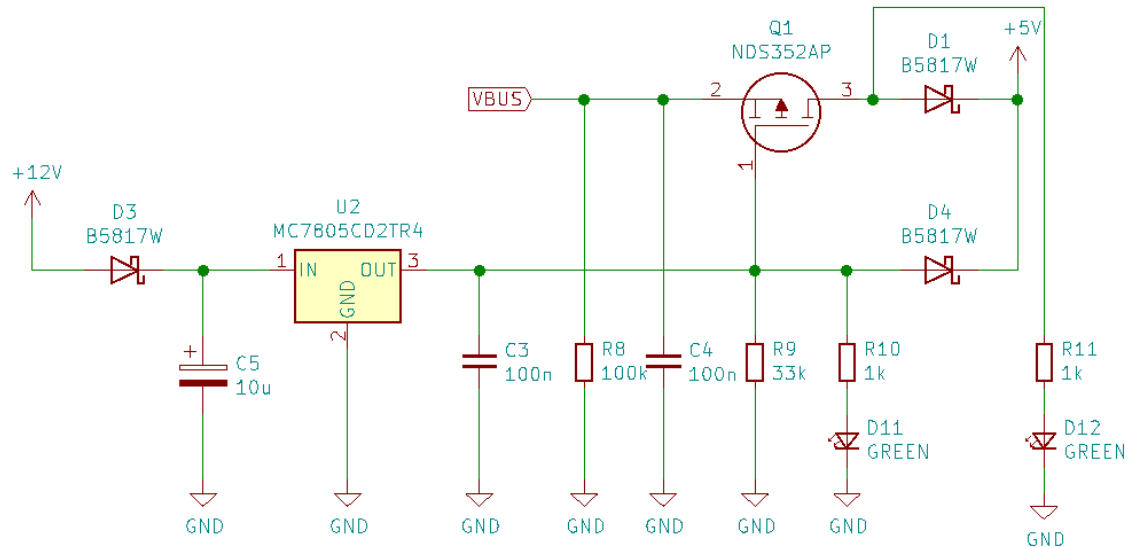
Figure 20. Electronic schematic diagram for the MCU power supply.

Figure 20. shows a schematic diagram of the MCU power supply. It consists of a fixed 5V regulator U2 IC MC7805 connected to a 12V main power supply. To prevent wrong connection at the input, i.e. wrong polarity, Schottky diode D3 is used. Capacitors C5 and C3 are selected according to the regulator datasheet to filter the input voltage and ensure output voltage stability. The output of the regulator is connected to a resistor R10 in series with LED D11 to indicate that the power supply is connected and ready for operation. However, this option limits the testing and debugging of the MCU. Therefore, additional components are added to allow the MCU to operate using only the USB power supply. This way, the MCU can be easily debugged and tested. For this function, a P-channel metal oxide field effect transistor (MOSFET) Q1 is used in a switch configuration. When the voltage at the gate of Q1 is high, Q1 acts as an open switch, therefore, the LED D12 indicator is off, and the 12V main power supply powers the MCU. However, for a low gate voltage and the high source voltage of Q1, Q1 acts as a closed switch. Therefore, power is supplied to the MCU via the USB, as indicated by LED D12.
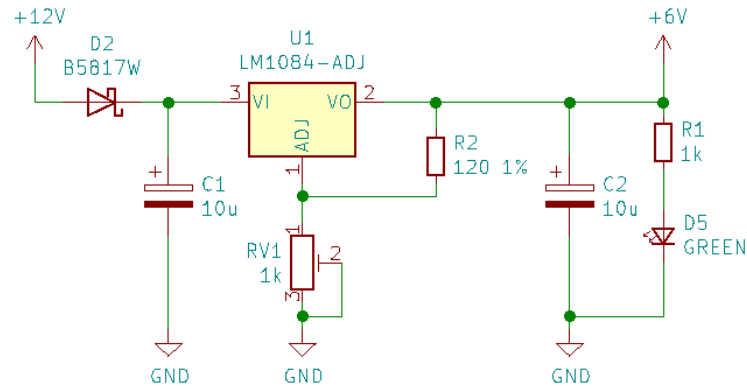
Figure 21. Electronic schematic diagram for the servo motor power supply.

Figure 21. represents a schematic diagram for the servo motor power supply. It consists of a U1 adjustable low dropout positive regulator IC LM1084 configured as a 6V regulator. According to a regulator datasheet, passive components C1, R2, RV1, and C2 are selected. Input Schottky diode D2 is used as opposite polarity protection for connection. Additional LED D5 is added to indicate power supply is connected and operational.
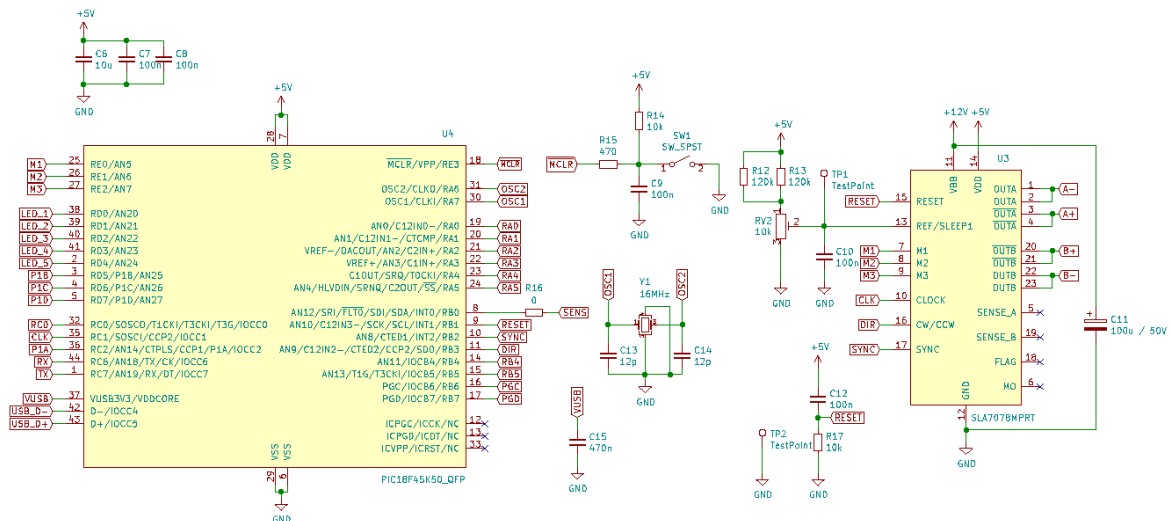

Figure 22. Electronic schematic diagram for MCU and step driver.

Figure 22. represents a schematic diagram for MCU and step driver. From the left side of Figure 22. it can be seen everything that MCU controls. The configuration of the pins and their function is described in Table 7.

Table 7. Description and functionality of the used I/O that MCU controls.

Pin	Digital configuration	Function
M1, M2, M3	Output	Controls the resolution of the step driver for driving step motor.
LED_1, LED_2, LED_3, LED_4 and LED_5	Output	Multipurpose indicator LED-s used mainly for debugging.
P1A, P1B, P1C, P1D	Output	PWM outputs for control of the servo motor, where four servo motors can be controlled separately.
CLK	Output	Step clock for step driver.
Rx, Tx	Input / Output	Pins used for serial RS232 communication.
USB_D-, USB_D+	Input / Output	USB data communication pins.
OSC1, OSC2	-	Used for an external oscillator.
SENS	Input	Used for limit switch information located under the spectrometer.
RESET	Output	Used for resetting the step driver.
SYNC	Output	Synchronous PWM control switch for step driver.
DIR	Output	Used for step motor rotation direction control.
PGC, PGD	Input / Output	Used for programming MCU.
RA0-RA5, RB4, RB5	Input / Output	Auxiliary multipurpose pins.
RC0	Output	Used for shutter control.

According to their datasheets, all passive components connected to an MCU and step driver are selected and connected. A full schematic circuit diagram can be seen in Appendix C.

5.2.3. PCB layout

PCB layout design begins with selecting the type of electronic components in the schematic. There are two types of components, surface mount devices (SMD) and through-hole plated devices (TH). Therefore, components that support the SMD type were selected, and the rest of the components were the TH type, i.e. step driver and connectors. However, there are different package sizes for the SMD components as well. Since the board dimensions are fixed (100 x 80 mm) and the area on the board is limited, the smallest SMD packages were used to fit all the components on it. It should be noted that the components in the power distribution part of the schematic (Figure 20 and Figure 21) were selected according to their power dissipation. Detailed information about the components and their respective type and package are listed in a bill of material (BOM) table at Appendix D. Next, using a generic standard for circuit board design IPC-2221 [109] that specifies PCB design requirements, the number of layers for the PCB, the width of the traces, and the size of the interconnecting points, called vias, had to be selected. Due to the high density of the components, two layers were used, i.e., one copper layer on the top and one on the bottom of the PCB, to allow the traces to cross over each other, resulting in easier connections. The width of the traces used to connect the components was determined according to their current rating. For the 12V main power line and the 6V power line for the servo motor, the trace width was set to 1.5 mm, as this allows current to flow up to 3A. For the 5V power supply line, the trace width was set to 0.5 mm and for the remaining data lines to 0.3 mm. The diameter of the vias was set to 0.8 mm.

(**a**)   (**b**)

(**c**)
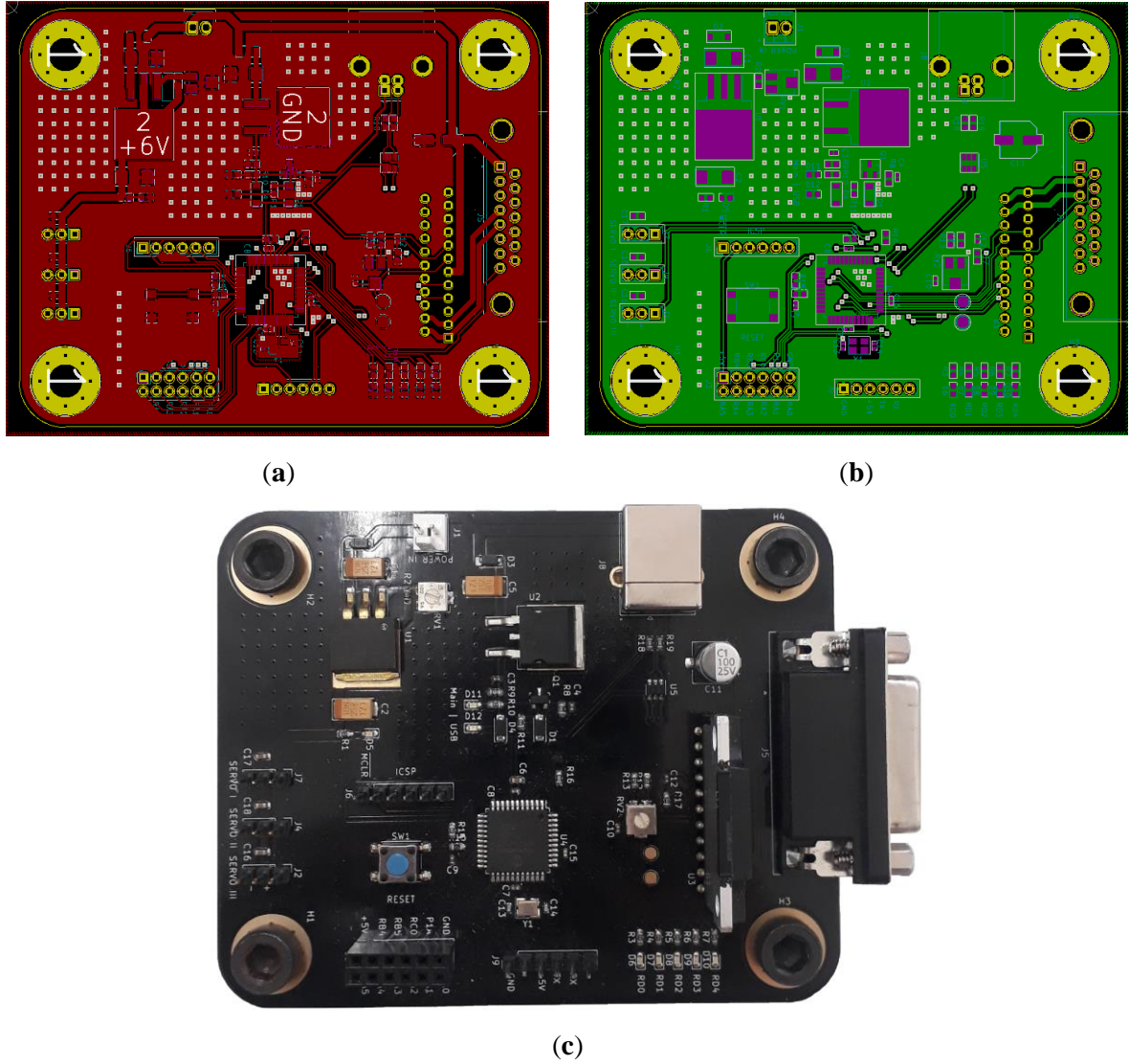
Figure 23. Completed two-layer PCB layout for an electronic hardware used in the prototype, where (a) is the top layer, (b) is the bottom layer and (c) is the assembled PCB.

The final PCB layout is shown in Figure 23a,b, where the red layer represents the top copper layer and the green layer represents the bottom copper layer. The components were placed so that on the upper left side of the PCB is the power supply circuit, on the lower part is the MCU along with an additional input/output header, RS232 communication pins and additional LED indicators. On the left side of the PCB are the connectors for the servo motors and on the far right of the PCB is the connector for the stepper motor. It should be noted that the upper and lower copper layers consist of a common ground plane connected with vias. Therefore, multiple vias were inserted, especially in the area of power distribution on the PCB, as this increases the total ground plane and thus increases the efficiency of thermal dissipation of the PCB. Next, the PCB layout along with a BOM list was sent to a PCB factory where the PCB would be manufactured and assembled. Figure 23c shows the finished and populated PCB for the proposed spectrometer.

5.3. Micro-Raman spectrometer assembly

The micro-Raman spectrometer with the variable spectral resolution was assembled after all necessary components were selected and designed.

To achieve this, the Raman spectrometer shown in Figure 24a is configured to replace the spectrometer's second collimating mirror with zoom optics (Figure 24b).
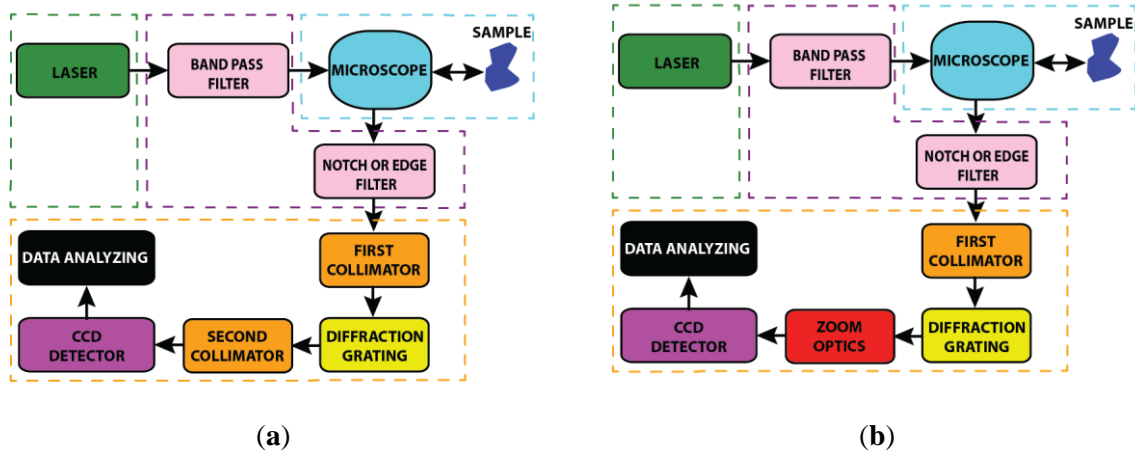

(**a**)   (**b**)

Figure 24. Implementation of a zoom optics in Raman spectrometer design to achieve spectrometer with variable spectral resolution.

Assembly begins with feeding the laser beam into the microscope. However, the laser beam must be centered on the entrance of the microscope. Therefore, the laser was placed on a stage consisting of several micrometer screws that allow precise positioning of the laser in the X, Y, and Z directions. However, it is not trivial to achieve a parallel beam throughout the system. Therefore, the laser power was set to 100 mW so that it could be measured and compared at different locations on the spectrometer. A laser power meter (Coherent Laser Check Power Meter) was used to measure the laser power. After centering the laser beam on the entrance of the microscope, the 30:70 beam splitter in the microscope was set to an angle of approximately 45° so that the reflected beam was parallel to a sample. The laser power measured at the sample was 21 mW, which proves that the beam is parallel because only 30% of the laser power is transmitted to the sample, and there is an additional power loss due to the optical losses of the optical components, such as the microscope objective. To obtain a parallel beam from the sample back to the output of the microscope, an aluminum plate was used as a sample since it reflects ~90% of the light. The reflected beam was transmitted to the output of the microscope using a 70:30 beam splitter with an angle of 45°. The beam power measured at the output was 12.5 mW, which is approximately 90% of the reflected beam and 70% of the transmitted beam. The transmitted beam was then transmitted to a focusing lens with two mirrors and then adjusted to be approximately the same height as the focusing lens and the entrance slit of the

spectrometer. The focusing lens was mounted on a movable stage so that the focal length between the lens and the spectrometer entrance slit could be adjusted. Therefore, only minor adjustments were required to focus the transmitted beam onto an entrance slit. The slit width was set to a minimum, allowing only parallel and centered beams to pass through. Inside the spectrometer, a first collimating mirror was adjusted using its tilting screw to reflect the beam onto a grating center. The beam was then reflected to an output of the spectrometer using a flat mirror. A zoom lens connected to a CCD was attached and screwed to the output of the spectrometer. Electronic hardware was placed and connected to a stepper motor, a servo motor, a shutter controller, and a PC. Finally, the shutter was placed in front of the laser, and BPF and NF filters were attached. Figure 25. shows the fully assembled micro-Raman spectrometer with variable spectral resolution.
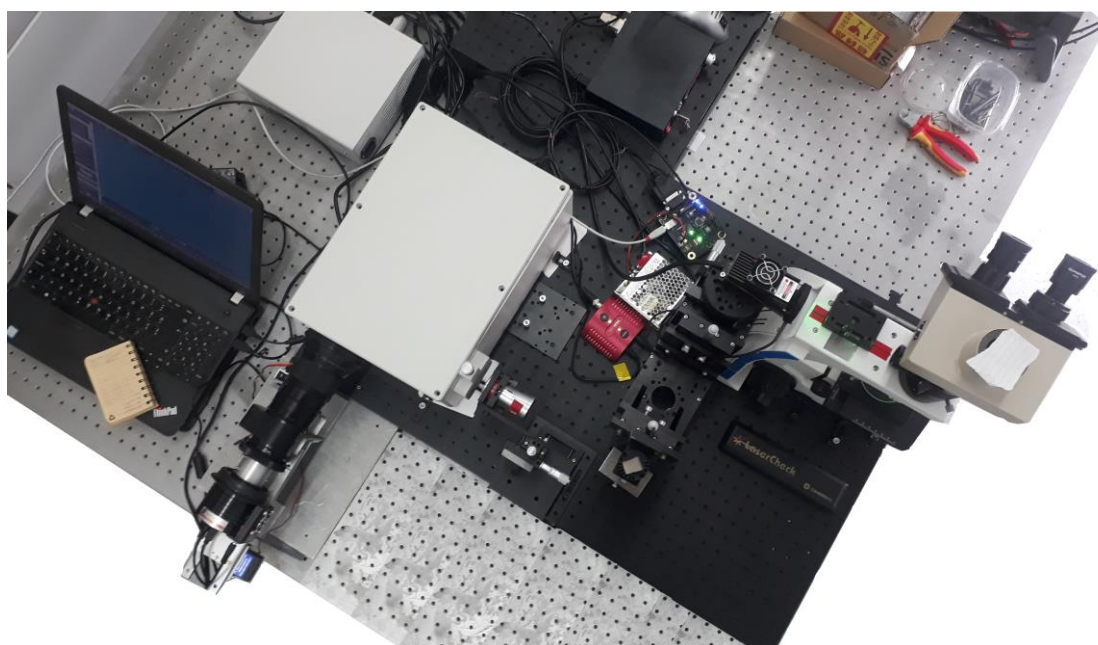

Figure 25. Fully assembled micro-Raman spectrometer for Raman and PL spectra acquisition.

5.4. Automation

An automated spectra acquisition system was implemented to acquire and analyse the samples with the proposed spectrometer and a minimum of human intervention. The Raman spectrometer with the variable spectral resolution control system is arranged as shown in Figure 26.
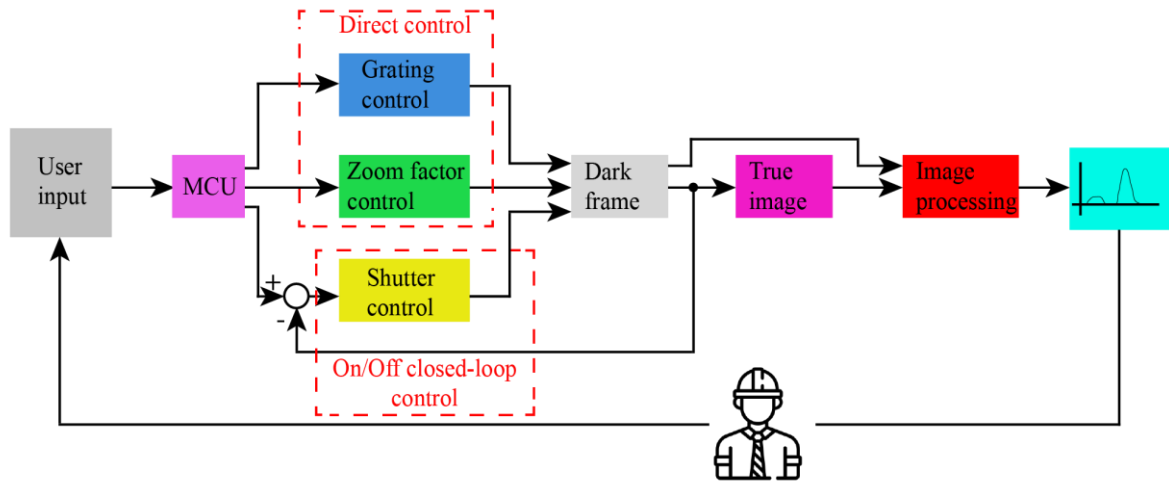
Figure 26. Raman spectrometer with variable spectral resolution control system.

The process starts with user input from the graphical user interface (GUI), where the user defines spectral range of a grating, zoom factor of the zoom lens, and exposure time for spectra acquisition. When acquisition begins, everything is done automatically, first, grating and zoom is rotated to a user-defined spectral range and zoom factor, the shutter is closed, and the dark frame is acquired. Then, the shutter opens, and a true image is acquired. Image processing consists of subtracting a dark frame from a true image, and as a result image with reduced noise is generated. Finally, the spectrum is created from a processed image and displayed in GUI. The control of the grating, zoom factor and shutter was achieved with the designed electronic board, while the spectra acquisition and processing was done with a custom designed GUI. Therefore, the firmware of the electronic board and the GUI are described next.

5.4.1. Firmware

Since the electronic board is specific hardware developed for the prototype, it requires custom software, i.e., firmware, to operate as required. Therefore, firmware is an integral part of the electronic board that processes and carries out commands received from the host, i.e., GUI. Firmware was coded in a standard C language using a proprietary freeware MPLAB integrated development environment (IDE). However, the first flow chart was created, shown as Figure 27.
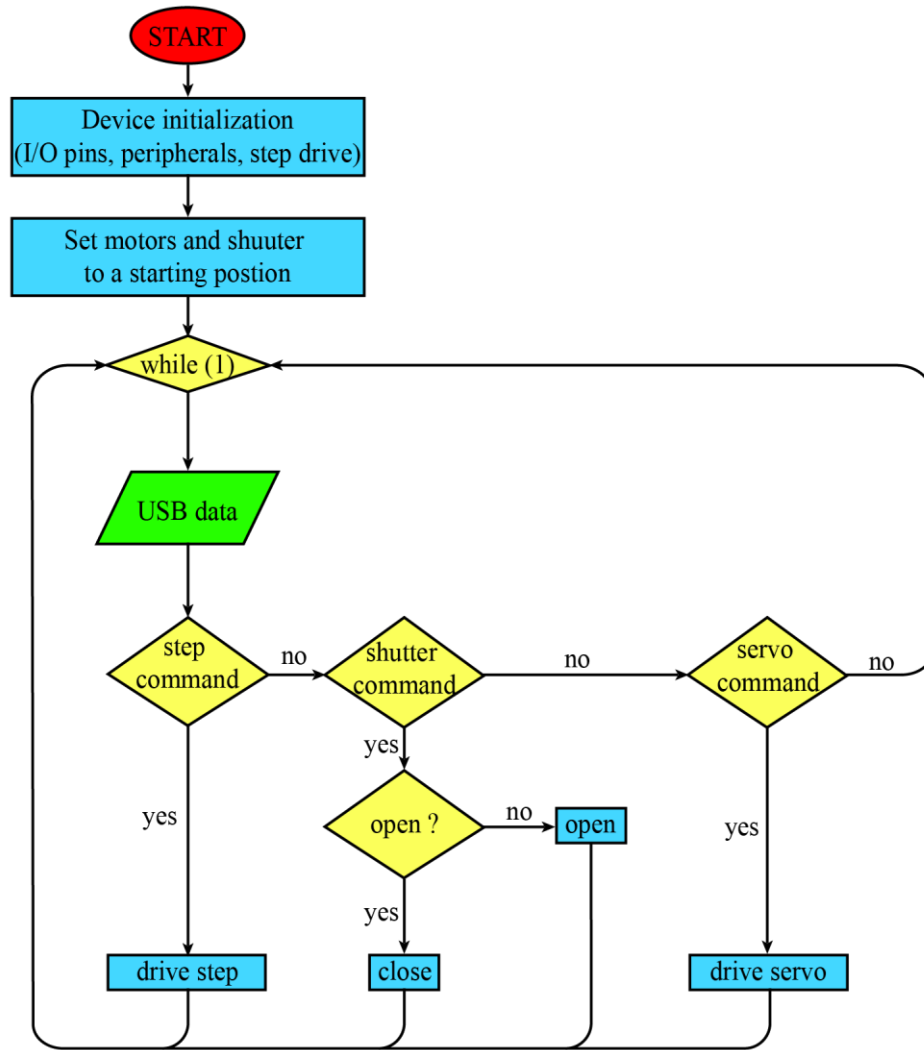
Figure 27. Flow chart of the firmware needed for electronic hardware to operate properly.

When the electronic board is powered up, the initialization process starts. During initialization, pins are defined as digital input or output, PWM and USB peripheral are enabled, and step driver resolution is selected. For the prototype, the resolution of a half step was selected because it was found out that every half-step of a motor corresponds to a spectral shift of four pixels which will suffice for all measurements. After the initialization process is done, a grating is rotated to a starting position, the zoom is rotated to a minimum zoom factor, and the shutter is closed. It has to be pointed out that starting position of a grating was set according to a laser wavelength meaning if the laser with a wavelength of 532 nm is used then the grating will rotate to an angle where the 532 nm wavelength will be positioned on the centre of the CCD. Next, an infinite while loop is created where a device (MCU) waits for USB commands from a host (GUI) and controls the motors and shutter accordingly. Source code for firmware functions is in Appendix E.

5.4.2. Software and Graphical user interface (GUI)

To control all the functions of the prototype, it was necessary to develop custom software and a corresponding graphical user interface (GUI). Software and GUI were developed using C# Visual Studio 2019 Windows Presentation Foundation (WPF).

Figure 28. Interface layout of custom developed GUI.

The custom developed GUI (Figure 28) consists of nine main components:

- CCD type (Figure 28 red dashed rectangle 1),

- CCD cooler setup (Figure 28 red dashed rectangle 2),

- data field for basic information about material name, laser type, laser power, grating type (Figure 28 red dashed rectangle 3),

- spectrometer control function for zoom factor, zoom lens shutter control, and zoom lens focus control (Figure 28 red dashed rectangle 4),

- CCD exposure time (Figure 28 red dashed rectangle 5),

- binning function for CCD (Figure 28 red dashed rectangle 6),

- dark frame enable/disable function (Figure 28 red dashed rectangle 7),

- spectrometer grating control and (Figure 28 red dashed rectangle 8),

- main area with graphical display of acquired spectra (Figure 28 red 9).
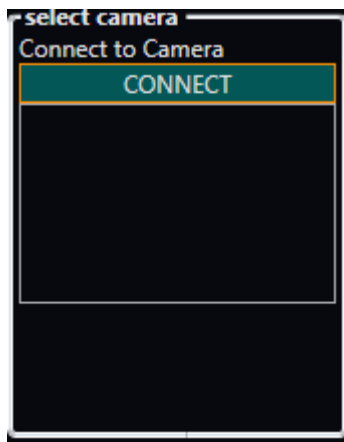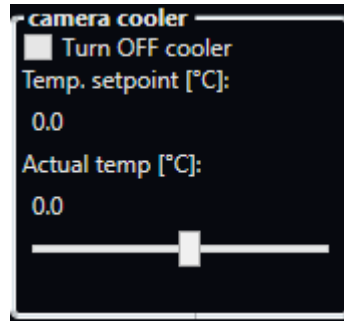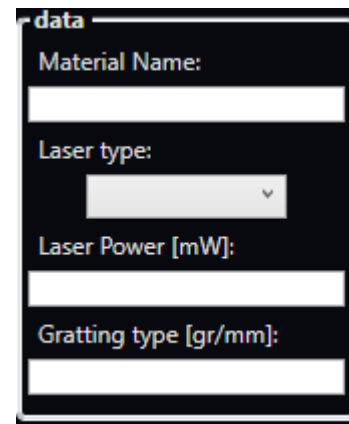
(**a**) (**b**) (**c**)

(**d**)
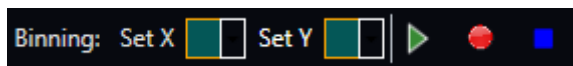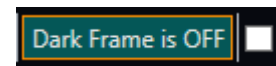
(**e**) (**f**)

(**g**)
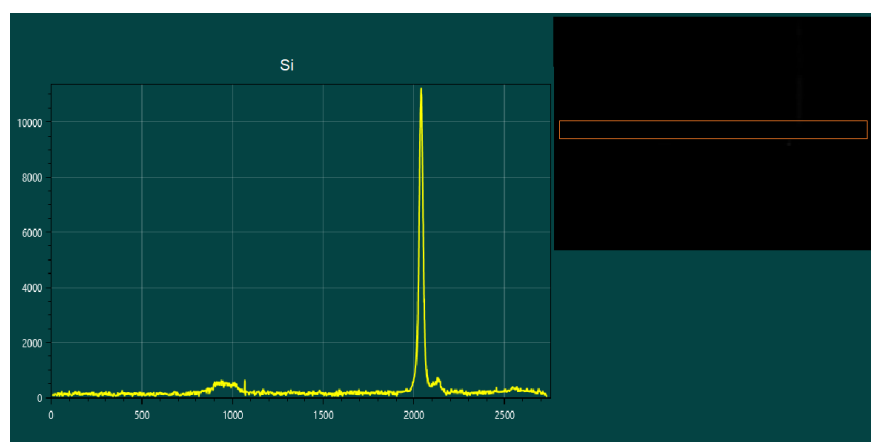
(**h**) (**i**)

Figure 29. Detailed view for a GUI control functions.

For all of these major components to operate accordingly, it is necessary to implement the logic behind GUI. The flow chart representing the logic behind is shown in Figure 30.
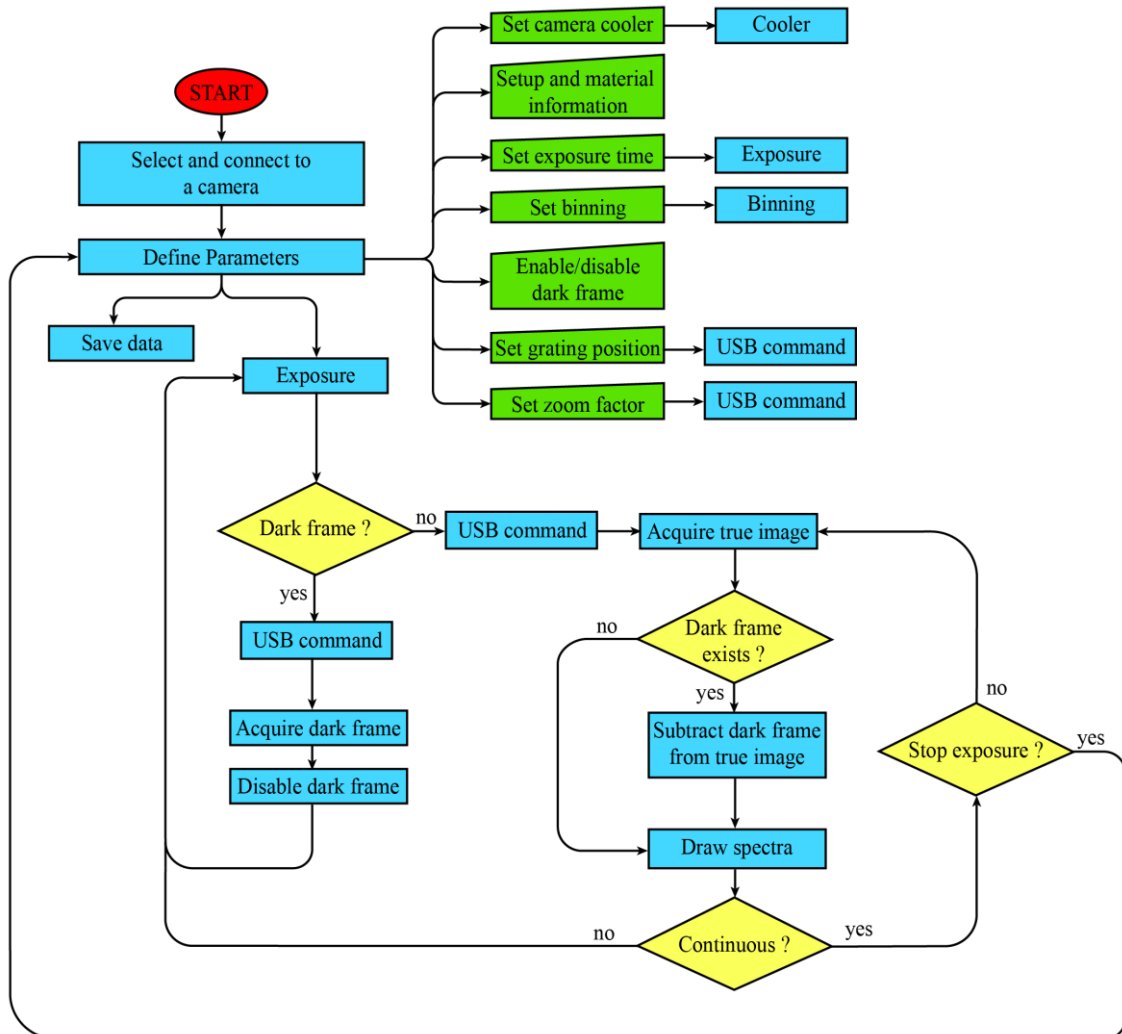
Figure 30. Flow chart of the software needed for GUI to operate properly.

When the application is first initialized, the user must select and connect to the appropriate camera model for the spectrometer (Figure 29a). This allows the user to define the parameters for recording spectra. First, the cooler must be set to the lowest possible temperature, reducing the noise generated by the camera (Figure 29b). When the user sets the cooling set point, it is automatically sent to the CCD. A minimum temperature of -20°C was set for the CCD used in this spectrometer. Next, information about the material and the spectrometer is not needed for the spectrometer to function properly, but it is good practice to have this feature for later data analysis (Figure 29c). Next, the exposure time, binning, and dark frame information will affect the results (Figure 29d,e,f). It is therefore essential to select them.

46

Detailed information about exposure time, binning and dark frame can be found in the next chapter. The position of the grating and the zoom factor are automatically sent to the electronic board via USB and positioned according to the user input (Figure 29g,h). There are two modes implemented for spectrum acquisition, namely single and continuous mode, which are self-explanatory. However, the continuous mode allows the user to continuously acquire and generate spectra while scanning the spectral range, i.e., controlling the position of the grating and the zoom factor. Single-mode is often used when the spectral range and zoom factor are known and defined. The captured image is processed when the dark frame function is activated in both modes. An algorithm has been implemented for image processing that subtracts the dark frame from the true image and will be explained in more detail in the next chapter. The captured image is stored in a 2D array that contains information about the light intensity for each pixel in the CCD. To generate spectra from the 2D pixel array, an algorithm is used that sums the corresponding pixels on the y-axis for each pixel on the x-axis. Therefore, in the generated spectra, the x-axis represents the horizontal pixel count of the CCD, and the y-axis represents the summed totals, which are arbitrary units (Figure 29i). However, Raman spectra are represented as Raman shift (cm^{-1}) on the x-axis, while PL spectra are represented as wavenumber (nm). Therefore, it is necessary to convert the pixels of the x-axis to a corresponding Raman shift or the wavenumber. First, the x-axis was calibrated, as this facilitates the conversion of pixels to the Raman shift. The calibration process was simple for the prototype by using two lasers with different wavelengths, i.e., $\lambda_{green} = 532$ nm and $\lambda_{red} = 785$ nm.

First, the zoom factor was set to a minimum 1x and the 532 nm laser was placed. Then the grating was rotated so that the laser spot could be seen on the far right of the CCD. Then the 532 nm laser was replaced with a 785 nm laser, and the spot was visible on the CCD. The distance between them was measured and was about 8 cm^{-1} per pixel. It should be noted that the calibration was performed for all different zoom factors. The result was about 0.44 cm^{-1} per pixel for the maximum zoom factor of 18x. The conversion between cm^{-1} and wavenumber was done with the formula 5 [41].

$$\lambda[nm] = \frac{1}{\frac{1}{\lambda_{ex}[nm]} - \frac{Raman\ shift\ [cm^{-1}]}{10^7}} \tag{5}$$

Since GUI will show generated spectra, user can save recorded data in a comma separated value (.csv) file. This will allow importing recorded data in some other analytical software and more detailed analysis. Source code of main GUI functions is in Appendix F.

5.4.3. Signal to noise ratio
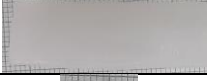
Since Raman scattering is weak, it can be masked by noise. This noise is a combination of unwanted ambient light and thermal and electrical effects in the CCD. When a CCD records the light scattered from a grating, the spectral image produced on the CCD is a thin horizontal line covering the surface of the CCD at the center of the y-axis and along the entire length of the x-axis. However, this image is composed of shot noise, dark noise, and readout noise associated with the CCD, and, finally, noise associated with ambient light. It is well-known that shot noise is associated with the square foot of an actual collected scattered light and, therefore, cannot be suppressed. The number of thermally generated electrons in a CCD generates dark noise, i.e., when there is no light, the signal level of a CCD is not zero. It has to be pointed out that this noise is strongly dependent on the device temperature and can therefore be suppressed by setting CCD to the cooling temperature of -20°C. Readout noise depends on the electronics inside the CCD when the electrons undergo analogue-to-digital conversion and amplification. It has to be mentioned that this noise cannot be eliminated, but it can be reduced by binning. Therefore, a binning of 1 x 16 is set to suppress the readout noise. At a maximum CCD resolution of 2750 x 2200, this binning produces an image with a resolution of 2750 x 137, preserving spectral resolution along the x-axis and increasing the readout speed, thereby reducing readout noise.

The exposure time determines how long the CCD captures the light, i.e., the photons emitted from the sample, increasing the signal-to-noise (S/N) ratio. Hence, the minimum and maximum exposure time range depends on the CCD and ranges from 0.001 s to 60 min. Therefore, the laser power set to 15 mW for all samples to determine the optimal exposure time. Also, the exposure time was 1 s for the Raman measurement and for the PL measurement using the trial and the error. Furthermore, to reduce noise caused by unwanted ambient light, a dark frame is used. However, a dark frame also reduces the noise generated in the CCD. Hence, the dark frame is an image taken with the shutter activated, which blocks the laser beam. It should be noted that the dark frame image must be acquired with the same parameters (exposure time, binning) that are used to acquire the true image. This image contains information about pixel values that are not part of the spectrum. Finally, they are subtracted from the true image, resulting in an image with reduced noise.

5.5. Materials

Various materials were used to record Raman and PL spectra, as listed in Table 8. All samples were provided by the Ruđer Bošković Institute. For the PL measurement, different Schott glasses were used because they provide a good PL spectrum with a known cutoff wavelength. Various minerals in solid-state were used for Raman measurement. Various powdered materials were also used for the Raman measurement. To test the effects of variable spectral resolution, sulphur and silicon were used because they provide strong Raman bands at known locations. All samples were placed on a separate microscopic glass and placed under the microscope. For each sample, the laser beam was manually focused on one sample.

Table 8. Materials used for Raman and PL spectra measurement with a prototype.

Material		Measurement	Material		Measurement
Schott 550		PL	Rhodonite		Raman
Schott 665		PL	Rock crystal		Raman
Amethyst		Raman	Rubelite		Raman
Azurite		Raman	Pure PE		Raman
Beryll		Raman	TiO_2 - P25		Raman
Garnet		Raman	Rutile commercial		Raman
SiO		Raman	TiO_2 - 800°C		Raman
Morion		Raman	Sulphur		Raman with variable ZF
Jasper		Raman	Silicon		Raman with variable ZF

6. RESULTS AND DISCUSSION

This chapter discusses the results obtained from the prototype and referent system (JY-T64000). First, the Raman spectra were recorded on silicon (Si) and sulphur (S) materials varying the zoom factor (ZF). Further, several different samples were recorded with maximum ZF. Next, for different ZFs, a spectral resolution and signal-to-noise ratio (SNR) has been analytically calculated. Then, the results are presented, compared with JY-T64000, and discussed. It has to be noted that obtained results are validated using statistical metrics, the analysis of variance (ANOVA) method. Finally, a comparison of the PL spectrum of a Schott 550 and 665 glass obtained from the prototype and JY-T64000 was performed.

6.1. Raman spectra

To record Raman spectra of all of the samples, the prototype was set to 1 s exposure time with Barlow lens set to 2x magnification, and laser power on the sample was 15 mW. The Raman spectra of the Si (Figure 31) and S (Figure 32) acquired with the prototype at the different ZFs are shown. From Figure 31 and Figure 32, the x-axis denotes Raman shift, and the unit is cm^{-1}, whereas the y-axis represents a relative intensity that has arbitrary units. It has to be highlighted that both Si and S ZF measurements were set to a value where Raman lines are visible and recognizable. Therefore, for both samples initial ZF was set to 4.5x. The grating was rotated, so the laser line was in the middle of the x-axis.

From Figure 31a,c and Figure 32a,c the spectral interval over the entire CCD region at a ZF of 4.5x ranges from 480 to 596 nm. Further, there are four noticeable spectral peaks. Spectral peaks appear around the expected wavelength of the 532 nm laser. The spectrum of Si (Figure 31a) shows a line of transferical optical (TO) vibrational mode at 520 cm^{-1}, a TO second harmonic mode at 900-1000 cm^{-1}, and the Anti-Stokes line spectrum. The S spectrum (Figure 32a) shows a strong optical phonon line at 472.8 cm^{-1}. However, it is necessary to increase a resolution by adjusting a ZF to see more spectral components. It has to be pointed out that lines become distinct with a further increase of ZF by factor 2x, i.e., from the initial value of ZF, the resolution is a 9x magnified. Figure 31b shows that for Si, Stokes and Anti-Stokes peaks become narrower due to the increased spectral resolution. The similar observation for Stokes, and Anti-Stokes peaks have been observed for S (Figure 32b). It has to be emphasized that the spectral interval over the entire CCD region at ZF of 9x ranges from 505 to 562 nm wavelength.
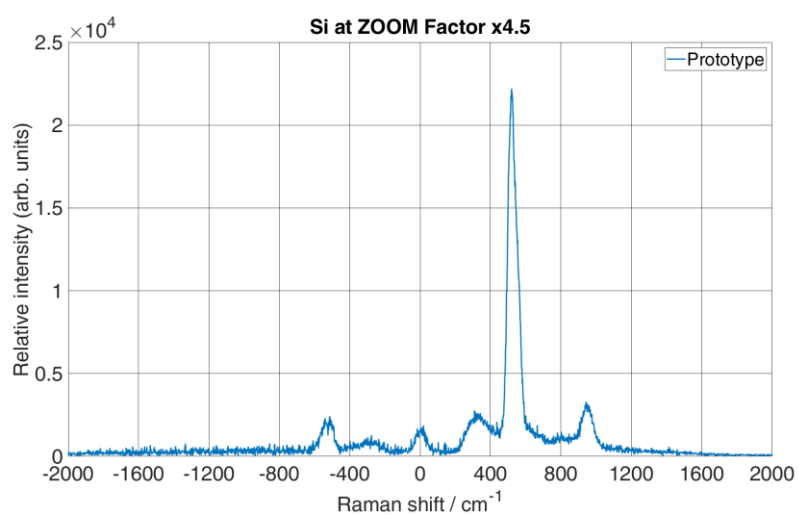
(**a**)
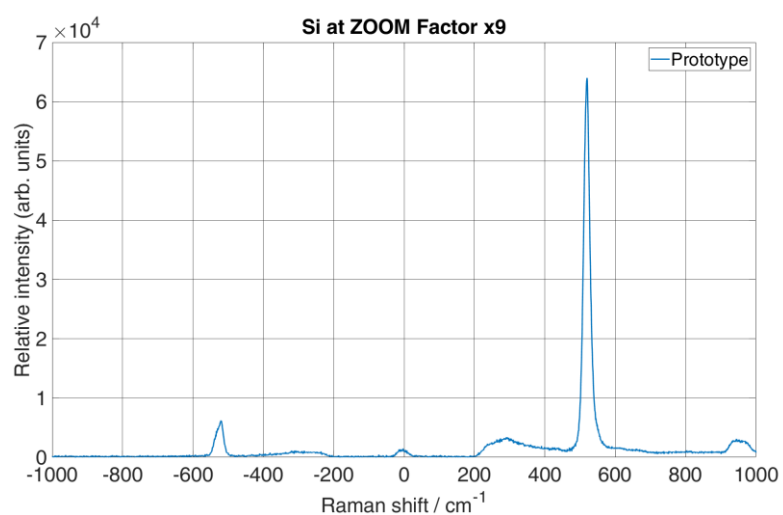
(**b**)
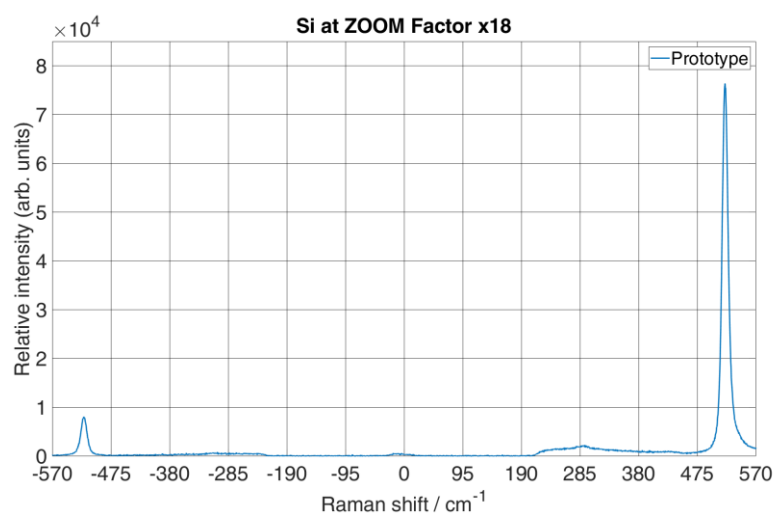
(**c**)

Figure 31. Acquired Raman spectrum with a prototype of a Si with different zoom factors 4.5x (a), 9x (b) and 18x (c).
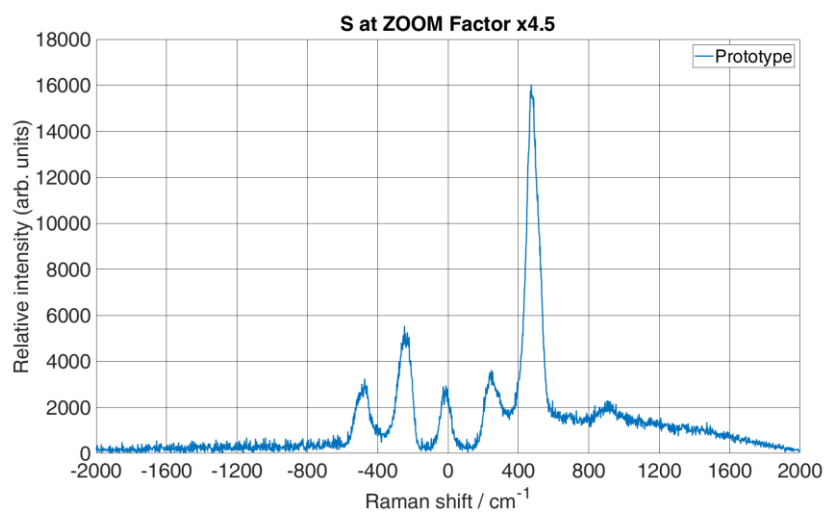
(**a**)
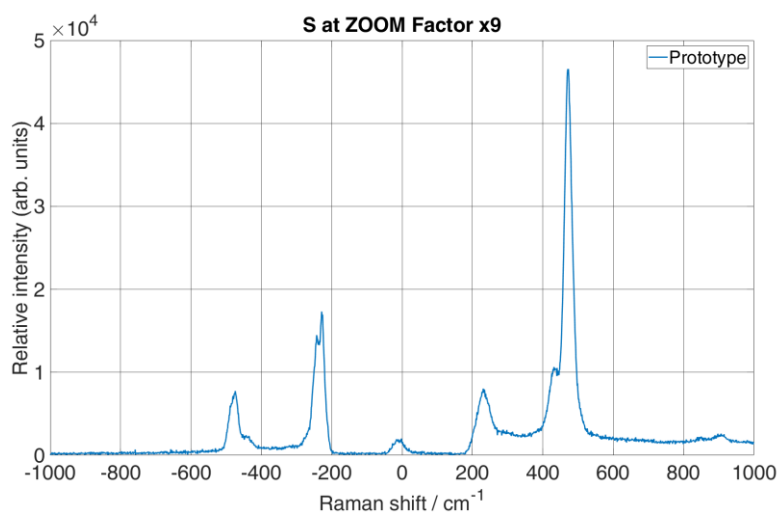
(**b**)
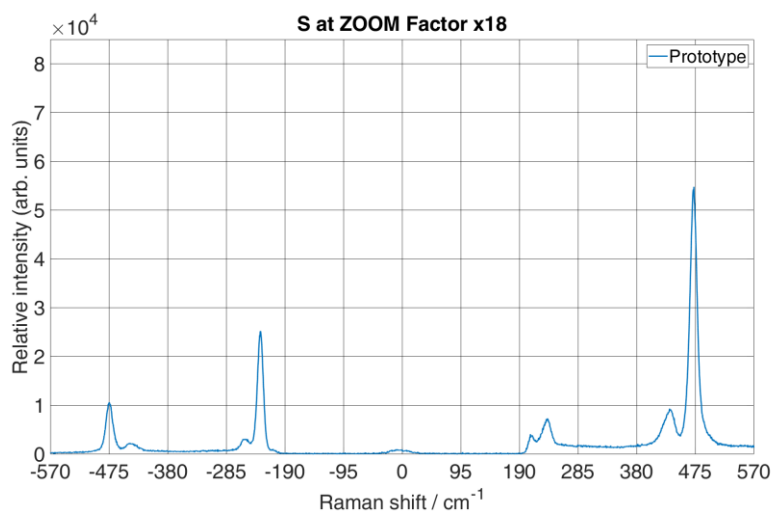
(**c**)

Figure 32. Acquired Raman spectrum with a prototype of a S with different zoom factors 4.5x (a), 9x (b) and 18x (c).

Furthermore, if the ZF is set to the maximum value of 18x (Figure 31c and Figure 32c), in that case, it can be observed that the spectral interval across the CCD region becomes even narrower, measuring 516 - 549 nm wavelengths. It has to be pointed out that spectral lines both for Si and S at this ZF are more distinct then at lower ZF. However, it can be seen from Figure 31c that the TO mode second harmonic for Si disappears. Therefore, the total spectral interval over the CCD region reduces by increasing the ZF, and the spectral resolution increases, which is evident from the line narrowing.

Furthermore, the first measure taken was the spectral dispersion which is calculated using the Equation (2) from Chapter 4 i.e., the reciprocal linear dispersion (D^{-1}). By substituting the value of the focal length f = 306 (with Barlow 2x magnification) for the maximal ZF, the order of diffraction n = 1, and the distance between rulings d = 1/1100 mm, the reciprocal linear dispersion (D^{-1}) for the first-order diffraction is 2.97 nm/mm. The per pixel width of this value is 0.0135 nm/pixel. The spectral dispersion at the position of the Raman band for Si at 520 cm^{-1}, as shown in Figure 31c (547.14 nm wavelength of light), and for S at 472.77 cm^{-1}, as shown in Figure 32c (545.73 nm wavelength of light), is 0.48 cm^{-1}/pixel. The same calculation for the reference system gives a dispersion of 0.756 cm^{-1}/pixel which is 1.575 times higher value. Therefore, in order to have a similar spectral resolution, a binning factor of 1x was used during recording by the prototype.

Spectral resolution depends on an input slit of a spectrometer. Therefore, the input slit width for the prototype was set to 35 μm. At the maximal ZF of 18x, the total magnification of the monochromator is $f_{co2}/f_{co1} = 1.33$, which gives an image size of 46.55 μm at the CCD. From the Nyquist limit, it can be determined whether the CCD can differentiate between the two neighbouring spectral lines, i.e., three times the image size over the CCD. Therefore, the CCD can differentiate between two adjacent spectral lines when their peaks are separated 139.65 μm on the CCD, which spectrally is 14.76 cm^{-1}. This can be seen from Figure 32b at a position about -240 cm^{-1}, where two peaks appear with a 15 cm^{-1} difference between them, while in Figure 32a same Raman band is shown as one peak.

Figure 33. shows the Raman spectra of several different materials in comparison with the reference Raman spectra. It has to be pointed out that the prototype was set to the maximum of 18x magnification so it could record approximately the same spectral range as the reference system, therefore the spectral interval from 540 - 571 nm was recorded.
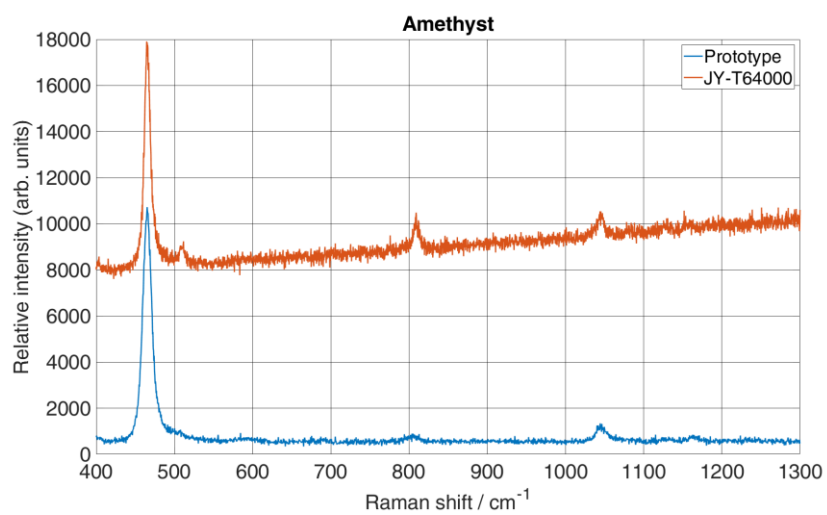
(**a**)
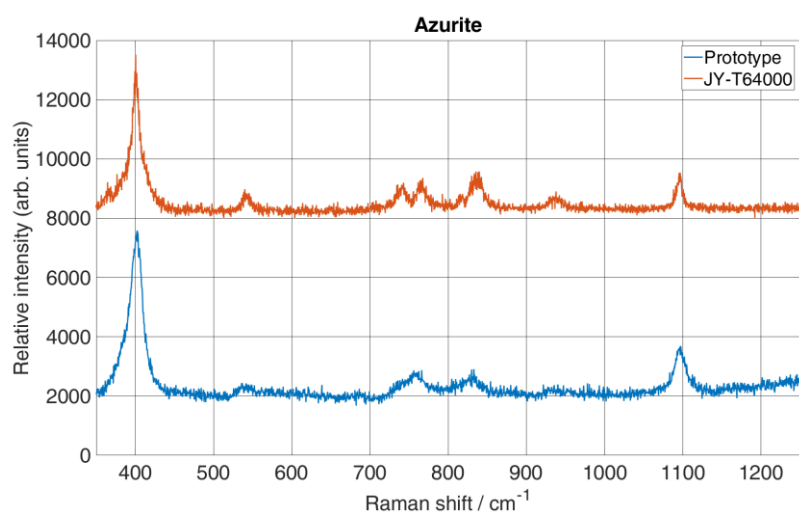
(**b**)
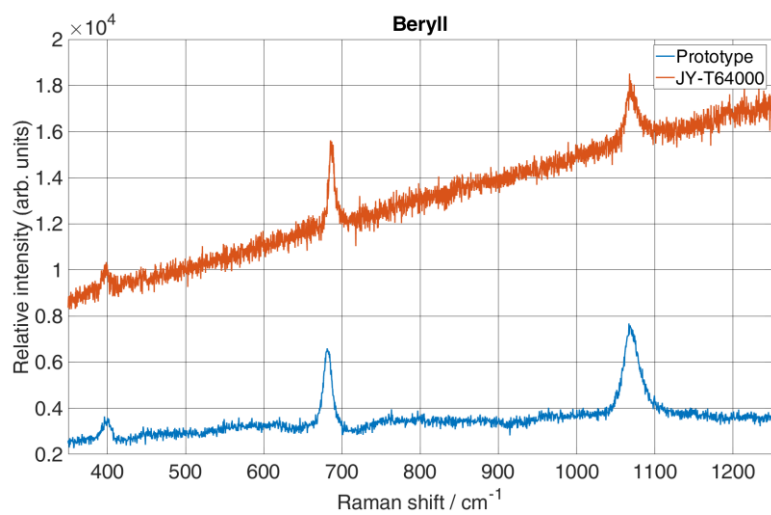
(**c**)
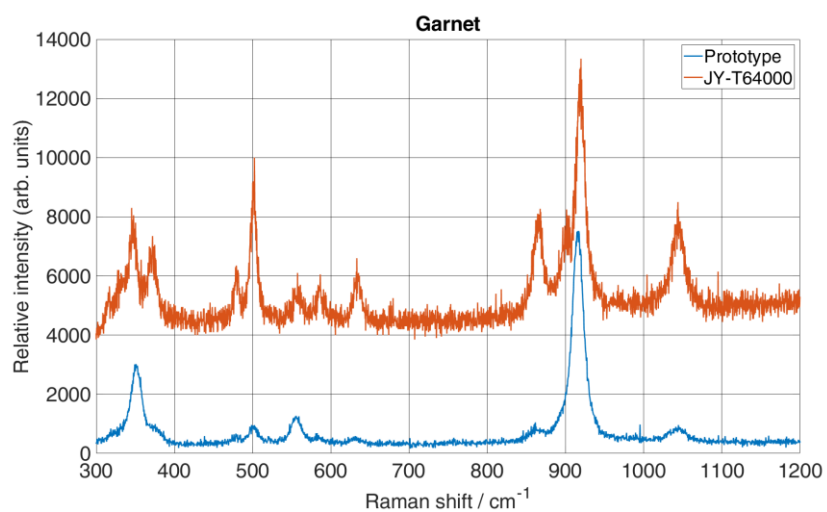
(**d**)
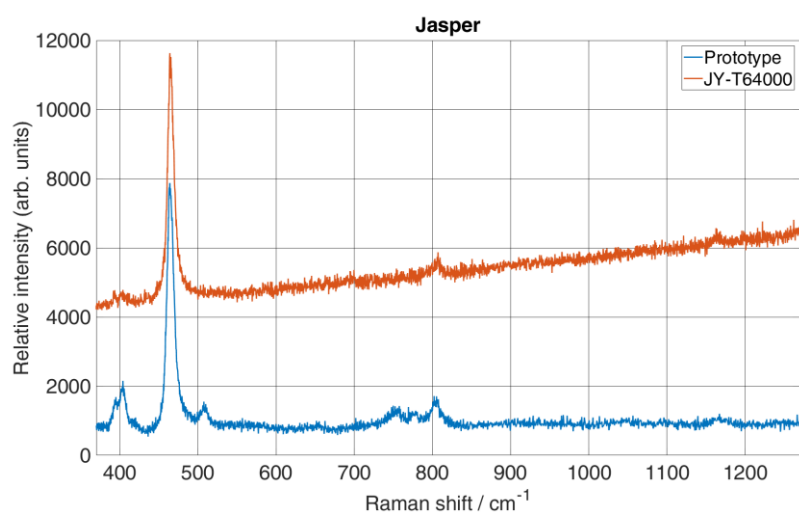
(**e**)
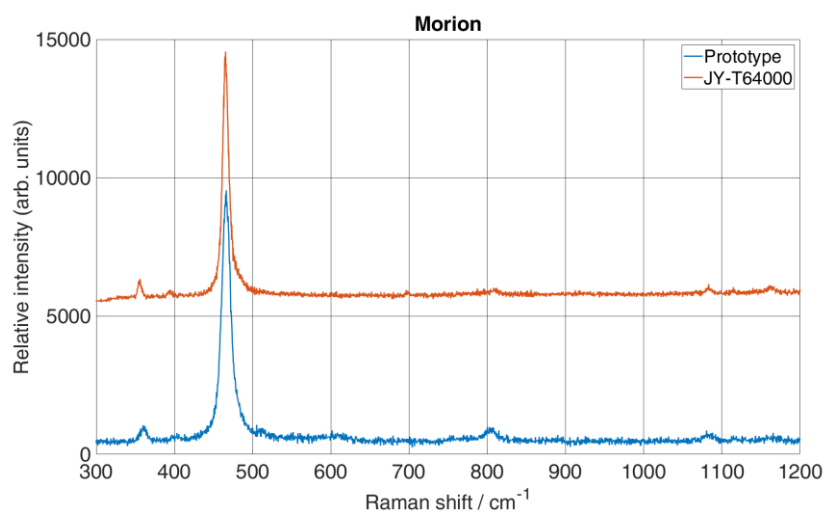
(**f**)
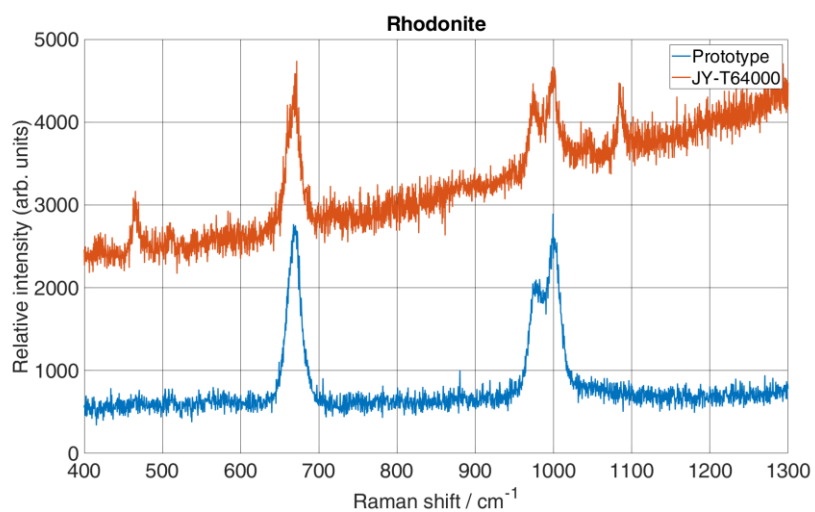
(**g**)
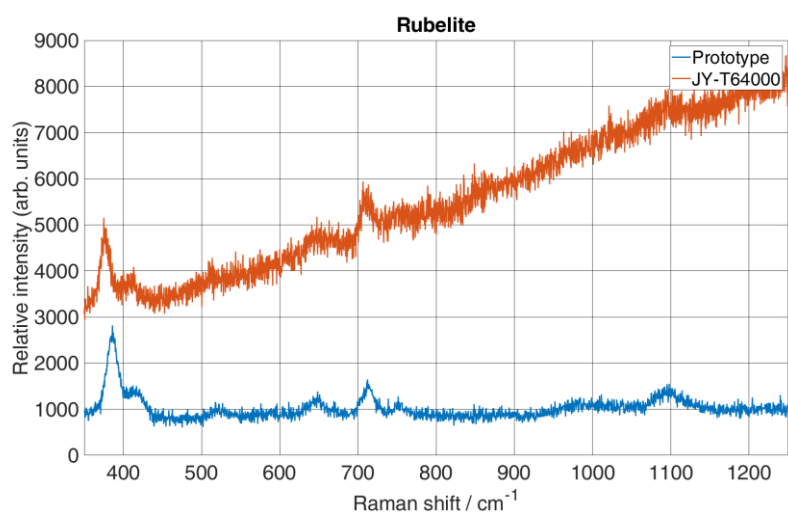
(**h**)
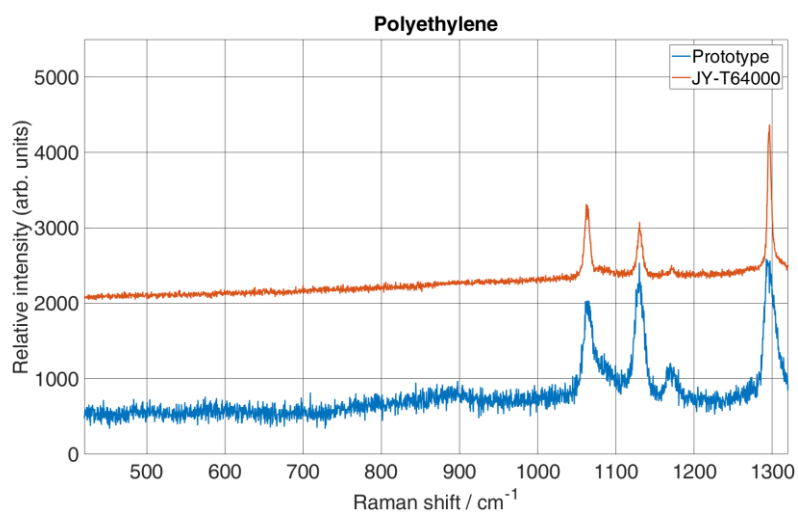
(**i**)
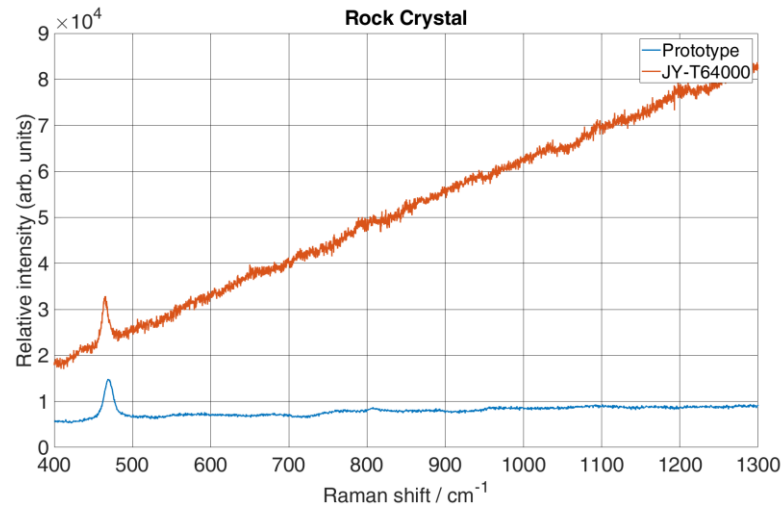
(j)
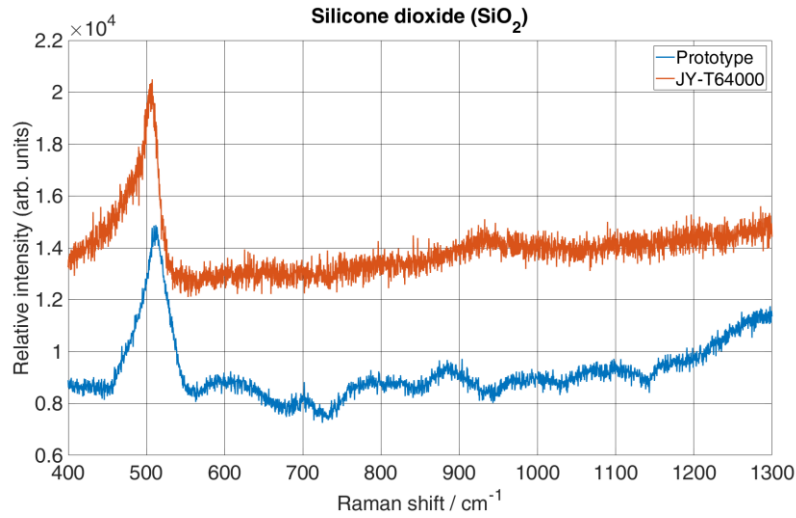
(k)

Figure 33. Comparison of the prototype Raman spectra of a several different materials with a reference system.

It should be noted that for the prototype, the intensity is calculated as the sum of the y-axis pixels for each x-axis pixel, whereas for the reference system, the intensity is calculated as the average of the y-axis pixels for each x-axis pixel. Therefore, to better compare the Raman spectrum of the reference system with that of the prototype, the intensity was multiplied by an arbitrary factor so that the intensity of each spectrum was approximately equal to the intensity of a spectrum obtained with a prototype. In addition, the Raman spectrum obtained with the reference system was shifted along the y-axis to facilitate comparison, as shown in Figure 33 and Figure 34 for all spectra. Furthermore, it should be pointed out that spectral range using maximum ZF is approximately two times larger than the spectral range of the reference system. Therefore, Raman spectra of the reference system was obtained by stitching two small sections

of the spectra into one spectrum. Furthermore, exposure time for both the prototype and reference system was 1 s (note that for the reference system it was total of 2 s because of recording two small sections), and the microscope objective used in the reference system was 50x-0.5 NA, whereas for the prototype was 80x-0.7 NA. Laser power for the prototype was 15 mW, whereas for the reference system was 11.5 mW.

It can be seen from Figure 33 that all Raman spectra recorded with the prototype have similar spectra to the Raman spectra of the reference system. However, the reference system performs better in some measurements. Therefore, the reference system shows additional Raman lines for Amethyst (Figure 33a) at about 510 cm^{-1}, for Azurite (Figure 33b) at about 740, for Garnet (Figure 33d) at about 380 and 590 cm^{-1}, for Jasper (Figure 33g) at about 460 and 1090 cm^{-1}. This can be contributed to a high-resolution spectrum obtained by stitching two smaller sections. However, the prototype also performs better for Jasper (Figure 33e) where distinct lines can be seen at about 400 cm^{-1}, and additional lines at about 510 and 750 cm^{-1}. It is also evident from Figure 33 from all of the spectrums obtained by the reference system that SNR is better than the prototype, which shows better results for lines that are small enough to be hidden under the noise. Therefore, (note that the reference spectrum was multiplied by an arbitrary factor so the spectrums can be compared), for Rhodonite (Figure 33g), the Raman lines at 460 and 1000 cm^{-1} are clearly visible with the reference system, while the prototype shows none at those locations.

(**a**)
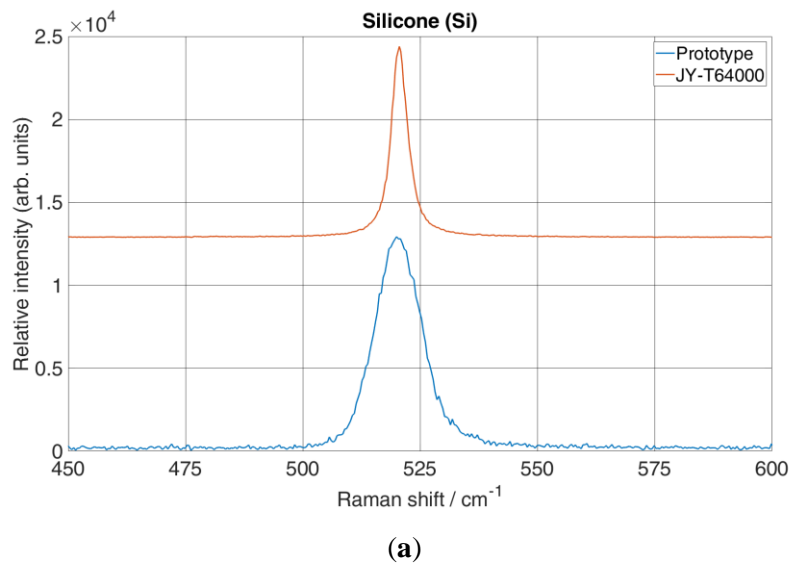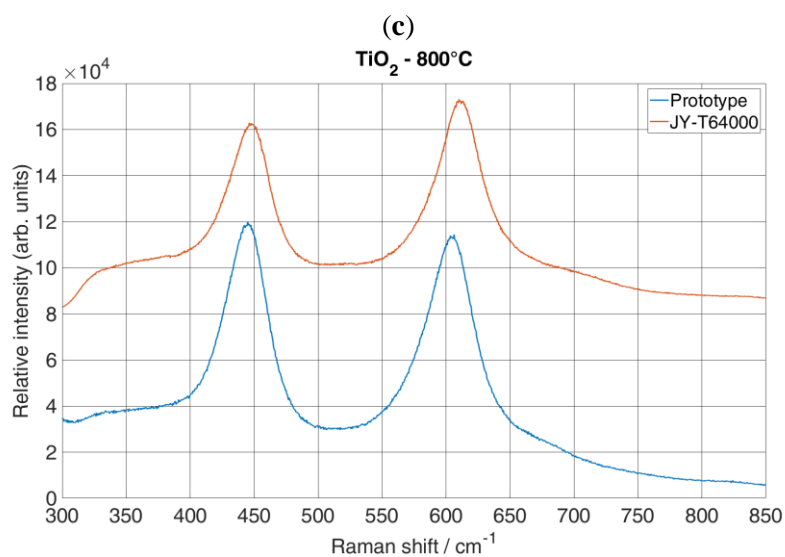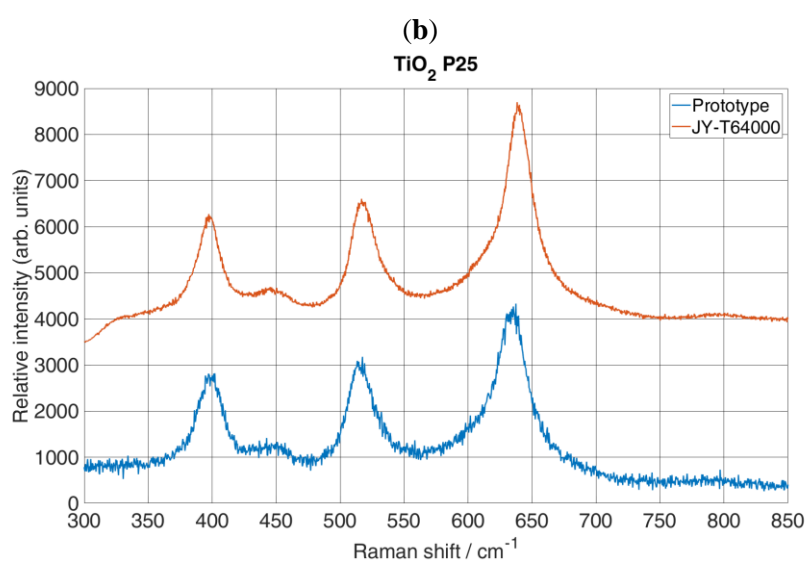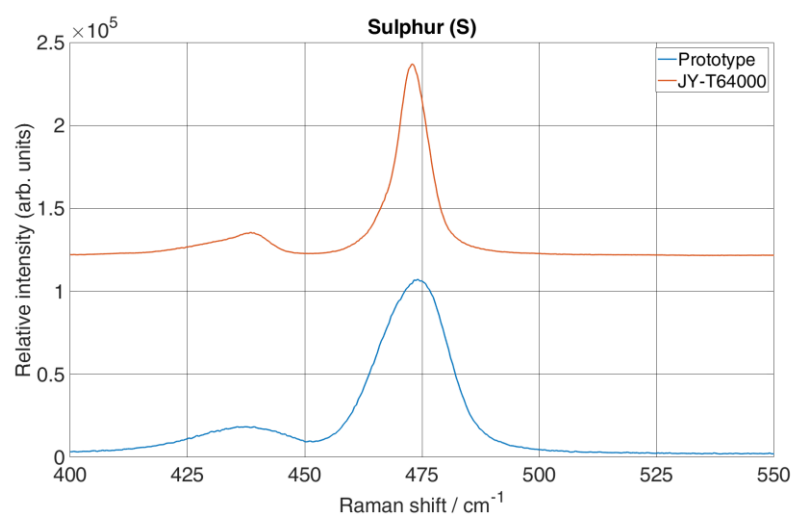
Figure 34. Spectrums obtained by reference system and proposed system used for difference analysis between two systems.

To evaluate the collected data between the two systems, a comparison of the spectra was calculated using ANOVA analysis [110–112]. Figure 34 shows the spectra of the prototype and the reference system for the analysis ANOVA. It should be noted that the spectral interval was set according to the strong Raman lines for each material. Therefore, the interval 450 - 600 cm^{-1} was chosen for Si (Figure 34a), 400 - 550 cm^{-1} for S (Figure 34b), and 350 - 800 cm^{-1} for TiO$_2$ P25 and TiO$_2$ - 800° (Figure 34c,d). An ANOVA table was calculated for each material using Excel. The calculated ANOVA analysis is shown in the Table 9. It contains the values for the sum of squares (*SS*), the degrees of freedom (*df*), the mean square values (*MS*), the *p*-values, and the calculated *F* and $F_{critical}$ values. The significance level alpha of 0.05 was chosen for all materials. However, the null hypothesis used for the analysis of ANOVA was that there is no significant difference between two sets of data, which means that the *F*-value must be less than the $F_{critical}$ value and the *p*-value must be greater than the alpha value. If the *F*-value is greater than the $F_{critical}$ value and the *p*-value is less than alpha, the null hypothesis is rejected, and the alternative hypothesis is accepted. In Table 9, the calculated *F*, $F_{critical}$ and *p*-values show no statistical differences in all the compared materials ($F < F_{critical}$ and $p > 0,05$), so the null hypothesis was not rejected.

Table 9. Calculated ANOVA analysis for Raman peaks obtained with two systems using alpha value of 0.05.

Interval [cm^{-1}]	Material	*SS*	*df*	*MS*	*F*	*p*-value	$F_{critical}$
	Si						
450 - 600	Between groups	63.43	1	63.43	0.035	0.86	3.86
	Within groups	1580470	852	1855.02			
	Total	1580533	853				
	S						
400 - 550	Between groups	5.79	1	5.79	0.003	0.96	3.86
	Within groups	1623253	862	1889.70			
	Total	1623259	863				
	TiO$_2$ P25						
350 - 800	Between groups	9096.03	1	9096.03	0.44	0.51	3.85
	Within groups	60507703	2899	20871.92			
	Total	60516799	2900				
	TiO$_2$ – 800°						
350 - 800	Between groups	5382.30	1	5382.30	0.32	0.58	3.85
	Within groups	44032882	2603	16916.21			
	Total	44038264	2604				

6.1.1. Spectral resolution

Spectral resolution of a prototype was calculated by measuring the linewidth for the line at 520 cm^{-1} for Si (Figure 31a) and for the line at 472 cm^{-1} for S (Figure 32a). Therefore, the Gaussian function (Equation (1) from Chapter 4) is used to fit Raman lines and Equation (2) from Chapter 4 shows a full width at half maximum (FWHM) for the stokes line of a Si and S Raman spectra at a minimum and maximum ZF.

Figure 35a,b shows measured linewidth for S, while Figure 36a,b shows measured linewidth for Si at the minimal and maximal ZF by fitting the Gaussian profile.

Using Gaussian profile (Figure 35a,b) measured linewidth at the minimal and maximal ZF for 472 cm^{-1} Raman band results in 76.97 cm^{-1} at the minimum ZF and 18.25 cm^{-1} at the maximum ZF. It has to be pointed out that due to the increased resolution new Raman bands also appear. From Figure 36a,b measured linewidth at the minimal and maximal ZF for 520 cm^{-1} Raman band results in 63.61 cm^{-1} at the minimum ZF and 12.81 cm^{-1} at the maximum ZF.
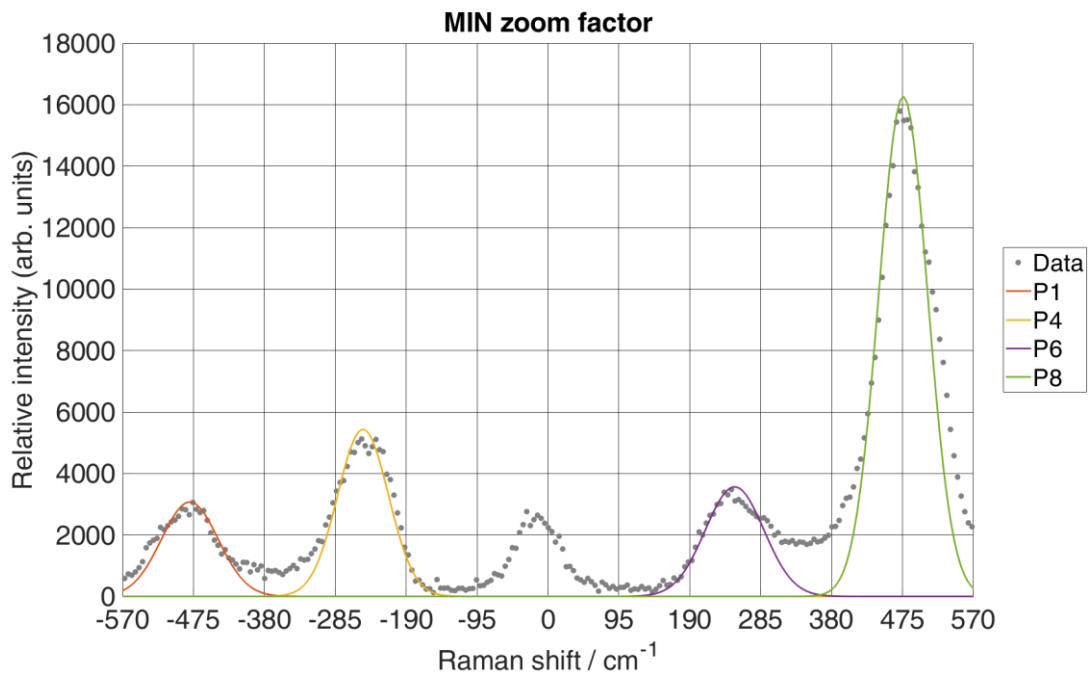
(**a**)
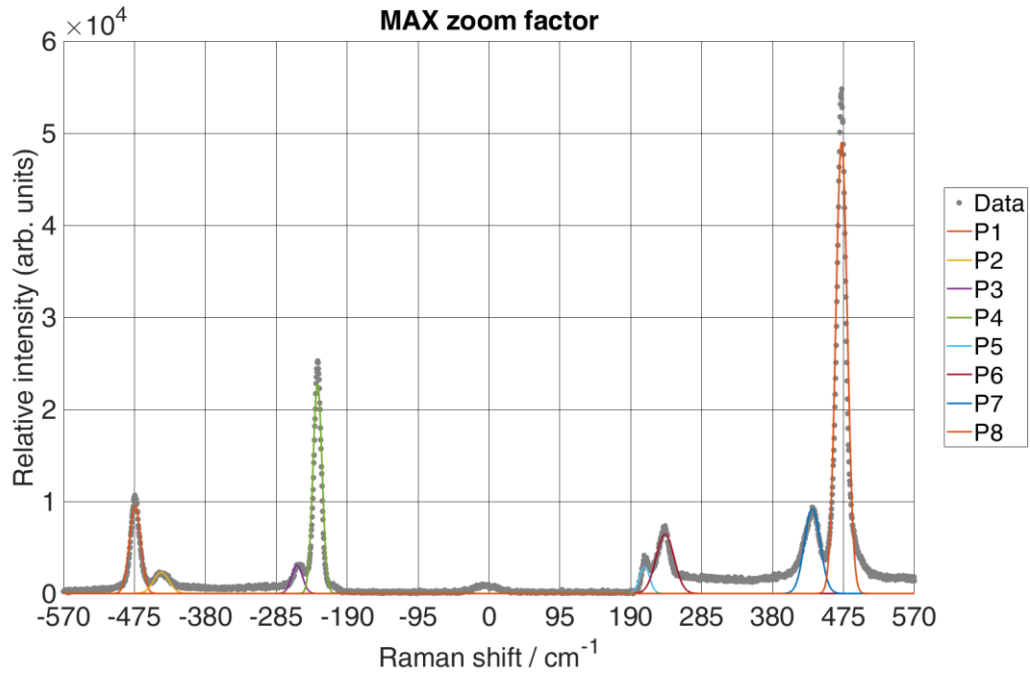
(**b**)

Figure 35. Calculation of the spectral resolution of the prototype with S Raman spectrum using the Gaussian fit for each peak at different ZF. Note that peak at location 0 cm^{-1} is the peak of the excitation source. Therefore, Gaussian fit was not calculated for that peak.

Table 10 shows the Gaussian fitting parameters for each peak at the minimum and maximum ZF, together with their FWHM value. It is evident from the calculated linewidth of the characteristic peak at 472.44 cm^{-1} that the spectral resolution increased significantly, i.e., the spectral resolution became more than four times larger.

Table 10. Gaussian fit parameters for obtained Si spectral peaks.

Curves	MIN ZF				MAX ZF			
	A	b[cm^{-1}]	σ	FWHM	A	b[cm^{-1}]	σ	FWHM
P1	3073.43	-479.84	37.99	89.48	9557.08	-474.18	7.55	17.8
P2	n/a	n/a			2308.23	-439.01	10.33	24.30
P3	n/a	n/a			3064.33	-257.13	7.26	17.11
P4	5439.25	-247.75	34.02	80.11	22768.3	-229.7	6.08	14.33
P5	n/a	n/a			2932.75	208.63	6.69	15.74
P6	3566.65	250.08	39.2	92.28	6432.75	236.03	12.07	28.43
P7	n/a	n/a			9199.62	433.24	10.51	24.76
P8	16268.6	475.94	32.68	76.97	49008.5	472.44	7.76	18.25

Using the values $\Delta\tilde{v}_{slit}$ = 4.92 cm^{-1}, $\Delta\tilde{v}_{line}$ = 9 cm^{-1} (measured by the reference system without binning factor), and fitted FWHM = 18.25 cm^{-1} for the characteristic peak at 472.44 cm^{-1} at maximum ZF (Table 10). By Equation (3) from Chapter 4, it was found that $\Delta\tilde{v}_{resolution}$ = 15.09 cm^{-1}. Furthermore, for the same peak and fitted FWHM = 76.97 cm^{-1} at the minimum ZF (Table 10), it was found out that $\Delta\tilde{v}_{resolution}$ = 76.28 cm^{-1}, which is 5 times worse resolution in comparison with the ZF = 18. However, the initial ZF = 4.5 is not exact, meaning that there are slightly aberrations due to the motor positioning the ZF on the zoom lens. Therefore, spectral resolution changes approximately the same as the ZF since the difference between the initial minimum ZF = 4.5 and maximum ZF = 18 is 4 times.

(**a**)
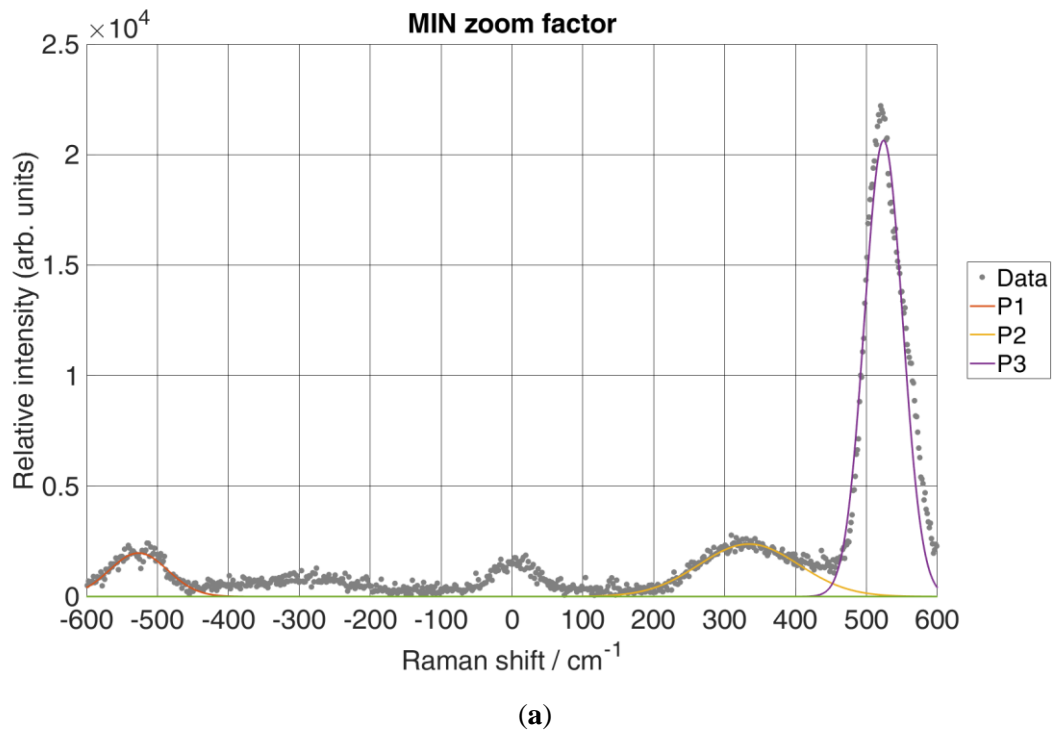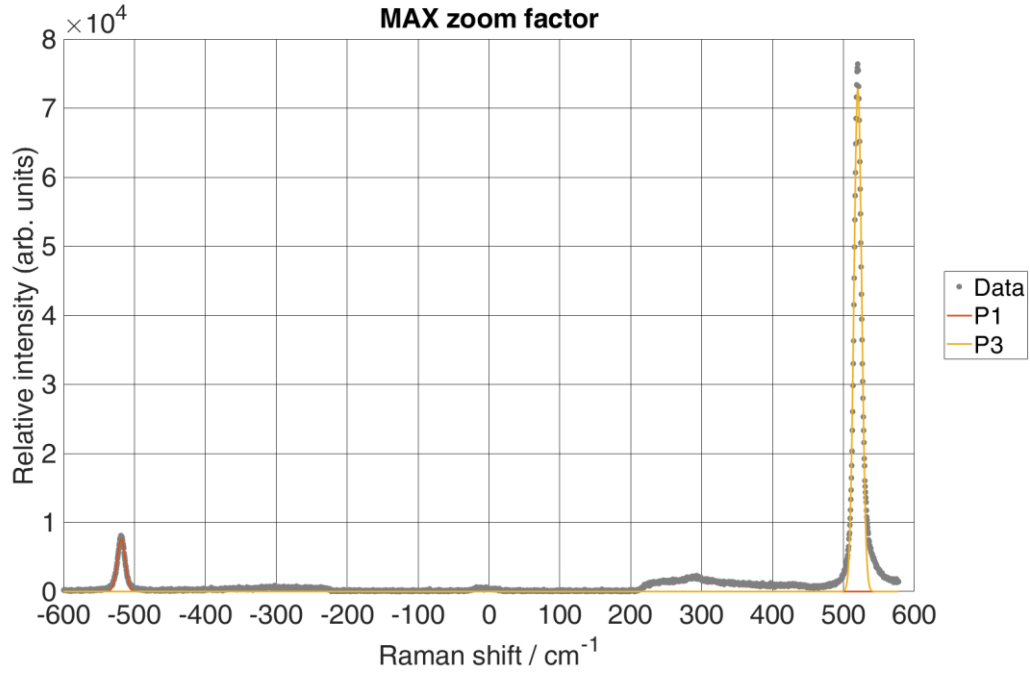
Figure 36. Calculation of the spectral resolution of the proposed system for Si Raman spectrum using Gaussian fit for each peak at different ZF. Note that peak at location 0 cm^{-1} is the peak of the excitation source. Therefore, Gaussian fit was not calculated for that peak.

Table 11 shows the Gaussian fitting parameters for each peak at the minimum and maximum ZF, together with their FWHM value. It is evident from the calculated linewidth of the characteristic peak at 520.24 cm^{-1} that the spectral resolution increased significantly, i.e., the spectral resolution became more than four times larger.

Table 11. Gaussian fit parameters for obtained S spectral peaks.

Curves	MIN ZF				MAX ZF			
	A	b[cm-1]	σ	FWHM	A	b[cm-1]	σ	FWHM
P1	1967.66	-526.66	41.20	97.02	7636.75	-518.7	5.67	13.35
P2	2372.75	334.23	70.51	166.05	n/a	n/a		
P3	20646.45	524.01	27.01	63.61	72894.81	520.24	5.44	12.81

Using the values $\Delta\tilde{v}_{slit} = 4.92$ cm^{-1}, $\Delta\tilde{v}_{line} = 4$ cm^{-1} (measured by the reference system without binning factor), and fitted FWHM = 12.81 cm^{-1} for the characteristic peak at 520.24 cm^{-1} at maximum ZF (Table 11). By Equation (3) from Chapter 4, it was found that $\Delta\tilde{v}_{resolution}$ = 11.13 cm^{-1}. Furthermore, for the same peak and fitted FWHM = 63.61 cm^{-1} at the minimum ZF (Table 11), it was found out that $\Delta\tilde{v}_{resolution}$ = 63.29 cm^{-1}, which is 5.6 times worse resolution in comparison with the ZF = 18. This difference in spectral resolution is the same as

mention above, slightly aberrations due to the motor positioning the ZF of the zoom lens. Using the values $\Delta\tilde{v}_{slit} = 1{,}01$ cm^{-1}, $\Delta\tilde{v}_{line} = 4$ cm^{-1} and FWHM = 5.66 cm^{-1} (calculated by Gaussian fit for the characteristic peak at 520.62 by the reference system from Figure 34a), it was found that $\Delta\tilde{v}_{resolution} = 3.87$ cm^{-1}. Therefore, due to the difference in the focal length of the reference spectrometer, which is more than twice that of the prototype, the spectral resolution of the reference system is more than twice as high.

6.1.2. Signal to noise ratio

To measure the quality of a peak versus background noise obtained by the prototype, the signal-to-noise ratio (SNR) is estimated from experimental spectra and then compared with the reference system. Therefore, Si and S were used as samples for experimental spectra since they provide a strong Raman band at 520 cm^{-1} for Si and 472.22 cm^{-1} for S. Estimation was done by applying Savitzky-Golay (SG) smoothing [113–115] on raw data recorded by both systems setting polynomial to 3rd order and window frame to 11. Then, filtered spectra were subtracted from the experimental spectra to create a noise spectrum. The standard deviation was calculated from the noise spectrum, showing a background noise level. The SNR was roughly estimated (Equation 6) by relation between max peak intensity (S) and standard deviation of background noise (σ_b) [116].

$$SNR = \frac{S}{\sigma_b}$$

(6)

It has to be pointed out that reference system intensity is calculated as a mean value of the y-axis pixels for each x-axis pixel, while for the prototype, intensity is calculated as a sum of the y-axis pixels for each x-axis pixel. Therefore, to evaluate the SNR difference between the two systems, the prototype intensity was calculated the same way as the reference system. Filtered spectrum using SG smoothing and generated background noise is shown in Figure 37a,b for Si, and Figure 38a,b for S.
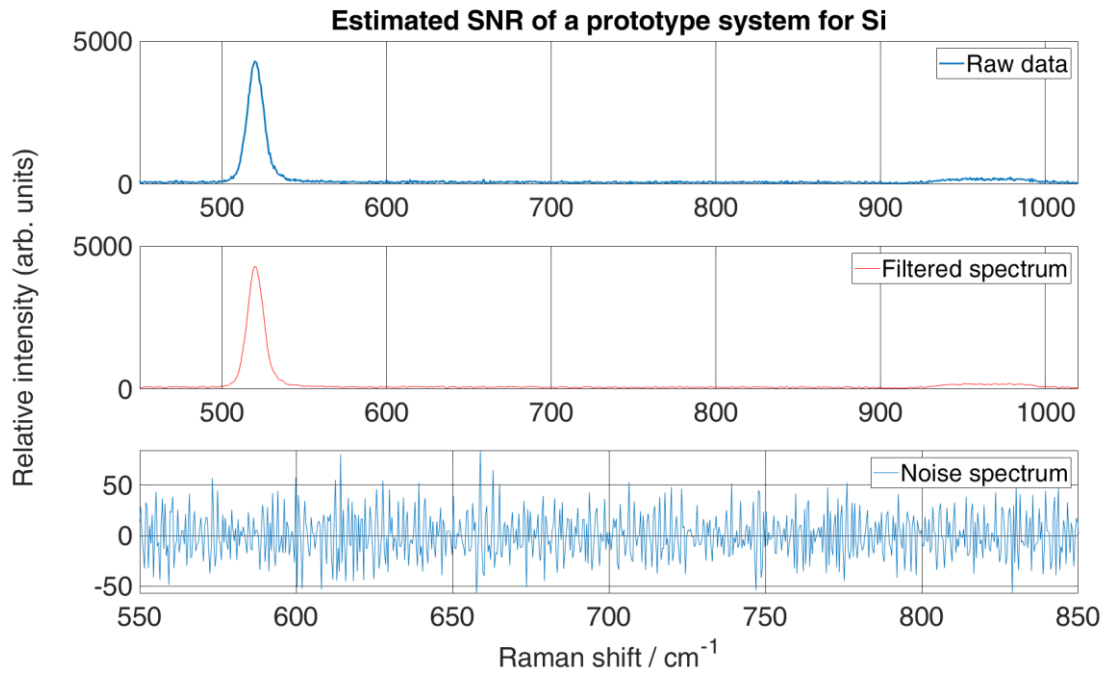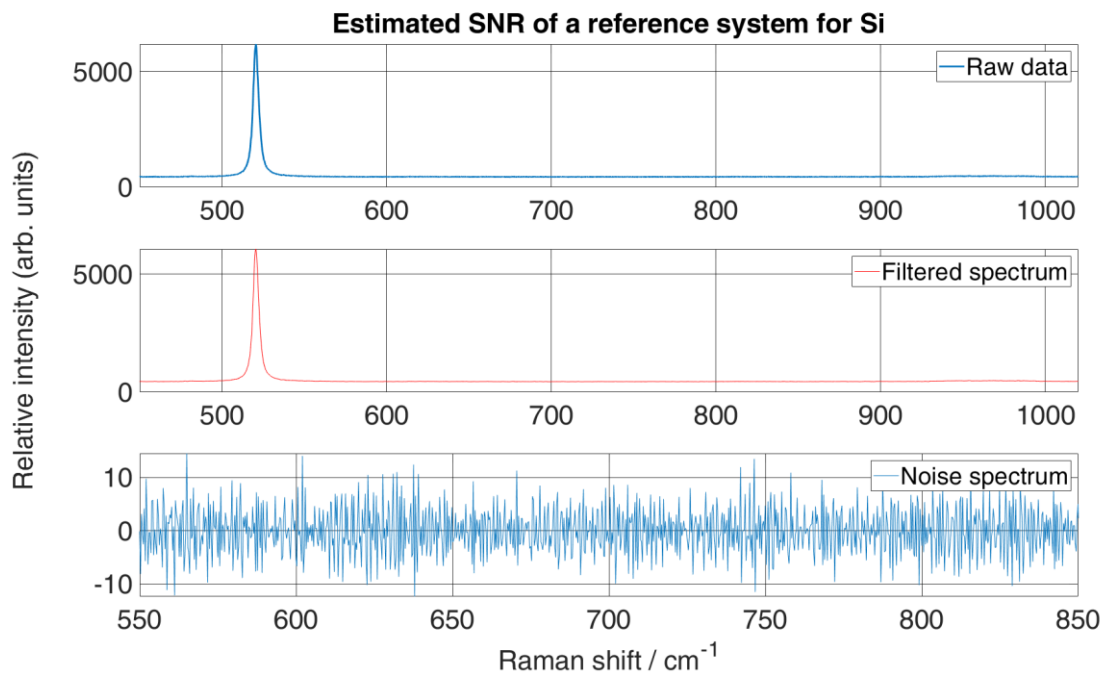
Figure 37. SNR estimation of obtained Si Raman spectra from prototype system a, and reference system b using SG smoothing.
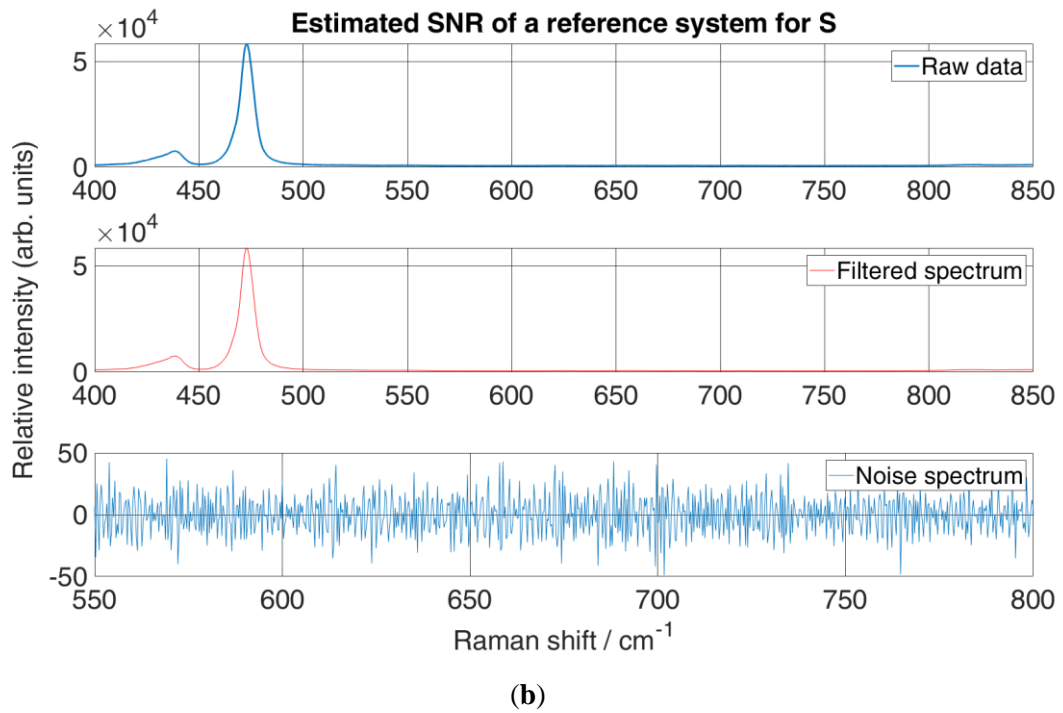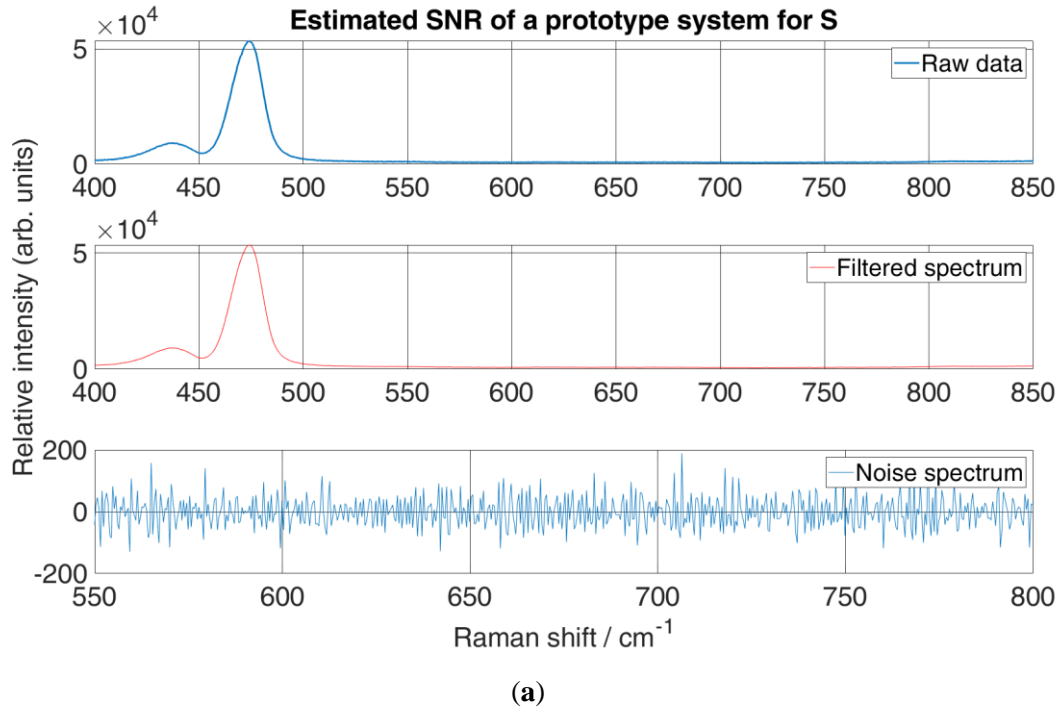
Figure 38. SNR estimation of obtained S Raman spectra from prototype system a, and reference system b using SG smoothing.

Estimated SNR (Table 12) shows that the reference system performs better regarding peak visibility over the background noise. Moreover, the background noise level is approximately 10 counts over the y-axis for Si, and 37 counts for S. However, a background noise level is higher for the prototype system, resulting in 22 counts over the y-axis for Si and 129.4 counts

for S. That can be due to the liquid-nitrogen-cooled CCD of a reference system, while CCD used for prototype was cooled only using its implemented Peltier cooling, reaching a maximum of -20°C. Therefore, as expected estimated SNR of a reference system is at least two times higher than the prototype regarding Si and approximately three times for S. Nevertheless, the background noise level of a prototype is consistent throughout spectra acquisition of different materials.

Table 12. Calculated parameters for SNR evaluation.

Material	System	Max peak [ar. units]	σ_b [arb. units]	SNR
Si	reference	6056.7	9.5831	632.0173
	prototype	4281.1	22.0753	193.9335
S	reference	58326	37.0754	1573.2
	prototype	106940	129.4070	826.3716

6.2. Photoluminescence spectra

The PL spectra of a Schott 550 and Schott 665 glass shown in Figure 39. were recorded using the proposed system and reference system using an excitation laser of 532 nm. The ZF of the proposed system was set to 4.5x so that the spectral interval of 116 nm could be recorded in a single shoot. The exposure time was set 2 s for both systems. However, the total acquisition time for the same spectral interval using the reference system was around five times longer because the signal of each spectral interval of 20 nm width (on average) was collected separately and then combined into one single spectrum. For Schott 550 (Figure 39a), the grating was rotated in such a way that the laser is on the far left side, and the visible spectral range for Schott 550 of 538 – 654 nm was captured. To reach the spectral interval for Schott 665, the grating was rotated so that the spectral interval from 570 – 686 nm can be recorded as can be seen in Figure 39b.

It can be seen from Figure 39 that both PL spectrum of a proposed system shows similar spectra as the PL spectra of a reference system. However, PL spectra is much more detailed for reference system, since each of the segment with wavelength width of 20 nm was recorded across full CCD (1024 pixels) resulting in more than 5000 data points. It has to be pointed out that PL spectra from 580 – 654 nm of the proposed system (Figure 39a) show descending spectra due to the procedure of a reference system for PL acquisition. This could be explained by the spectral stitching of small spectrums into one spectrum in the reference spectrum. However, these small sections tend to carry a variable background of the slope which change during the recording. Therefore, the PL spectrum recorded by the reference system can show

wrong information. Moreover, from Figure 38b it can be seen that the PL spectrum recorded by the prototype shows the same PL spectrum as the reference system, also this points out that reference system during this measurement stitched all spectrums correctly. Therefore, resulting in same PL spectrum as a PL spectrum of a prototype.
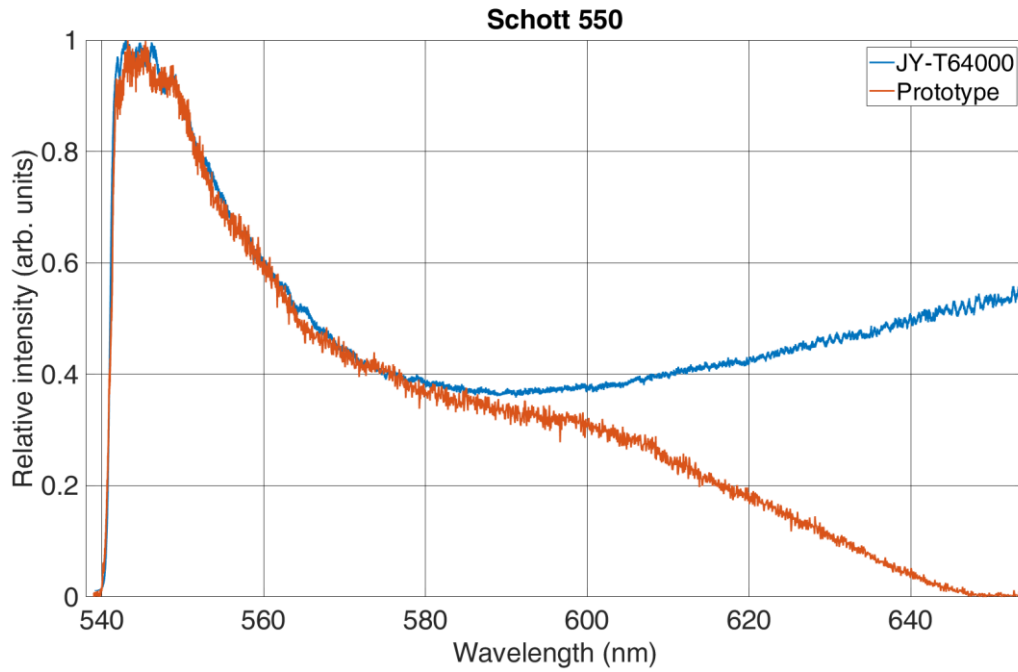
(**a**)
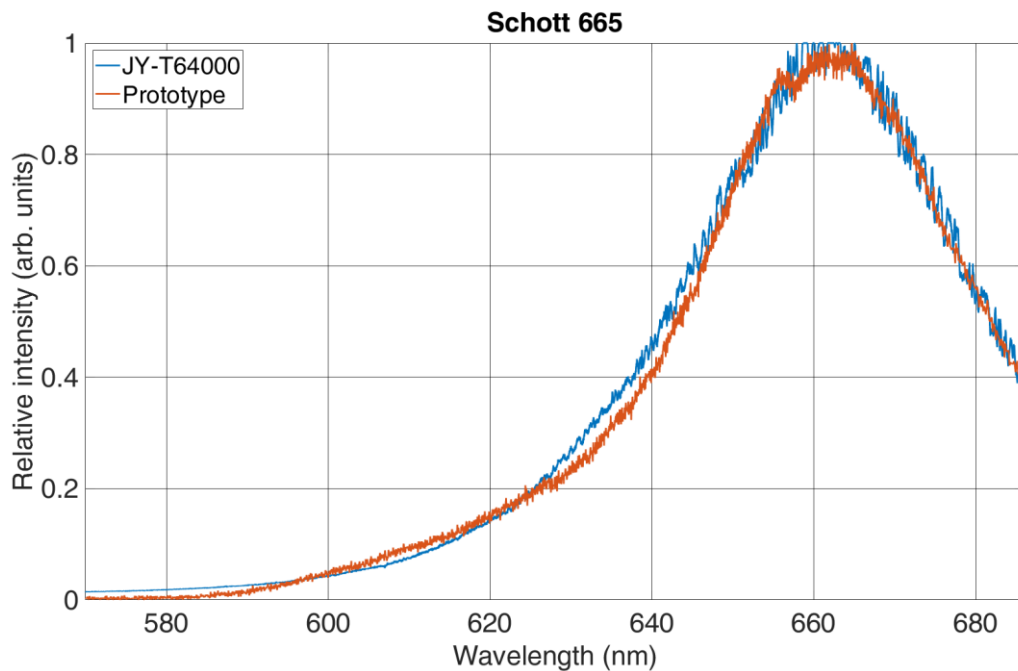
(**b**)

Figure 39. PL spectra of a Schott 550 (a), and Schott 665 (b) recorded with prototype and reference system.

7. INVENTION – PATENT APPLICATION

In this chapter, an invention with the patent application number P20211818A will be described. It has to be pointed out that the invention was applied for a patent at the State Intellectual Property Office of the Republic of Croatia and was done in collaboration with Institute Ruđer Bošković and Maritime Faculty – Split.

The present invention relates to a Raman spectrometer which allows a change in a spectral resolution that is suitable for recording Raman and photoluminescent spectra.

In order to obtain information about the vibrational states of an individual molecule, one of the most commonly used techniques for analysis is Raman spectroscopy, and devices developed for this purpose are called Raman spectrometers. On the other hand, in order to gain insight into the electronic states of the molecule, photoluminescent spectroscopy is used, for which there are appropriate instruments. When shooting the Raman effect, which is essentially very weak, there is a problem of backlighting, i.e., a strong photoluminescent background that needs to be eliminated. Since photoluminescent phenomena occur when recording the Raman effect, it can be concluded that the Raman spectrometer is also suitable for photoluminescent recordings. Today's Raman spectrometers are mainly intended for recording the Raman effect, but they also have the ability to record the photoluminescent spectrum. However, currently, during the acquisition of the photoluminescent spectrum, the time of acquisition is long, and there is a possibility of creating artifacts that are not part of the observed spectrum. This innovation brings the Raman spectrometer as a unique device for recording Raman and photoluminescent spectra. This design of the Raman spectrometer significantly reduces the acquisition time and undesirable artifacts during photoluminescent recordings and allows the selection of the desired spectrum recording areas that are suitable for the Raman effect. The presented innovation is the implementation of a zoom lens on the output part of the spectrometer. This innovation allows changing the spectral resolution during spectrum acquisition. An increased zoom factor allows you to record the desired range of the spectrum, which is suitable for recording the Raman effect and the full spectrum at a minimum zoom factor suitable for the photoluminescent spectrum.

The innovation consists of the mechanical connection of additional optics (zoom lens) to the optical detector of the Raman spectrometer. The zoom lens can be any gain factor with the possibility of infinite focusing and may or may not contain a built-in shutter function. It should

also be suitable for easy installation in such a way that its connection connectors are of the standardized type for today's optical detectors. Zoom, focus, and cover position control can be divided into two variants: a) manual. b) automatically.

It is an object of the presented invention to provide a Raman spectrometer capable of recording the Raman and photoluminescent spectra.

In accordance with the presented invention, a Raman spectrometer for recording the Raman and photoluminescent spectra is provided, comprising an excitation laser source, a Raman head or a microscope for supplying an excitation source to the observed material and scattering scattered light from the material, with collimation lens and diffraction grating, zoom lens and an optical detector.

The following figures show an embodiment of the presented invention:

Figure 40 is a schematic representation of Raman spectrometer components in accordance with the presented invention

Figure 41 is a schematic representation of the operation of a zoom lens in accordance with the presented invention

The method of using the invention is as follows: The laser source (1) produces a laser beam that passes through a band-pass filter (2) and is fed via a light guide (3) or otherwise to a Raman head (4) or microscope (4) on the observed sample (5). Feeding the beam on the sample (5) creates Raman and photoluminescent light which is collected by the Raman's head (4) or microscope (4) and led to the notch filter (6), then through the fiber (3) or otherwise reflected to the spectrometer (8) via the input collimation lens (7). In the spectrometer (8) on the diffraction grating (11), the scattered light is decomposed into frequency components and directed to the optical detector (10). On the path to the optical detector (10), light passes through the zoom lens (9), which with its magnification factor regulates the range of light in the physical space that falls on the optical detector (10). Since light is decomposed in physical space into frequency components, the zoom lens (9) thus actually regulates the frequency range of light incident on the optical detector (10). Furthermore, because the optical detector (10) consists of a certain finite number of pixels, changing the frequency range of the light falling on the optical detector (10) simultaneously regulates the resolution or level of detail of the spectrum. In other

words, if the zoom lens (9) is adjusted to cover the narrower part of the spectrum (zoom-in), better spectral resolution, i.e., more detail, will appear at the same time. The opposite is also true, by decreasing the zoom factor of the lens (9), the spectral resolution decreases, but the range of the spectrum falling on the optical detector (10) increases.

In this way, this innovation allows a single device to capture wide photoluminescence spectra in one mode (when the zoom lens (9) is at low magnification), and narrow and detailed Raman tapes in another mode (when the zoom lens (9) is at high magnification). In addition to the high flexibility in the work that innovation brings, the advantage of the invention is the ability to record wide ranges of frequencies or photoluminescent bands in a single window without the need for a so-called stitching method. In addition to being slow, the stitching method results in artifacts in the measurement due to the merging of multiple spectra into one. Furthermore, the advantage of the invention is that with wide spectrum ranges, the device quickly becomes a high-resolution spectrometer that can display very fine details of narrow Raman bands in such a way that the zoom lens (9) zooms. The upper limit of the spectral resolution of our invention is defined by the zoom lens magnification factor, which is a parameter that is very adaptable in the production of the invention, and subsequently easily changeable for various customer needs.

A zoom lens (9) can be any set of optical elements within a housing that allows for dynamic optical zooming or reduction of the image it transmits. It can consist of different convergent and divergent lenses at different distances and mutual arrangement. The magnification factor can be dynamically changed by rotating the inner or outer element, manually or via an electric motor or otherwise. The proposed invention's zoom lens (9) receives diffused light at its input to the spectral components, optically increases or decreases it depending on the magnification factor, and sends it to the optical detector (10) at the output.

In the case of the manual zoom lens variant (9), control of the zoom lens (9) is performed by manually rotating the zoom lens elements (9).

In the case of a motorized zoom lens variant (9), control of the zoom lens (9) is performed by starting the motor via the program interface. The software interface can be any interface that has the ability to collect data from the optical detector (10) and control the zoom lens control motors (9) and depends on the user's requirements. A typical sample measurement might look like this: The experimenter places the sample (5) of interest under the microscope (4) or in front of the Raman head (4), focuses the microscopic (4) lens on the position of interest on the sample (5), and illuminates it with a laser beam. (1) through a microscopic (4) lens. It then sets the

spectral range manually or via a programming interface to the widest mode. It triggers a quick acquisition and gets a rough but wide spectral window. He notices various peaks and bands in the recorded spectrum. He notices a wide photoluminescent tape, which was recorded correctly and completely without artifacts in this way. In addition, he notices narrow bands on the tail of the photoluminescent hill. It indicates a small part of the interest in the measured spectrum via a user-friendly software interface. Then the zoom lens magnification factor (9) is increased manually or motorized to the marked or selected spectral range. The experimenter starts the second measurement where a much narrower beam of light now falls in the same range on the optical detector (10), and therefore the collected spectrum has a much higher spectral resolution. The experimenter can now spot the tiny details of the narrow Raman ribbons.

The mode of operation of the zoom lens (9) in the present invention is as follows: Figure 2 shows the role of the zoom lens (9) in the Raman spectrometer. The beam of decomposed frequency components coming from the diffraction grating (11) comes to the zoom lens (9). The zoom lens (9) consists of a focusing lens (12), a zoom lens (14), and a cover (15). The focusing lens (12) is set to the infinite focusing position. The zoom lens position (14) is set to the lowest gain factor, and the cover (15) is set to the fully open position. The beam of decomposed frequency components of spectral width λ_1 coming from the spectrometer (8) passes to the zoom lens (9). Such a beam passes through the focusing lens (12) and passes through the zoom lens (14) to the optical detector (10). Then the focal length d_2 is smaller between the zoom lens (14) and the optical detector. The spectral interval of the beam λ_1 is larger and falls on the surface (17) of the optical detector (10). When moving the zoom lens (14) to the maximum magnification in position (13) then, a smaller part of the beam of decomposed frequency components of spectral width λ_2 enters through the focusing lens (12) on the zoom lens (14). Then the focal length d_1 is greater between the zoom lens (14) and the optical detector (10). The spectral beam interval λ_2 is narrower and falls on the surface (17) of the optical detector (10).

To record photoluminescence, it is necessary to include the largest possible spectral interval of the beam divided by the diffraction grating (11), this is achieved with the smallest possible zoom factor (14), or the smallest focal length between the zoom lens (14) and optical detector (10). To record the Raman effect and spectral range of spectral interest, i.e. to increase the spectral resolution, it is necessary to increase the zoom lens gain factor (14) to the appropriate gain, i.e., increase the focal length between the zoom lens (14) and optical detector (10).

List of marks

(1) Laser source, laser wavelength from 229 - 1064 nm, various technologies: gas (eg Ar, He: Cd, He: Ne), diode and semiconductor (eg Nd: YAG) and the like.

(2) Bandpass filter whose wavelength transmittance corresponds to the wavelength of the selected laser source (1)

(3) Fiber optic that can be the single or multimodal optical fiber of different core sizes

(4) Raman head or microscope capable of collecting Raman scattering and reflect it to the spectrometer (8)

(5) Measurement sample

(6) Notch filter which has the role of eliminating strong illumination of the laser source (1) from the output beam consisting of scattered Raman light of the sample (6) together with strong illumination of the laser source (1).

(7) The collimation input lens into the spectrometer (8), which can be part of the spectrometer or additionally installed, is an important function of creating a parallel beam toward the diffraction grating (11) inside the spectrometer (8).

(8) A spectrometer consisting of an input collimation lens (7) for generating a parallel beam within the spectrometer (8), a mirror guiding the parallel beam to the diffraction grating (11) and a mirror for directing the split beam of the diffraction grating (11) towards the spectrometer output (8).

(9) The zoom lens that contains the least focus lens (12) and zoom lens (14). It may or may not contain a cover (15), and control of all these elements can be automatic or manual. It is also necessary to have a standard type of connector to connect to the various optical detectors (10). The maximum gain factor can be different. It may also contain a barrel lens for added focal length enhancement.

(10) Optical detector of various technologies CCD, CMOS, EMCCD detector that can be adapted to various computer software solutions. High resolution, 16 - bit A / D conversion.

(11) Diffraction grating which can contain from 600 - 1200 grooves / mm.

(12) Focus lens zoom lens (9)

(13) Position at maximum zoom lens magnification factor (14).

(14) Zoom lens

(15) Lens cover

(16) Magnified display of recorded spectrum on CCD or CMOS detector (10) at minimum and maximum zoom lens magnification factor (14). For the smallest magnification factor the recorded wider spectral interval λ_1 falls on the whole surface (17) of the CCD or CMOS detector (10), while for the largest magnification, the recorded narrower spectral interval λ_2 falls on the whole surface (17) OR CMOS detector (10).

(17) The area of a CCD or CMOS detector (10) depends on the number and pixel size of the CCD or CMOS detector (10) itself.

Patent claims

1. Optoelectronic device for recording Raman and photoluminescent spectra with a variable spectral resolution, comprising:

a laser light source (1) with a corresponding bandpass filter (2);

a microscope (4) or Raman heads (4);

a notch filter (6), and a spectrometer (8) containing a minimum of an input collimation lens (7) and a diffraction grating (11) with 600 - 1200 grooves / mm;

and by which the output collimation lens of the spectrometer (8) is **thus** replaced by a zoom lens (9).

An optoelectronic device, according to claim 1**, characterized in** that the zoom lens (9) minimally comprises a focusing lens (12) and a zoom lens (14).

An optoelectronic device, according to claims 1 and 2, **characterized in** that the control of the zoom lens (9) can be manual or automatic

An optoelectronic device, according to any one of the preceding claims, **characterized in** that said device is useful for recording Raman and photoluminescent spectra with different zoom lens magnification factors (14)

An optoelectronic device, according to any one of the preceding claims, **characterized in that it** is adaptable to standard computer interfaces and software solutions for displaying data obtained from CCD or CMOS detectors (10).
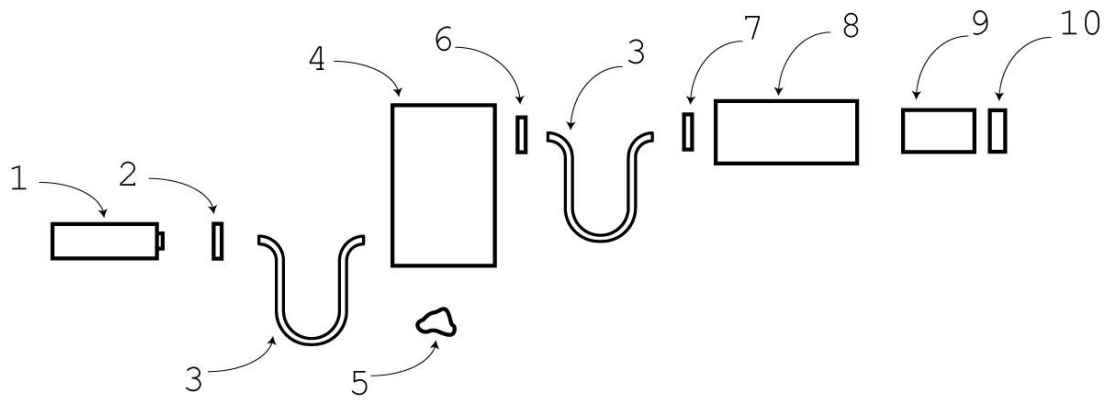
Figure 40. Schematic representation of Raman spectrometer components in accordance with the presented invention.
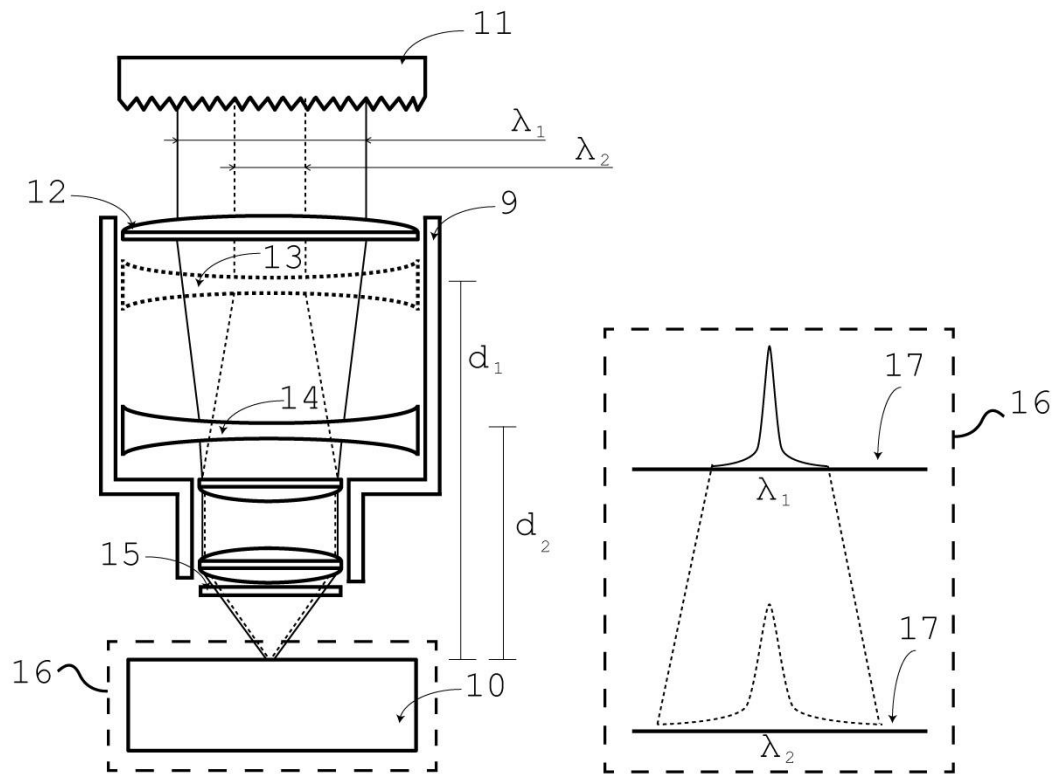
Figure 41. Schematic representation of the operation of a zoom lens in accordance with the presented invention.

8. CONCLUSION

The maritime industry is the key factor in terms of trade and industry. It includes all activities related to maritime transport, including shipping, shipbuilding, marine engineering, marine infrastructure, etc. Therefore, the maritime industry is directly associated with the marine environment. However, in order to reduce maintenance caused by the marine environment and damage caused to a marine environment, it is necessary to comply with the regulations addressed by the International Maritime Organisation (IMO). Nowadays, this is achieved by analysing existing problems (corrosion, microplastics, emissions, etc.). One of the techniques used for the analysis is Raman and Photoluminescence (PL) spectroscopy.

Therefore, a novel Raman and PL spectroscopy approach based on a variable spectral resolution is described. The variable spectral resolution was achieved in Raman and PL spectra by using zoom optics in the collimating part of the monochromator. It was shown that the collimating optics' different zoom factors directly affect the spectral resolution of the observed spectral interval. With the minimum ZF, the lowest spectral resolution with the largest spectral interval was achieved on the detector. On the contrary, with the maximum ZF, the largest spectral resolution with the smallest spectral interval was demonstrated on the detector. This approach allows the acquisition of Raman and PL spectra with significantly reduced time. Also, the ability to vary the spectral range and resolution offers great flexibility in terms of applications and conditions of usage of such device, ranging from quick Raman measurements for detection or confirmation of substances, broad spectrum PL measurements, to high-resolution measurements for detailed chemical and structural analysis.

First, a macro-Raman spectrometer prototype was built with a maximum 6x zoom lens and a low-resolution CCD. To evaluate the performance of the first prototype, a comparison was made with a high-grade laboratory Raman spectrometer (a reference spectrometer). The prototype approach outperforms the reference spectrometer in the PL measurement, in acquisition time as well as in avoiding the spectral artifacts produced by combining more spectral intervals into a single spectrum. Contrary to the spectral merging of the reference spectrometer, the variable spectral resolution spectrometer can be easily adjusted to collect the whole PL spectral interval at once.

Next, a second prototype was built for micro-Raman spectroscopy using a zoom lens with a maximum 18x magnification and a high-resolution CCD. It has to be pointed out that the

second prototype was automated at a certain level, enabling the rotation of grating, positioning the zoom lens, and shutter control. Therefore, custom electronic hardware was built together with custom software for spectra acquisition. To evaluate performance of a second prototype, the comparison was made on the same reference spectrometer. Also, evaluation of obtained spectrums was done using statistical analysis of variance (ANOVA). Results indicate no significant difference between the prototype and the reference spectrometer. Furthermore, the signal-to-noise (SNR) ratio of a second prototype was evaluated, indicating at least two times higher background noise in comparison with the reference spectrometer. However, this is due to the liquid-nitrogen-cooled CCD used in the reference spectrometer.

Finally, it is shown that the Raman spectrometer with the zoom lens changes the spectral resolution and significantly reduces the acquisition time for Raman and PL spectra. However, there are already a couple of commercial spectrometers holding dual-mode acquisition for both Raman and PL. Simply rotating the grating to reach blazing angles, a broad spectrum is obtained quickly. The advantage of the proposed system is not only to have two gratings and two fixed sizes of the spectrum on the detector but to have a variable spectrum size at the detector with a single grating. This approach enables the recording of Raman and PL spectra within a single spectrometer. Moreover, results show that such a spectrometer could be an efficient tool for inspecting Raman and PL bands of unknown materials and, thereafter, recording wanted spectral intervals at appropriate spectral resolution.

In addition, the developed prototype requires special software, i.e., a graphical user interface (GUI) that records and displays the spectra of the observed material. It is well known that GUI must implement functions to control hardware components (grating, shutter, and zoom) and functions to acquire pixel information from the detector. Therefore, the developed GUI is designed only for the developed prototype, as all functions are intended for the components used in the prototype.

Potential limitations of this prototype in terms of software are the lack of a multi-platform GUI and flexibility in operation with different CCD/CMOS, motors, and electronic boards. Since the developed GUI can only record and display spectra, it can be enhanced with additional algorithms that improve data analysis. Therefore, algorithms such as baseline correction, Gaussian fit, smoothing, etc. can be implemented. In addition, a database of acquired spectra can also be implemented so that the acquired spectra can be automatically classified. Potential limitations of this prototype in terms of hardware are the overall dimensions, the

influence of the surrounding light, manual adjustment of the optical path, optical mirrors, and few excitation sources. To reduce the dimensions, the monochromator can be replaced by a grating attached directly to the output of the microscope. Surrounding light can be eliminated by using optical fibers as the optical path for the scattered light. Finally, the adjustment of the optical components can also be automated by using appropriate electronic and mechanical components.

The possibilities for further development of this approach are numerous. For example, one way is to increase the spectral resolution by using a zoom lens with higher ZF, using a longer focus at the highest ZF, and involving a higher pixel number CCD/CMOS detector.

On the software side, there is room for improvement in the ability to vary the resolution, data acquisition, and similar elements.

Finally, it has to be noted that the proposed prototype and this novel approach to spectroscopy will also serve as a foundation for future research and open up a gateway for future innovative developments in the field.

9. LITERATURE

1. Fugazza, M.; Hoffmann, J. Liner Shipping Connectivity as Determinant of Trade. *J. Shipp. Trade* **2017**, *2*, 1, doi:10.1186/s41072-017-0019-5.

2. Li, Q.; Li, Y.; Li, H.; Yan, X.; Han, G.; Chen, F.; Song, Z.; Zhang, J.; Fan, W.; Yi, C.; et al. Highly Luminescent Copper Nanoclusters Stabilized by Ascorbic Acid for the Quantitative Detection of 4-Aminoazobenzene. *Nanomaterials* **2020**, *10*, 1531, doi:10.3390/nano10081531.

3. Matuszewska, A. The Use of Synchronous Luminescence Spectroscopy in Qualitative Analysis of Aromatic Fraction of Hard Coal Thermolysis Products. *Talanta* **2000**, *52*, 457–464, doi:10.1016/S0039-9140(00)00369-6.

4. Bakkar, M.A.; Nawaz, H.; Majeed, M.I.; Naseem, A.; Ditta, A.; Rashid, N.; Ali, S.; Bajwa, J.; Bashir, S.; Ahmad, S.; et al. Raman Spectroscopy for the Qualitative and Quantitative Analysis of Solid Dosage Forms of Sitagliptin. *Spectrochim. Acta Part A Mol. Biomol. Spectrosc.* **2021**, *245*, 118900, doi:10.1016/j.saa.2020.118900.

5. RAMAN, C. V.; KRISHNAN, K.S. A New Type of Secondary Radiation. *Nature* **1928**, *121*, 501–502, doi:10.1038/121501c0.

6. Society, R. The 1930 Nobel Prize for Physics : A Close Decision ? Author (s): Rajinder Singh and Falk Riess Source : Notes and Records of the Royal Society of London , Vol . 55 , No . 2 (May , 2001), Pp . Published by : Royal Society Stable URL : Https://Www.Jsto. **2019**, *55*, 267–283.

7. Polubotko, A.M.; Chelibanov, V.P. The Theory of Surface Enhanced Hyper Raman Scattering (A Review). *Opt. Spectrosc.* **2016**, *120*, 86–108, doi:10.1134/S0030400X1512019X.

8. Madzharova, F.; Heiner, Z.; Kneipp, J. Surface-Enhanced Hyper Raman Spectra of Aromatic Thiols on Gold and Silver Nanoparticles. *J. Phys. Chem. C* **2020**, *124*, 6233–6241, doi:10.1021/acs.jpcc.0c00294.

9. Verma, P. Tip-Enhanced Raman Spectroscopy: Technique and Recent Advances. *Chem. Rev.* **2017**, *117*, 6447–6466, doi:10.1021/acs.chemrev.6b00821.

10. Shao, F.; Zenobi, R. Tip-Enhanced Raman Spectroscopy: Principles, Practice, and Applications to Nanospectroscopic Imaging of 2D Materials. *Anal. Bioanal. Chem.* **2019**, *411*, 37–61, doi:10.1007/s00216-018-1392-0.

11. Begley, R.F.; Harvey, A.B.; Byer, R.L. Coherent Anti-Stokes Raman Spectroscopy. *Appl. Phys. Lett.* **1974**, *25*, 387–390, doi:10.1063/1.1655519.

12. Prince, R.C.; Frontiera, R.R.; Potma, E.O. Stimulated Raman Scattering: From Bulk to Nano. *Chem. Rev.* **2017**, *117*, 5070–5094, doi:10.1021/acs.chemrev.6b00545.

13. Samek, O.; Bernatová, S.; Dohnal, F. The Potential of SERS as an AST Methodology in Clinical Settings. *Nanophotonics* **2021**, *10*, 2537–2561, doi:10.1515/nanoph-2021-0095.

14. Foti, A.; D'Andrea, C.; Villari, V.; Micali, N.; Donato, M.; Fazio, B.; Maragò, O.; Gillibert, R.; Lamy de la Chapelle, M.; Gucciardi, P. Optical Aggregation of Gold Nanoparticles for SERS Detection of Proteins and Toxins in Liquid Environment: Towards Ultrasensitive and Selective Detection. *Materials (Basel).* **2018**, *11*, 440, doi:10.3390/ma11030440.

15. Guilbault, G.G. III. TYPES OF LUMINESCENCE. In *Practical Fluorescence*; Marcel Dekker, INC., 1990; pp. 1–825 ISBN 0-8247-8350-6.

16. Bebb, H.B.; Williams, E.W. Chapter 4 Photoluminescence I: Theory. In *Semiconductors and*

Semimetals Volume 8; 1972; pp. 181–320 ISBN 978-0-12-752108-4.

17. Cadusch, P.J.; Hlaing, M.M.; Wade, S.A.; McArthur, S.L.; Stoddart, P.R. Improved Methods for Fluorescence Background Subtraction from Raman Spectra. *J. Raman Spectrosc.* **2013**, *44*, 1587–1595, doi:10.1002/jrs.4371.

18. Kostamovaara, J.; Tenhunen, J.; Kögler, M.; Nissinen, I.; Nissinen, J.; Keränen, P. Fluorescence Suppression in Raman Spectroscopy Using a Time-Gated CMOS SPAD. *Opt. Express* **2013**, *21*, 31632, doi:10.1364/OE.21.031632.

19. Chiuri, A.; Angelini, F. Fast Gating for Raman Spectroscopy. *Sensors* **2021**, *21*, 2579, doi:10.3390/s21082579.

20. Crawford, S.E.; Ohodnicki, P.R.; Baltrus, J.P. Materials for the Photoluminescent Sensing of Rare Earth Elements: Challenges and Opportunities. *J. Mater. Chem. C* **2020**, *8*, 7975–8006, doi:10.1039/D0TC01939A.

21. Cai, C.; Nishimura, T.; Hwang, J.; Hu, X.-M.; Kuroda, A. Asbestos Detection with Fluorescence Microscopy Images and Deep Learning. *Sensors* **2021**, *21*, 4582, doi:10.3390/s21134582.

22. Eaton-Magaña, S.; Breeding, C.M.; Palke, A.C.; Homkrajae, A.; Sun, Z.; McElhenny, G. Raman and Photoluminescence Mapping of Gem Materials. *Minerals* **2021**, *11*, 177, doi:10.3390/min11020177.

23. Alshehawy, A.M.; Mansour, D.-E.A.; Ghali, M.; Lehtonen, M.; Darwish, M.M.F. Photoluminescence Spectroscopy Measurements for Effective Condition Assessment of Transformer Insulating Oil. *Processes* **2021**, *9*, 732, doi:10.3390/pr9050732.

24. Huang, C.-C. Applications of Raman Spectroscopy in Herbal Medicine. *Appl. Spectrosc. Rev.* **2016**, *51*, 1–11, doi:10.1080/05704928.2015.1092154.

25. Bernatová, S.; Rebrošová, K.; Pilát, Z.; Šerý, M.; Gjevik, A.; Samek, O.; Ježek, J.; Šiler, M.; Kizovský, M.; Klementová, T.; et al. Rapid Detection of Antibiotic Sensitivity of Staphylococcus Aureus by Raman Tweezers. *Eur. Phys. J. Plus* **2021**, *136*, 233, doi:10.1140/epjp/s13360-021-01152-1.

26. Kong, K.; Kendall, C.; Stone, N.; Notingher, I. Raman Spectroscopy for Medical Diagnostics — From in-Vitro Biofluid Assays to in-Vivo Cancer Detection. *Adv. Drug Deliv. Rev.* **2015**, *89*, 121–134, doi:10.1016/j.addr.2015.03.009.

27. Matteini, P.; Cottat, M.; Tavanti, F.; Panfilova, E.; Scuderi, M.; Nicotra, G.; Menziani, M.C.; Khlebtsov, N.; de Angelis, M.; Pini, R. Site-Selective Surface-Enhanced Raman Detection of Proteins. *ACS Nano* **2017**, *11*, 918–926, doi:10.1021/acsnano.6b07523.

28. Krafft, C.; Popp, J. The Many Facets of Raman Spectroscopy for Biomedical Analysis. *Anal. Bioanal. Chem.* **2015**, *407*, 699–717, doi:10.1007/s00216-014-8311-9.

29. Izake, E.L. Forensic and Homeland Security Applications of Modern Portable Raman Spectroscopy. *Forensic Sci. Int.* **2010**, *202*, 1–8, doi:10.1016/j.forsciint.2010.03.020.

30. Ong, T.T.X.; Blanch, E.W.; Jones, O.A.H. Surface Enhanced Raman Spectroscopy in Environmental Analysis, Monitoring and Assessment. *Sci. Total Environ.* **2020**, *720*, 137601, doi:10.1016/j.scitotenv.2020.137601.

31. Li, Y.; Igne, B.; Drennen, J.K.; Anderson, C.A. Method Development and Validation for Pharmaceutical Tablets Analysis Using Transmission Raman Spectroscopy. *Int. J. Pharm.* **2016**, *498*, 318–325, doi:10.1016/j.ijpharm.2015.11.049.

32. Somekawa, T.; Fujita, M.; Izawa, Y.; Kasaoka, M.; Nagano, Y. Furfural Analysis in Transformer Oils Using Laser Raman Spectroscopy. *IEEE Trans. Dielectr. Electr. Insul.* **2015**, *22*, 229–231, doi:10.1109/TDEI.2014.004800.

33. Olson, D.H.; Deijkers, J.A.; Quiambao-Tomko, K.; Gaskins, J.T.; Richards, B.T.; Opila, E.J.; Hopkins, P.E.; Wadley, H.N.G. Evolution of Microstructure and Thermal Conductivity of Multifunctional Environmental Barrier Coating Systems. *Mater. Today Phys.* **2021**, *17*, 100304, doi:10.1016/j.mtphys.2020.100304.

34. De Coster, D.; Loterie, D.; Ottevaere, H.; Vervaeke, M.; Van Erps, J.; Missinne, J.; Thienpont, H. Free-Form Optics Enhanced Confocal Raman Spectroscopy for Optofluidic Lab-on-Chips. *IEEE J. Sel. Top. Quantum Electron.* **2015**, *21*, 79–86, doi:10.1109/JSTQE.2014.2381470.

35. Yu, H.; Tan, Y.; Cunningham, B.T. Smartphone Fluorescence Spectroscopy. *Anal. Chem.* **2014**, *86*, 8805–8813, doi:10.1021/ac502080t.

36. Somerville, W.R.C.; Le Ru, E.C.; Northcote, P.T.; Etchegoin, P.G. High Performance Raman Spectroscopy with Simple Optical Components. *Am. J. Phys.* **2010**, *78*, 671–677, doi:10.1119/1.3427413.

37. Emmanuel, N.; Nair, R.B.; Abraham, B.; Yoosaf, K. Fabricating a Low-Cost Raman Spectrometer to Introduce Students to Spectroscopy Basics and Applied Instrument Design. *J. Chem. Educ.* **2021**, *98*, 2109–2116, doi:10.1021/acs.jchemed.0c01028.

38. Mohr, C.; Spencer, C.L.; Hippler, M. Inexpensive Raman Spectrometer for Undergraduate and Graduate Experiments and Research. *J. Chem. Educ.* **2010**, *87*, 326–330, doi:10.1021/ed800081t.

39. R., E.H.M.; L., A.A.; E., O.R.B. A Homemade Cost Effective Raman Spectrometer with High Performance. *J. Lab. Chem. Educ.* **2015**, *3*, 67–75, doi:10.5923/j.jlce.20150304.02.

40. Youngworth, R.N.; Betensky, E.I. Fundamental Considerations for Zoom Lens Design.; Betensky, E.I., Yamanashi, T., Eds.; October 19 2012; p. 848806.

41. Jürgen Popp and Thomas Mayerhöfer Micro-Raman Spectroscopy. In; Popp, J., Mayerhöfer, T., Eds.; Walter de Gruyter GmbH, Berlin/Boston, 2020; pp. 41–42 ISBN 978-3-11-051479-7.

42. Pavić, I.; Šoda, J.; Gašparić, V.; Ivanda, M. Raman and Photoluminescence Spectroscopy with a Variable Spectral Resolution. *Sensors* **2021**, *21*, 7951, doi:10.3390/s21237951.

43. Kudelski, A. Analytical Applications of Raman Spectroscopy. *Talanta* **2008**, *76*, 1–8, doi:10.1016/j.talanta.2008.02.042.

44. Das, R.S.; Agrawal, Y.K. Raman Spectroscopy: Recent Advancements, Techniques and Applications. *Vib. Spectrosc.* **2011**, *57*, 163–176, doi:10.1016/j.vibspec.2011.08.003.

45. Day, J.S.; Edwards, H.G.M.; Dobrowski, S.A.; Voice, A.M. The Detection of Drugs of Abuse in Fingerprints Using Raman Spectroscopy I: Latent Fingerprints. *Spectrochim. Acta Part A Mol. Biomol. Spectrosc.* **2004**, *60*, 563–568, doi:10.1016/S1386-1425(03)00263-4.

46. Keresztury, G. Raman Spectroscopy: Theory. In *Handbook of Vibrational Spectroscopy*; Chalmers, J.M., Ed.; John Wiley & Sons, Ltd: Chichester, UK, 2006.

47. Movasaghi, Z.; Rehman, S.; Rehman, I.U. Raman Spectroscopy of Biological Tissues. *Appl. Spectrosc. Rev.* **2007**, *42*, 493–541, doi:10.1080/05704920701551530.

48. Naumann, D. FT-INFRARED AND FT-RAMAN SPECTROSCOPY IN BIOMEDICAL RESEARCH. *Appl. Spectrosc. Rev.* **2001**, *36*, 239–298, doi:10.1081/ASR-100106157.

49. Kong, K.; Kendall, C.; Stone, N.; Notingher, I. Raman Spectroscopy for Medical Diagnostics — From in-Vitro Biofluid Assays to in-Vivo Cancer Detection. *Adv. Drug Deliv. Rev.* **2015**, *89*, 121–134, doi:10.1016/j.addr.2015.03.009.

50. Diem, M. *Modern Vibrational Spectroscopy and Micro-Spectroscopy*; John Wiley & Sons, Ltd: Chichester, UK, 2015; ISBN 9781118824924.

51. Sluder, G.; Nordberg, J.J. *Microscope Basics*; 4th ed.; Elsevier Inc., 2013; Vol. 114; ISBN 9780124077614.

52. Naeem, M.; Fatima, N.; Hussain, M.; Imran, T.; Bhatti, A.S. Design Simulation of Czerny–Turner Configuration-Based Raman Spectrometer Using Physical Optics Propagation Algorithm. *Optics* **2022**, *3*, 1–7, doi:10.3390/opt3010001.

53. Mamouri, R.-E.; Ansmann, A. Potential of Polarization/Raman Lidar to Separate Fine Dust, Coarse Dust, Maritime, and Anthropogenic Aerosol Profiles. *Atmos. Meas. Tech.* **2017**, *10*, 3403–3427, doi:10.5194/amt-10-3403-2017.

54. Yacobi, B.G.; Holt, D.B. Luminescence Phenomena. In *Cathodoluminescence Microscopy of Inorganic Solids*; Springer US: Boston, MA, 1990; pp. 21–54.

55. Lakowicz, J.R. *Principles of Fluorescence Spectroscopy, 3rd Principles of Fluorescence Spectroscopy, Springer, New York, USA, 3rd Edn, 2006.*; 2006; ISBN 0387312781.

56. Tebyetekerwa, M.; Zhang, J.; Xu, Z.; Truong, T.N.; Yin, Z.; Lu, Y.; Ramakrishna, S.; Macdonald, D.; Nguyen, H.T. Mechanisms and Applications of Steady-State Photoluminescence Spectroscopy in Two-Dimensional Transition-Metal Dichalcogenides. *ACS Nano* **2020**, *14*, 14579–14604, doi:10.1021/acsnano.0c08668.

57. Zettler, E.R.; Mincer, T.J.; Amaral-Zettler, L.A. Life in the "Plastisphere": Microbial Communities on Plastic Marine Debris. *Environ. Sci. Technol.* **2013**, *47*, 7137–7146, doi:10.1021/es401288x.

58. Lenz, R.; Enders, K.; Stedmon, C.A.; MacKenzie, D.M.A.; Nielsen, T.G. A Critical Assessment of Visual Identification of Marine Microplastic Using Raman Spectroscopy for Analysis Improvement. *Mar. Pollut. Bull.* **2015**, *100*, 82–91, doi:10.1016/j.marpolbul.2015.09.026.

59. Gomiero, A.; Strafella, P.; Fabi, G. From Macroplastic to Microplastic Litter: Occurrence, Composition, Source Identification and Interaction with Aquatic Organisms. Experiences from the Adriatic Sea. In *Plastics in the Environment*; IntechOpen, 2019.

60. Zhao, S.; Zhu, L.; Li, D. Microplastic in Three Urban Estuaries, China. *Environ. Pollut.* **2015**, *206*, 597–604, doi:10.1016/j.envpol.2015.08.027.

61. Naidu, S.A.; Ranga Rao, V.; Ramu, K. Microplastics in the Benthic Invertebrates from the Coastal Waters of Kochi, Southeastern Arabian Sea. *Environ. Geochem. Health* **2017**, *40*, 1377–1383, doi:10.1007/s10653-017-0062-z.

62. Choy, C.A.; Robison, B.H.; Gagne, T.O.; Erwin, B.; Firl, E.; Halden, R.U.; Hamilton, J.A.; Katija, K.; Lisin, S.E.; Rolsky, C.; et al. The Vertical Distribution and Biological Transport of Marine Microplastics across the Epipelagic and Mesopelagic Water Column. *Sci. Rep.* **2019**, *9*, doi:10.1038/s41598-019-44117-2.

63. Peng, X.; Chen, M.; Chen, S.; Dasgupta, S.; Xu, H.; Ta, K.; Du, M.; Li, J.; Guo, Z.; Bai, S. Microplastics Contaminate the Deepest Part of the World's Ocean. *Geochemical Perspect. Lett.* **2018**, *9*, 1–5, doi:10.7185/geochemlet.1829.

64. Araujo, C.F.; Nolasco, M.M.; Ribeiro, A.M.P.; Ribeiro-Claro, P.J.A. Identification of

Microplastics Using Raman Spectroscopy: Latest Developments and Future Prospects. *Water Res.* **2018**, *142*, 426–440, doi:10.1016/j.watres.2018.05.060.

65. Saeed, T.; Al-Jandal, N.; Al-Mutairi, A.; Taqi, H. Microplastics in Kuwait Marine Environment: Results of First Survey. *Mar. Pollut. Bull.* **2020**, *152*, 110880, doi:10.1016/j.marpolbul.2019.110880.

66. Pan, Z.; Guo, H.; Chen, H.; Wang, S.; Sun, X.; Zou, Q.; Zhang, Y.; Lin, H.; Cai, S.; Huang, J. Microplastics in the Northwestern Pacific: Abundance, Distribution, and Characteristics. *Sci. Total Environ.* **2019**, *650*, 1913–1922, doi:10.1016/j.scitotenv.2018.09.244.

67. Wang, S.; Chen, H.; Zhou, X.; Tian, Y.; Lin, C.; Wang, W.; Zhou, K.; Zhang, Y.; Lin, H. Microplastic Abundance, Distribution and Composition in the Mid-West Pacific Ocean. *Environ. Pollut.* **2020**, *264*, 114125, doi:10.1016/j.envpol.2020.114125.

68. Fytianos, G.; Ioannidou, E.; Thysiadou, A.; Mitropoulos, A.C.; Kyzas, G.Z. Microplastics in Mediterranean Coastal Countries: A Recent Overview. *J. Mar. Sci. Eng.* **2021**, *9*, 1–13, doi:10.3390/jmse9010098.

69. Li, Q.; Sun, X. Progress on Microplastics Research in the Yellow Sea, China. *Anthr. Coasts* **2020**, *3*, 43–52, doi:10.1139/anc-2018-0033.

70. Clunies-Ross, P.J.; Smith, G.P.S.; Gordon, K.C.; Gaw, S. Synthetic Shorelines in New Zealand? Quantification and Characterisation of Microplastic Pollution on Canterbury's Coastlines. *New Zeal. J. Mar. Freshw. Res.* **2016**, *50*, 317–325, doi:10.1080/00288330.2015.1132747.

71. Esiukova, E.; Zobkov, M.; Chubarenko, I. Data on Microplastic Contamination of the Baltic Sea Bottom Sediment Samples in 2015–2016. *Data Br.* **2020**, *28*, 104887, doi:10.1016/j.dib.2019.104887.

72. Mehdinia, A.; Dehbandi, R.; Hamzehpour, A.; Rahnama, R. Identification of Microplastics in the Sediments of Southern Coasts of the Caspian Sea, North of Iran. *Environ. Pollut.* **2020**, *258*, 113738, doi:10.1016/j.envpol.2019.113738.

73. Chen, M.; Du, M.; Jin, A.; Chen, S.; Dasgupta, S.; Li, J.; Xu, H.; Ta, K.; Peng, X. Forty-Year Pollution History of Microplastics in the Largest Marginal Sea of the Western Pacific. *Geochemical Perspect. Lett.* **2020**, *13*, 42–47, doi:10.7185/geochemlet.2012.

74. Wiener, M.S.; Salas, B.V. Corrosion of the Marine Infrastructure in Polluted Seaports. *Corros. Eng. Sci. Technol.* **2005**, *40*, 137–142, doi:10.1179/174327805X46931.

75. Sun, F.; Li, X.; Zhang, F.; Cheng, X.; Zhou, C.; Wu, N.; Yin, Y.; Zhao, J. Corrosion Mechanism of Corrosion-Resistant Steel Developed for Bottom Plate of Cargo Oil Tanks. *Acta Metall. Sin. (English Lett.* **2013**, *26*, 257–264, doi:10.1007/s40195-012-0231-0.

76. Li, H.; Chai, F.; Yang, C. fu; Li, C.; Luo, X. bing Corrosion Behavior of Low Alloy Steel for Cargo Oil Tank under Upper Deck Conditions. *J. Iron Steel Res. Int.* **2018**, *25*, 120–130, doi:10.1007/s42243-017-0004-0.

77. Morcillo, M.; Alcantara, J.; Diaz, I.; Chico, B.; Simancas, J.; De La Fuente, D. Marine Atmospheric Corrosion of Carbon Steels. *Rev. Metal.* **2015**, *51*, doi:10.3989/revmetalm.045.

78. James, M.N.; Hattingh, D.G. Case Studies in Marine Concentrated Corrosion. *Eng. Fail. Anal.* **2015**, *47*, 1–15, doi:10.1016/j.engfailanal.2014.08.013.

79. de la Fuente, D.; Alcántara, J.; Chico, B.; Díaz, I.; Jiménez, J.A.; Morcillo, M. Characterisation of Rust Surfaces Formed on Mild Steel Exposed to Marine Atmospheres Using XRD and SEM/Micro-Raman Techniques. *Corros. Sci.* **2016**, *110*, 253–264,

doi:10.1016/j.corsci.2016.04.034.

80. Husain, E.; Abdel Nazeer, A.; Alsarraf, J.; Al-Awadi, K.; Murad, M.; Al-Naqi, A.; Shekeban, A. Corrosion Behavior of AISI 316 Stainless Steel Coated with Modified Fluoropolymer in Marine Condition. *J. Coatings Technol. Res.* **2018**, *15*, 945–955, doi:10.1007/s11998-017-0025-4.

81. Jones, G. The battle against marine biofouling: a historical review. In *Advances in Marine Antifouling Coatings and Technologies*; Elsevier, 2009; pp. 19–45.

82. Oliveira, D.; Larsson, A.I.; Granhag, L. Effect of Ship Hull Form on the Resistance Penalty from Biofouling. *Biofouling* **2018**, *34*, 262–272, doi:10.1080/08927014.2018.1434157.

83. Inbakandan, D.; Kumar, C.; Abraham, L.S.; Kirubagaran, R.; Venkatesan, R.; Khan, S.A. Silver Nanoparticles with Anti Microfouling Effect: A Study against Marine Biofilm Forming Bacteria. *Colloids Surfaces B Biointerfaces* **2013**, *111*, 636–643, doi:10.1016/j.colsurfb.2013.06.048.

84. Salta, M.; Wharton, J.A.; Blache, Y.; Stokes, K.R.; Briand, J.-F. Marine Biofilms on Artificial Surfaces: Structure and Dynamics. *Environ. Microbiol.* **2013**, n/a-n/a, doi:10.1111/1462-2920.12186.

85. Schultz, M.P.; Bendick, J.A.; Holm, E.R.; Hertel, W.M. Economic Impact of Biofouling on a Naval Surface Ship. *Biofouling* **2011**, *27*, 87–98, doi:10.1080/08927014.2010.542809.

86. Wang, P.; Zhang, D.; Lu, Z. Slippery Liquid-Infused Porous Surface Bio-Inspired by Pitcher Plant for Marine Anti-Biofouling Application. *Colloids Surfaces B Biointerfaces* **2015**, *136*, 240–247, doi:10.1016/j.colsurfb.2015.09.019.

87. Yong, H.E.; Krishnamoorthy, K.; Hyun, K.T.; Kim, S.J. Preparation of ZnO Nanopaint for Marine Antifouling Applications. *J. Ind. Eng. Chem.* **2015**, *29*, 39–42, doi:10.1016/j.jiec.2015.04.020.

88. Feng, T.; Wu, J.; Chai, K.; Yang, P. The Anti-Biofouling Behavior of High Voltage Pulse Electric Field (HPEF) Mediated by Carbon Fiber Composite Coating in Seawater. *Bioelectrochemistry* **2018**, *123*, 137–144, doi:10.1016/j.bioelechem.2018.04.015.

89. Long, W.; Li, H.; Yang, B.; Huang, N.; Liu, L.; Gai, Z.; Jiang, X. Superhydrophobic Diamond-Coated Si Nanowires for Application of Anti-Biofouling'. *J. Mater. Sci. Technol.* **2020**, *48*, 1–8, doi:10.1016/j.jmst.2019.10.040.

90. Organization, I.M. Anti-fouling systems, in: Focus on IMO. In; London, 2002; p. 31.

91. Yee, M.S.L.; Khiew, P.S.; Chiu, W.S.; Tan, Y.F.; Kok, Y.Y.; Leong, C.O. Green Synthesis of Graphene-Silver Nanocomposites and Its Application as a Potent Marine Antifouling Agent. *Colloids Surfaces B Biointerfaces* **2016**, *148*, 392–401, doi:10.1016/j.colsurfb.2016.09.011.

92. Zhang, X.; Årstøl, E.; Nymark, M.; Fages-Lartaud, M.; Mikkelsen, Ø. The Development of Polydimethysiloxane/ZnO–GO Antifouling Coatings. *J. Clust. Sci.* **2021**, *7*, doi:10.1007/s10876-021-02165-7.

93. Tian, L.; Jin, E.; Wang, J.; Wang, X.; Bing, W.; Jin, H.; Zhao, J.; Ren, L. Exploring the Antifouling Effect of Elastic Deformation by DEM-CFD Coupling Simulation. *RSC Adv.* **2019**, *9*, 40855–40862, doi:10.1039/c9ra06761b.

94. Gu, J.; Li, L.; Huang, D.; Jiang, L.; Liu, L.; Li, F.; Pang, A.; Guo, X.; Tao, B. <I>In Situ</I> Synthesis of Graphene@cuprous Oxide Nanocomposite Incorporated Marine Antifouling Coating with Elevated Antifouling Performance. *Open J. Org. Polym. Mater.* **2019**, *09*, 47–62, doi:10.4236/ojopm.2019.93003.

95. Jin, H.; Bing, W.; Tian, L.; Wang, P.; Zhao, J. Combined Effects of Color and Elastic Modulus on

Antifouling Performance: A Study of Graphene Oxide/Silicone Rubber Composite Membranes. *Materials (Basel).* **2019**, *12*, doi:10.3390/ma12162608.

96. Bouranta, A.; Tudose, I.V.; Georgescu, L.; Karaiskou, A.; Vrithias, N.R.; Viskadourakis, Z.; Kenanakis, G.; Sfakaki, E.; Mitrizakis, N.; Strakantounas, G.; et al. 3D Printed Metal Oxide-Polymer Composite Materials for Antifouling Applications. *Nanomaterials* **2022**, *12*, doi:10.3390/nano12060917.

97. Kasper, A.; Aufdenblatten, S.; Forss, A.; Mohr, M.; Burtscher, H. Particulate Emissions from a Low-Speed Marine Diesel Engine. *Aerosol Sci. Technol.* **2007**, *41*, 24–32, doi:10.1080/02786820601055392.

98. Lee, W.-J.; Park, S.-H.; Jang, S.-H.; Kim, H.; Choi, S.K.; Cho, K.-H.; Cho, I.-S.; Lee, S.-M.; Choi, J.-H. Carbon Nanostructure of Diesel Soot Particles Emitted from 2 and 4 Stroke Marine Engines Burning Different Fuels. *J. Nanosci. Nanotechnol.* **2017**, *18*, 2128–2131, doi:10.1166/jnn.2018.14963.

99. Choi, J.H.; Cho, I. soon; Lee, J.S.; Park, S.K.; Lee, W.J.; Kim, H.; Chang, H.J.; Kim, J.Y.; Jeong, S.; Park, S.H. Characterization of Carbonaceous Particulate Matter Emitted from Marine Diesel Engine. *J. Mech. Sci. Technol.* **2016**, *30*, 2011–2017, doi:10.1007/s12206-016-0407-z.

100. Baek, H.M.; Kim, D.Y.; Lee, W.J.; Kang, J. Application of Soot Discharged from the Combustion of Marine Gas Oil as an Anode Material for Lithium Ion Batteries. *RSC Adv.* **2020**, *10*, 36478–36484, doi:10.1039/d0ra07195a.

101. Baszanowska, E.; Otremba, Z. TESTING OF THE SYNCHRONOUS FLUORESCENCE IN APPLICATION TO USED LUBRICATE OIL CHARACTERISATION. *J. KONES. Powertrain Transp.* **2016**, *23*, 39–44, doi:10.5604/12314005.1213548.

102. Shokair, I.R.; Johnson, M.S.; Schmitt, R.L.; Sickafoose, S.M. Concept for Maritime Near-Surface Surveillance Using Water Raman Scattering. *Appl. Opt.* **2018**, *57*, 4858, doi:10.1364/AO.57.004858.

103. Baszanowska, E.; Otremba, Z. Seawater Fluorescence Near Oil Occurrence. *Sustainability* **2020**, *12*, 4049, doi:10.3390/su12104049.

104. Bukin, O.A.; Mayor, A.Y.; Proschenko, D.Y.; Bukin, I.O.; Bolotov, V. V.; Chekhlenok, A.A.; Mun, S.A. Laser Spectroscopy Methods in the Development of Laser Sensor Elements for Underwater Robotics. *Atmos. Ocean. Opt.* **2017**, *30*, 475–480, doi:10.1134/S1024856017050049.

105. Horiba Jobin Yvon T64000 and U1000 12.

106. Dhankhar, D.; Nagpal, A.; Rentzepis, P.M. Cell-Phone Camera Raman Spectrometer. *Rev. Sci. Instrum.* **2021**, *92*, 054101, doi:10.1063/5.0046281.

107. Yuan, X.; Mayanovic, R.A. An Empirical Study on Raman Peak Fitting and Its Application to Raman Quantitative Research. *Appl. Spectrosc.* **2017**, *71*, 2325–2338, doi:10.1177/0003702817721527.

108. Liu, C.; Berg, R.W. Determining the Spectral Resolution of a Charge-Coupled Device (CCD) Raman Instrument. *Appl. Spectrosc.* **2012**, *66*, 1034–1043, doi:10.1366/11-06508.

109. IPC-2221A. **2003**.

110. Firmani, P.; Vitale, R.; Ruckebusch, C.; Marini, F. ANOVA-Simultaneous Component Analysis Modelling of Low-Level-Fused Spectroscopic Data: A Food Chemistry Case-Study. *Anal. Chim. Acta* **2020**, *1125*, 308–314, doi:10.1016/j.aca.2020.05.059.

111.	Gualerzi, A.; Kooijmans, S.A.A.; Niada, S.; Picciolini, S.; Brini, A.T.; Camussi, G.; Bedoni, M. Raman Spectroscopy as a Quick Tool to Assess Purity of Extracellular Vesicle Preparations and Predict Their Functionality. *J. Extracell. Vesicles* **2019**, *8*, 1568780, doi:10.1080/20013078.2019.1568780.

112.	Par, M.; Gamulin, O.; Marovic, D.; Klaric, E.; Tarle, Z. Raman Spectroscopic Assessment of Degree of Conversion of Bulk-Fill Resin Composites – Changes at 24 Hours Post Cure. *Oper. Dent.* **2015**, *40*, E92–E101, doi:10.2341/14-091-L.

113.	Člupek, M.; Matějka, P.; Volka, K. Noise Reduction in Raman Spectra: Finite Impulse Response Filtration versus Savitzky-Golay Smoothing. *J. Raman Spectrosc.* **2007**, *38*, 1174–1179, doi:10.1002/jrs.1747.

114.	Barton, S.J.; Ward, T.E.; Hennelly, B.M. Algorithm for Optimal Denoising of Raman Spectra. *Anal. Methods* **2018**, *10*, 3759–3769, doi:10.1039/C8AY01089G.

115.	Barton, S.; Alakkari, S.; O'Dwyer, K.; Ward, T.; Hennelly, B. Convolution Network with Custom Loss Function for the Denoising of Low SNR Raman Spectra. *Sensors* **2021**, *21*, 4623, doi:10.3390/s21144623.

116.	Chen, S.; Lin, X.; Yuen, C.; Padmanabhan, S.; Beuerman, R.W.; Liu, Q. Recovery of Raman Spectra with Low Signal-to-Noise Ratio Using Wiener Estimation. *Opt. Express* **2014**, *22*, 12102, doi:10.1364/OE.22.012102.

10. LIST OF FIGURES

11. LIST OF TABLES

12. BIOGRAPHY

Ivan Pavić was born on December 10, 1988 in Split. After finishing elementary school, he graduated from the Technical School in Split as a mechatronics technician. He continued his education by enrolling at the Faculty of Maritime Studies in Split in 2007 to study electrical and information technologies. In 2013, he obtained a master's degree in electrical and information technologies.

His professional experience starts as a IT referent at the Faculty of Maritime Studies in Split, after which he enrols as an electrical cadet on board the vessel M/V HS CHOPIN. Since April 1, 2016, he has been working as an assistant at the Faculty of Maritime Studies at the University of Split. In the academic year 2016/2017 he starts his postgraduate studies at the Faculty of Electrical Engineering, Mechanical Engineering and Naval Architecture in Split. In the second year of his postgraduate studies, he becomes interested in Raman spectroscopy and starts a collaboration with Ruđer Bošković Institute in Zagreb. From the academic year 2019/2020, he will move to the postgraduate programme "Technologies in Maritime Affairs" at the Faculty of Maritime Studies in Split and continue his work on Raman and photoluminescence spectroscopy in maritime applications.

He is fluent in written and spoken English.

Published papers and conference proceedings:

Pavić, **Ivan**; Šoda, Joško; Gašparić, Vlatko; Ivanda, Mile Raman and Photoluminescence Spectroscopy with a Variable Spectral Resolution. // Sensors, 21 (2021), 23; 7951, 11 doi:10.3390/s21237951.

Petković, Miro; Zubčić, Marko; Krčum, Maja; **Pavić**, **Ivan** Wind Assisted Ship Propulsion Technologies – Can they Help in Emissions Reduction?. // Naše more : (2021), 2; 102-109 doi:10.17818/NM/2021/2.6

Petković, Miro; Kezić, Danko; Vujović, Igor; **Pavić**, **Ivan** Target detection for visual collision avoidance system. // Pedagogika-Pedagogy, 93 (2021), 7s; 159-166 doi:10.53656/ped21-7s.14targ.

Jaram, Hrvoje; Vidan, Pero; Vukša, Srđan; **Pavić**, **Ivan** SITUATIONAL AWARENESS – KEY SAFETY FACTOR FOR THE OFFICER OF THE WATCH. // Pedagogika-Pedagogy, 93 (2021), 7s; 225-240 doi:10.53656/ped21-7s.20situ.

Bukljaš, Mihaela; **Pavić**, **Ivan**; Vidan, Pero; Vukša, Srđan Aerial foggy image rejection using single neuron. // Journal of applied engineering science, 18 (2020), 2; 238-242 doi:10.5937/jaes18-24786.

Bratić, Karlo; **Pavić**, **Ivan**; Vukša, Srđan; Stazić, Ladislav Review of Autonomous and Remotely Controlled Ships in Maritime Sector. // Transactions on Maritime Science, 8 (2019), 2; 253-265 doi:10.7225/toms.v08.n02.011.

Bukljaš Skočibušić, Mihaela; Vukša, Srđan; **Pavić**, **Ivan** Method for analyzing AIS data and determination of simultaneously ships passage via Strait of Split. // TransNav, 12 (2018), 4; 679-684 doi:10.12716/1001.12.04.05.

Vidan, Pero; Bukljaš Skočibušić, Mihaela; **Pavić**, **Ivan**; Vukša, Srđan Autonomous Systema & Ships - Training and Education on Maritime Faculties. // IMSC 2018 Book of Proceedings / Ivošević, Špiro ; Vidan, Pero (ur.). Kotor: Pomorski fakultet u Splitu, 2019. str. 91-101.

Nur, Assani; **Pavić**, **Ivan**; Vukša, Srđan; Laušić, Marina Analysis of the Nomoto ship model response to course changes using PID controller in Matlab/Simulink. // ICTS 2020 : Maritime, transport and logistics science - conference proceedings / Zanne, Marina ; Bajec, Patricija; Twrdy, Elen (ur.). Portorož: Faculty of Maritime Studies and Transport, Portoroz, 2020. str. 14-18. (https://www.bib.irb.hr/1080376).

Pavić, **Ivan**; Bonković, Mirjana; Šoda, Joško; Ivanda, Mile New Approach in Designing Raman Spectrometer with Variable Spectral Resolution. // Humboldt-Kolleg Zagreb, Hrvatska, 2019. str. 50-50.

APPENDIX A: PIC18(L)F2X/45K50 SPECIFICATIONS

Universal Serial Bus Features:
- USB V2.0 Compliant
- Crystal-less Full Speed (12 Mb/s) and Low-Speed Operation (1.5 Mb/s)
- Supports Control, Interrupt, Isochronous and Bulk Transfers
- Supports up to 32 Endpoints (16 Bidirectional)
- 1 Kbyte Dual Access RAM for USB
- On-Chip USB Transceiver

Flexible Oscillator Structure:
- 3x and 4xPLL Clock Multipliers
- Two External Clock modes, Up to 48 MHz (12 MIPS)
- Internal 31 kHz Oscillator
- Internal Oscillator, 31 kHz to 16 MHz
  - Factory calibrated to ± 1%
  - Self-tune to ± 0.20% max. from USB or secondary oscillator
- Secondary Oscillator using Timer1 @ 32 kHz
- Fail-Safe Clock Monitor:
  - Allows for safe shutdown if any clock stops

Peripheral Highlights:
- Up to 33 I/O pins plus 3 Input-Only Pins:
  - High-current Sink/Source 25 mA/25 mA
  - Three programmable external interrupts
  - 11 programmable interrupts-on-change
  - Nine programmable weak pull-ups
- Programmable slew rate
- SR Latch
- Enhanced Capture/Compare/PWM (ECCP) module:
  - One, two or four PWM outputs
  - Selectable polarity
  - Programmable dead time
  - Auto-shutdown and auto-restart
  - Pulse steering control

- Capture/Compare/PWM (CCP) module
- Master Synchronous Serial Port (MSSP) module Supporting 3-Wire SPI (all four modes) and I2C™ Master and Slave modes
- Two Analog Comparators with Input Multiplexing
- 10-Bit Analog-to-Digital (A/D) Converter module:
  - Up to 25 input channels
  - Auto-acquisition capability
  - Conversion available during Sleep
- Digital-to-Analog Converter (DAC) module:
  - Fixed Voltage Reference (FVR) with 1.024V, 2.048V and 4.096V output levels
  - 5-bit rail-to-rail resistive DAC with positive and negative reference selection
- High/Low-Voltage Detect module
- Charge Time Measurement Unit (CTMU):
  - Supports capacitive touch sensing for touch screens and capacitive switches
- Enhanced USART module:
  - Supports RS-485, RS-232 and LIN/J2602
  - Auto-wake-up on Start bit
  - Auto-Baud Detect

Extreme Low-Power Management with XLP:
- Sleep mode: 20 nA, typical
- Watchdog Timer: 300 nA, typical
- Timer1 Oscillator: 800 nA @ 32 kHz
- Peripheral Module Disable

Special Microcontroller Features:
- Low-Power, High-Speed CMOS Flash Technology
- C Compiler Optimized Architecture for Re-Entrant Code
- Power Management Features:

- Run: CPU on, peripherals on, SRAM on
- Idle: CPU off, peripherals on, SRAM on
- Sleep: CPU off, peripherals off, SRAM on
- Priority Levels for Interrupts
- Self-Programmable under Software Control
- 8 x 8 Single-Cycle Hardware Multiplier
- Extended Watchdog Timer (WDT):
  - Programmable period from 4 ms to 131s
- Single-Supply In-Circuit Serial Programming™ (ICSP™) via Two Pins
- In-Circuit Debug (ICD) with Three Breakpoints via Two Pins
- Optional dedicated ICD/ICSP Port (44-pin TQFP Package Only)
- Wide Operating Voltage Range:
  - F devices: 2.3V to 5.5V
  - LF devices: 1.8V to 3.6V
- Flash Program Memory of 10,000 Erase/Write Cycles Minimum and 20-year Data Retention.

APPENDIX B: SLA7078WPRT SPECIFICATION

- Power supply voltages, V_{BB}: 46 V(max.), 10 to 44 V normal operating range
- Logic supply voltages, V_{DD} : 3.0 to 5.5 V
- Maximum output currents: 1 A, 1.5 A, 2 A, 3 A
- Built-in sequencer
- Simplified clock-in stepping control
- Microstepping at full-, half-, quarter-, eighth-, and sixteenth-steps
- Built-in sense resistor, R_{SInt}
- All variants are pin-compatible for enhanced design flexibility
- ZIP type 23-pin molded package (SLA package)
- Self-excitation PWM current control with fixed off-time; off-time adjusted automatically by step reference current ratio (3 levels)
- Built-in synchronous rectifying circuit reduces losses at PWM off
- Synchronous PWM chopping function prevents motor noise in Hold mode
- Sleep mode for reducing the IC input current in stand-by state
- Built-in protection circuitry against motor coil opens/shorts option available

ABSOLUTE MAXIMUM RATINGS AT TA = 25°C

- Load (Motor) Supply Voltage, V_M 46 V
- Main Power Supply Voltage, V_{BB} 46 V
- Logic Supply Voltage, V_{DD} 7 V
- Output Current, I_{OUT}
  SLA7075M ....1.0 A
  SLA7076M ....1.5 A
  SLA7077M ....2.0 A
  SLA7078M ....3.0 A
- Logic Input Voltage, V_{IN} –0.3 to V_{DD} +0.3 V
- REF Input Voltage, V_{REF} –0.3 to V_{DD} +0.3 V
- Sense Voltage, V_{Sint} t_w < 1 μs is not considered ±2 V

- Power Dissipation, P_D
  Without heat sink 4.7 W
- Junction Temperature, T_J 150°C
- Ambient Temperature, T_A –20 to 85°C
- Storage Temperature, Tstg –30 to 150°C

APPENDIX C: ELECTRONIC SCHEMATIC FOR SERVO MOTOR CONTROL

SERVO MOTOR CONTROL BOARD

FACULTY OF MARITIME STUDIES IN SPLIT
Sheet: /
File: Doktorat_V2.sch
Title: SERVO MOTOR CONTROL BOARD
Size: A3   Date: 2021-06-16   Rev: 2.0
KiCad E.D.A. kicad (5.1.6)-1   Id: 1/1

APPENDIX D: BILL OF MATERIALS

Designator	Qty	Manufacturer	Mfg Part#	Package/Footprint	Type
C1, C2, C5	3	KEMET	T491C106K025AT	CAP TANTALUM 10UF 25V 10% CASE-C 6032	SMD
C3, C4, C7, C8, C9, C10, C12	7	Samsung Electro-Mechanics	CL05B104KO5NNNC	CAP CER 100nF 16V 10% X7R 0402	SMD
C6, C16, C17, C18	4	Samsung Electro-Mechanics	CL10A106MO8NQNC	CAP CER 10UF 16V 20% X5R 0603	SMD
C11	1	Lelon	VE-101M1ETR-0607	CAP ELEC 100UF 25V 20% 6.3x7.7mm	SMD
C13, C14	2	Samsung Electro-Mechanics	CL05C120JB5NNNC	CAP CER 12pF 50V 5% 0402	SMD
C15	1	Samsung Electro-Mechanics	CL05A474KP5NNNC	CAP CER 470nF 10V 10% X5R 0402	SMD
D1, D2, D3, D4	4	BORN	B5817W	SBD 20V 1A 750mV SOD-123	SMD
D5, D11, D12	3	Everlight Elec	19-217/GHC-YR1S2/3T	LED Green 0603	SMD
D6, D7, D8, D9, D10	5	Everlight Elec	19-217/BHC-ZL1M2RY/3T	LED Blue 0603	SMD
J1	1	MOLEX	0022272021	CONN HEADER 2POS 2.54MM VERT	thru-hole
J2, J4, J7	3	MOLEX	0022284036	CONN HEADER 3POS 2.54MM VERT	thru-hole
J3	1	CJT(Changjiang Connectors)	A2541HWV-2x6P	FEMALE HEADER 12 2 STRAIGHT 2.54MM 180degrees	thru-hole
J5	0				
J6, J9	2	MOLEX	0022284061	CONN HEADER 6POS 2.54MM VERT	thru-hole
J8	0				
Q1	1	VBsemi Elec	NDS352AP	P-CHANNEL 30V (D-S) MOSFET SOT-23-3	SMD
R1, R3, R4, R5, R6, R7, R10, R11	8	UNI-ROYAL(Uniroyal Elec)	0603WAF1001T5E	RESISTOR 1KOhm 1% 0.1W 0603	SMD

R2	1	UNI-ROYAL(Uniroyal Elec)	0603WAF1200T5E	RESISTOR 120Ohm 1% 0.1W 0603	SMD
R8	1	UNI-ROYAL(Uniroyal Elec)	0603WAF1003T5E	RESISTOR 100KOhm 1% 1/10W 0603	SMD
R9	1	UNI-ROYAL(Uniroyal Elec)	0603WAF3302T5E	RESISTOR 33KOhm 1% 0.1W 0603	SMD
R12, R13	2	UNI-ROYAL(Uniroyal Elec)	0603WAF1203T5E	RESISTOR 120KOhm 1% 0.1W 0603	SMD
R14, R17	2	UNI-ROYAL(Uniroyal Elec)	0603WAF1002T5E	RESISTOR 10KOhm 1% 1/10W 0603	SMD
R15	1	UNI-ROYAL(Uniroyal Elec)	0603WAF4700T5E	RESISTOR 470Ohm 1% 0.1W 0603	SMD
R16, R18, R19	3	UNI-ROYAL(Uniroyal Elec)	0603WAF0000T5E	RESISTOR 0Ohm 1% 0.1W 0603	SMD
RV1	1	Bourns Inc.	3314J-1-102E	TRIMMER 1KOhm 0.25W J LEAD TOP	SMD
RV2	1	Bourns Inc.	3314J-1-103E	TRIMMER 10KOhm 0.25W J LEAD TOP	SMD
SW1	1	C&K	PTS645SM43SMTR92LFS	SWITCH TACTILE SPST-NO 0.05A 12V	SMD
U1	1	Texas Instruments	LM1084IS-ADJ/NOPB	IC ADJ VOL REG 5A TO-263-3 (D2-PAK)	SMD
U2	1	STMicroelectronics	L7805ABD2T-TR	IC REG LINEAR 5V 1.5A TO-263-2 (D2-PAK)	SMD
U3	0				
U4	1	Microchip Technology	PIC18F45K50-I/PT	IC PIC MCU 32K FLASH, 2K RAM, 44TQFP	SMD
U5	1	STMicroelectronics	USBLC6-2SC6	ESD PROTECTION DEVICE 5.25V 6V 17V 5A SOT-23-6	SMD
Y1	1	Yangxing Tech	X322516MLB4SI	CRYSTAL 16MHz SMD-3225_4P 3.2x2.5mm	SMD

APPENDIX E: C# GUI SOURCE CODE

Main Window XAML Code:

```xml
<Window x:Class="RamanApp.MainWindow"
        xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
        xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
        xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
        xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
        xmlns:oxy="http://oxyplot.org/wpf"
        xmlns:local="clr-namespace:RamanApp"
        mc:Ignorable="d"
        Title="Raman &amp; PL"
        WindowState="Maximized"
        Height="800"
        Width="1200"
        Loaded="Window_Loaded"
        Closing="Window_Closing"
        Background="#222222">

<DockPanel Background="#07090f">
        <ToolBarTray DockPanel.Dock="Top" Background="#07090f">
            <ToolBar Background="#07090f">
                <Button ToolTip="Open File">
                    <Image Source="./Resources/Open.png"></Image>
                </Button>
                <Button x:Name="btnSaveData" ToolTip="Save File"
Click="btnSaveData_Click" IsEnabled="False">
                    <Image Source="./Resources/Save.png"></Image>
                </Button>
                <Separator></Separator>
                <Label Content="Set Exposure time:" Foreground="#E4E4E4"/>
                <RibbonRadioButton x:Name="radioMin" Label="Min." Margin="0,5,0,5"
CheckedBackground="#045757" ToolTip="Select minutes for exposure"
Checked="radioMin_Checked"  Foreground="#E4E4E4"></RibbonRadioButton>
                <RibbonRadioButton x:Name="radioSec" Label="Sec." Margin="0,5,0,5"
CheckedBackground="#045757" ToolTip="Select seconds for exposure"
Checked="radioSec_Checked"  Foreground="#E4E4E4"></RibbonRadioButton>
                <RibbonRadioButton x:Name="radioTen" Label="Ten." Margin="0,5,0,5"
CheckedBackground="#045757" ToolTip="Select tenths for exposure"
Checked="radioTen_Checked"  Foreground="#E4E4E4"></RibbonRadioButton>
                <RibbonRadioButton x:Name="radioMs" Label="ms" Margin="0,5,0,5"
CheckedBackground="#045757" ToolTip="Select miliseconds for exposure"
Checked="radioMs_Checked"  Foreground="#E4E4E4"></RibbonRadioButton>
                <Slider x:Name="timeSlider" Value="20" Minimum="1" Maximum="100"
Width="100" SmallChange="1" VerticalAlignment="Center"
ValueChanged="timeSlider_ValueChanged"  LargeChange="5" TickFrequency="1"
IsSnapToTickEnabled="True" Foreground="#E4E4E4"></Slider>
                <Label x:Name="lblSetTime" Content="0 sec." Width="70"
Foreground="#E4E4E4"></Label>
                <Button x:Name="btnSetExpTime" Content="Set Time" Foreground="#E4E4E4"
ToolTip="Apply Exposure Time" Width="65" Height="Auto"
VerticalContentAlignment="Center" HorizontalContentAlignment="Center"
Margin="0,0,10,0"  Background="#045757" BorderBrush="DarkOrange" IsEnabled="False"
Click="btnSetExpTime_Click">
                </Button>
                <Label x:Name="lblRemain" Width="100" Content="T - 0 ms"
Foreground="#E4E4E4"></Label>
                <Separator></Separator>
                <Label Content="Binning:" Foreground="#E4E4E4"></Label>
```

```xml
<Label Content="Set X" Foreground="#E4E4E4"></Label>
<ComboBox x:Name="ComBoxBinX" Width="30" Background="#045757"
BorderBrush="Orange" BorderThickness="1"
SelectionChanged="ComBoxBinX_SelectionChanged_1">
        <ComboBoxItem Content="1" Foreground="#E4E4E4"></ComboBoxItem>
        <ComboBoxItem Content="2" Foreground="#E4E4E4"></ComboBoxItem>
        <ComboBoxItem Content="3" Foreground="#E4E4E4"></ComboBoxItem>
        <ComboBoxItem Content="4" Foreground="#E4E4E4"></ComboBoxItem>
        <ComboBoxItem Content="8" Foreground="#E4E4E4"></ComboBoxItem>
        <ComboBoxItem Content="16" Foreground="#E4E4E4"></ComboBoxItem>
</ComboBox>

<Label Content="Set Y" Foreground="White"></Label>
<ComboBox x:Name="ComBoxBinY" Width="30" Background="#045757"
BorderBrush="Orange" BorderThickness="1"
SelectionChanged="ComBoxBinY_SelectionChanged">
        <ComboBoxItem Content="1" Foreground="#E4E4E4"></ComboBoxItem>
        <ComboBoxItem Content="2" Foreground="#E4E4E4"></ComboBoxItem>
        <ComboBoxItem Content="3" Foreground="#E4E4E4"></ComboBoxItem>
        <ComboBoxItem Content="4" Foreground="#E4E4E4"></ComboBoxItem>
        <ComboBoxItem Content="8" Foreground="#E4E4E4"></ComboBoxItem>
        <ComboBoxItem Content="16" Foreground="#E4E4E4"></ComboBoxItem>
</ComboBox>
<Separator></Separator>
<Button x:Name="btnOneExposure" ToolTip="One Time Exposure"
Width="Auto" Height="Auto" Margin="0,0,10,0" IsEnabled="False"
Click="btnOneExposure_Click">
        <Image Source="./Resources/Run_16x.png" Height="15"></Image>
</Button>
<Button x:Name="btnConExposure" ToolTip="Continuos Exposure"
Width="Auto" Height="Auto" Margin="0,0,10,0" IsEnabled="False"
Click="btnConExposure_Click">
        <Image Source="./Resources/RecordG.png" Height="15"></Image>
</Button>
<Button x:Name="btnStopExposure" ToolTip="Stop Exposure" Width="Auto"
Height="Auto" Margin="0,0,10,0" IsEnabled="False" Click="btnStopExposure_Click">
        <Image Source="./Resources/StopG.png" Height="15"></Image>
</Button>
<Separator></Separator>
<Label Content="Download Time:" Foreground="#E4E4E4"></Label>
<Label x:Name="lblDwn" Content="{Binding LblDown}" Width="60"
Foreground="#E4E4E4"></Label>
<Separator></Separator>
<Grid>
        <ProgressBar x:Name="proBar" Width="80" Height="15"
Background="#E4E4E4" Foreground="Orange"></ProgressBar>
        <TextBlock x:Name="proBarText" HorizontalAlignment="Center"
VerticalAlignment="Center" Foreground="#222222"></TextBlock>
</Grid>
<Separator></Separator>
<Button x:Name="btnCanvas" Content="Hide" Click="btnCanvas_Click"
Width="40" Background="#045757" BorderBrush="DarkOrange" Foreground="#E4E4E4"
ToolTip="Show/Hide Image" HorizontalContentAlignment="Center"
VerticalContentAlignment="Center"></Button>
<Separator></Separator>
<Button x:Name="btnDarkFrame" Grid.Column="0" Content="Dark Frame is
OFF" Click="btnDarkFrame_Click" Width="105" Background="#045757"
BorderBrush="DarkOrange" Foreground="#E4E4E4" ToolTip="Enable/disable Dark Frame Mode"
HorizontalContentAlignment="Center" VerticalContentAlignment="Center"></Button>
<Separator></Separator>
<Grid>
```

```xml
                        <CheckBox x:Name="chcRemoveDark" Grid.Row="0" Grid.Column="1"
IsChecked="False" Checked="chcRemoveDark_Checked" Unchecked="chcRemoveDark_Unchecked"
Width="15" Height="15" HorizontalAlignment="Right"></CheckBox>
                    </Grid>
                    <Separator></Separator>
                </ToolBar>
            </ToolBarTray>
            <ScrollViewer>

<Grid DockPanel.Dock="Left" Width="170">
                    <Grid.RowDefinitions>
                        <RowDefinition Height="220"></RowDefinition>
                        <RowDefinition Height="160"></RowDefinition>
                        <RowDefinition Height="210"></RowDefinition>
                        <RowDefinition Height="270"></RowDefinition>
                    </Grid.RowDefinitions>
                    <GroupBox Grid.Row="0">
                        <GroupBox.Header>
                            <TextBlock FontWeight="Bold" Foreground="#E4E4E4">
                            <Run Text="select camera"></Run>
                            </TextBlock>
                        </GroupBox.Header>
                        <StackPanel Background="#07090f">
                            <TextBlock Foreground="#E4E4E4">
                            <Run Text="Connect to Camera"></Run>
                            </TextBlock>
                            <Button x:Name="btnConnect" Content="CONNECT"
Foreground="#E4E4E4" Background="#045757" BorderBrush="DarkOrange"
Click="btnConnect_Click"></Button>
                            <TextBox x:Name="txtBox" Background="#07090f" Height="100"
TextWrapping="Wrap" VerticalScrollBarVisibility="Auto" Foreground="White"></TextBox>
                            <Label x:Name="lblSelCam" Content="Select camera model"
Visibility="Hidden" HorizontalAlignment="Left" Foreground="White"></Label>
                            <ComboBox x:Name="comBoxSelCam" Visibility="Hidden"
SelectionChanged="comBoxSelCam_SelectionChanged"></ComboBox>
                        </StackPanel>
                    </GroupBox>
                    <GroupBox Grid.Row="1">
                        <GroupBox.Header>
                            <TextBlock FontWeight="Bold" Foreground="#E4E4E4">
                            <Run Text="camera cooler"></Run>
                            </TextBlock>
                        </GroupBox.Header>
                        <StackPanel Background="#07090f">
                            <CheckBox x:Name="checkBoxCool" Content="Turn OFF cooler"
Checked="checkBoxCool_Checked"  Unchecked="checkBoxCool_Checked"
Foreground="#E4E4E4"></CheckBox>
                            <TextBlock Foreground="#E4E4E4" ><Run Text="Temp. setpoint
[°C]:"/></TextBlock>
                            <Label x:Name="lblTempSet" Content="0.0"
Foreground="#E4E4E4"/>
                            <TextBlock Foreground="#E4E4E4" ><Run Text="Actual temp
[°C]:"/></TextBlock>
                            <Label x:Name="lblTempAct" Content="0.0"
Foreground="#E4E4E4"/>
                            <Slider x:Name="slidTemp" Value="0" Maximum="350" Minimum="-
400" SmallChange="1" LargeChange="5" TickFrequency="50" IsSnapToTickEnabled="True"
ValueChanged="slidTemp_ValueChanged"></Slider>
                        </StackPanel>
                    </GroupBox>
                    <GroupBox Grid.Row="2">
                        <GroupBox.Header>
                            <TextBlock FontWeight="Bold" Foreground="#E4E4E4">
```

```xml
                    <Run Text="data"></Run>
                </TextBlock>
            </GroupBox.Header>
            <StackPanel Background="#07090f">
                <Label Content="Material Name:" Foreground="#E4E4E4"></Label>
                <TextBox x:Name="testBox"></TextBox>
                <Label Content="Laser type:" Foreground="#E4E4E4"></Label>
                <ComboBox x:Name="ComBoxLaserType" Width="100"
Background="#045757" BorderBrush="Orange" BorderThickness="1"
SelectionChanged="ComBoxLaserType_SelectionChanged">
                    <ComboBoxItem Content="532"></ComboBoxItem>
                    <ComboBoxItem Content="785"></ComboBoxItem>
                </ComboBox>
                <Label Content="Laser Power [mW]:"
Foreground="#E4E4E4"></Label>
                <TextBox x:Name="textLaserPower"></TextBox>
                <Label Content="Gratting type [gr/mm]:"
Foreground="#E4E4E4"></Label>
                <TextBox x:Name="textGrating"></TextBox>
            </StackPanel>
        </GroupBox>
        <GroupBox Grid.Row="3">
            <GroupBox.Header>
                <TextBlock FontWeight="Bold" Foreground="#E4E4E4">
                <Run Text="camera controls"></Run>
                </TextBlock>
            </GroupBox.Header>
            <StackPanel Background="#07090f">
                <Label x:Name="lblBoardStatus" Content="Board Connected"
Foreground="#E4E4E4"></Label>
                <Label Content="Camera Zoom" Foreground="#E4E4E4"></Label>
                <ComboBox Width="40"></ComboBox>
                <Label Content="CCD position" Foreground="#E4E4E4"></Label>
                <Button x:Name="btnCCD" Width="50"
Click="btnCCD_Click"></Button>
                <Label Content="Camera Shutter" Foreground="#E4E4E4"></Label>
                <ComboBox Width="40"></ComboBox>
                <Label Content="Gratting Angle" Foreground="#E4E4E4"></Label>

                <Button x:Name="btnMove" Content="Move" Margin="0,5,0,0"
Background="GreenYellow" Click="btnMove_Click"></Button>
            </StackPanel>
        </GroupBox>

    </Grid>

</ScrollViewer>
<Grid DockPanel.Dock="Bottom">

    <Grid.RowDefinitions>
        <RowDefinition Height="40"></RowDefinition>
        <RowDefinition></RowDefinition>
    </Grid.RowDefinitions>
    <Grid.ColumnDefinitions>
        <ColumnDefinition></ColumnDefinition>
        <ColumnDefinition Width="120"></ColumnDefinition>
    </Grid.ColumnDefinitions>
    <TextBox x:Name="txtGrating" Text="{Binding ElementName=slidGrating,
Path=Value, Mode=TwoWay}" Grid.Row="0" Grid.Column="0" VerticalAlignment="Center"
HorizontalAlignment="Center" Width="70" Height="20" Margin="0,0,200,0"></TextBox>
    <Button x:Name="btnStopMove" Content="GO TO" Grid.Row="0" Grid.Column="0"
HorizontalAlignment="Center" VerticalAlignment="Center" Width="70" Height="20"
```

```xml
Background="#045757" BorderBrush="DarkOrange" Foreground="#E4E4E4"
Click="btnStopMove_Click"></Button>
            <Slider x:Name="slidGrating" Grid.Row="1" Grid.Column="0"
ValueChanged="slidGrating_ValueChanged"  IsSnapToTickEnabled="True" Width="Auto"
Maximum="65535" AutoToolTipPlacement="TopLeft" AutoToolTipPrecision="0" Value="200"
Minimum="1"></Slider>
            <ProgressBar x:Name="proMoving" Grid.Row="0" Grid.Column="0" Width="100"
HorizontalAlignment="Center" Margin="200,0,0,0" Background="#E4E4E4"
Foreground="Orange" Height="15"></ProgressBar>
            <CheckBox x:Name="chcBoxCont" Grid.Row="0" Grid.Column="0"
HorizontalAlignment="Left" IsChecked="False" Checked="chcBoxCont_Checked"
Unchecked="chcBoxCont_Unchecked" IsEnabled="False"></CheckBox>
            <CheckBox x:Name="masterBiasFrame" Grid.Row="0" Grid.Column="0"
HorizontalAlignment="Right" Margin="0,0,250,0" Checked="masterBiasFrame_Checked"
Unchecked="masterBiasFrame_Unchecked"></CheckBox>
            <CheckBox x:Name="masterFlatFrame" Grid.Row="0" Grid.Column="0"
HorizontalAlignment="Right" Margin="0,0,200,0" Checked="masterFlatFrame_Checked"
Unchecked="masterFlatFrame_Unchecked"></CheckBox>
            <CheckBox x:Name="masterBiasGen" Grid.Row="0" Grid.Column="0"
HorizontalAlignment="Right" Margin="0,0,150,0" Checked="masterBiasGen_Checked"
Unchecked="masterBiasGen_Unchecked"></CheckBox>
            <CheckBox x:Name="masterBiassubfromDark" Grid.Row="0" Grid.Column="0"
HorizontalAlignment="Right" Margin="0,0,100,0" Checked="masterBiassubfromDark_Checked"
Unchecked="masterBiassubfromDark_Unchecked"></CheckBox>
            <CheckBox x:Name="masterFlatMax" Grid.Row="0" Grid.Column="0"
HorizontalAlignment="Right" Margin="0,0,50,0" Checked="masterFlatMax_Checked"
Unchecked="masterFlatMax_Unchecked"></CheckBox>
            <CheckBox x:Name="flatDivide" Grid.Row="0" Grid.Column="0"
HorizontalAlignment="Right" Margin="0,0,0,0" Checked="flatDivide_Checked"
Unchecked="flatDivide_Unchecked"></CheckBox>


            <Label x:Name="lblRealTime" Grid.Row="0" Grid.Column="1"
Background="#1E2529" Height="30" Foreground="#E4E4E4"></Label>
            <Label x:Name="lblClock" Grid.Row="1" Grid.Column="1" Background="#1E2529"
Height="30" Foreground="#E4E4E4"></Label>
        </Grid>

        <Grid x:Name="LayoutRoot">
            <oxy:PlotView x:Name="myPlot">

            </oxy:PlotView>
            <Canvas x:Name="Backplane" HorizontalAlignment="Right"
VerticalAlignment="Top" Width="550" Height="440"  Margin="0,10,50,0"
MouseLeftButtonDown="Backplane_MouseLeftButtonDown"
MouseLeftButtonUp="Backplane_MouseLeftButtonUp"  MouseLeave="Backplane_MouseLeave"
MouseMove="Backplane_MouseMove"  MouseRightButtonDown="Backplane_MouseRightButtonDown"
Visibility="Hidden">

                <Canvas.Background>
                    <VisualBrush Stretch="Fill">
                        <VisualBrush.Visual>
                            <Image x:Name="img"></Image>
                        </VisualBrush.Visual>
                    </VisualBrush>

                </Canvas.Background>

                <Rectangle Stroke="DarkOrange" Margin="270,230,0,0"
HorizontalAlignment="Right" VerticalAlignment="Top" Width="4" Height="4"
Visibility="Visible" Fill="DarkOrange" ></Rectangle>
```

```xml
                <Rectangle x:Name="Rect" Stroke="DarkOrange" Visibility="Collapsed"
></Rectangle>
                <Thumb DragDelta="Thumb_DragDelta" Canvas.Left="0" Canvas.Top="0">
                    <Thumb.Template>
                        <ControlTemplate>
                            <Line x:Name="uppLine" X1="0" X2="550" Y1="10" Y2="10"
StrokeThickness="2" Stroke="DarkOrange" StrokeDashArray="1"></Line>
                        </ControlTemplate>
                    </Thumb.Template>
                </Thumb>
                <Thumb DragDelta="Thumb_DragDelta" Canvas.Left="0" Canvas.Top="0">
                    <Thumb.Template>
                        <ControlTemplate>
                            <Line X1="0" X2="550" Y1="430" Y2="430"
StrokeThickness="2" Stroke="DarkOrange"></Line>
                        </ControlTemplate>
                    </Thumb.Template>
                </Thumb>
            </Canvas>

        </Grid>
    </DockPanel>
</Window>
```

Main Window CS Code:

```csharp
public BackgroundWorker worker1; // thread for one time exposure
public BackgroundWorker worker2; // thread for drawing image
public static BackgroundWorker worker3; // thread for USB
public BackgroundWorker worker4; // thread for expo timer
public static BackgroundWorker worker6; // thread for clock display
public static DispatcherTimer gratingTimer;
public static Color buttonBack;
public MainWindow()
{
        InitializeComponent();
        Timers timers = new Timers(this);
        Exposure exposure = new Exposure(this);
        DrawImage drawImage = new DrawImage(this);
        DrawSpectra drawSpectra = new DrawSpectra(this);
        Admin admin = new Admin(this);
        buttonBack = Color.FromArgb(255,4,87,87);

        DrawSpectra.initPlotModel();

        worker1 = new BackgroundWorker();
        worker1.DoWork += Worker1_DoWork;
        worker1.RunWorkerCompleted += Worker1_RunWorkerCompleted;
        worker1.WorkerSupportsCancellation = true;

        worker2 = new BackgroundWorker();
        worker2.DoWork += Worker2_DoWork;
        worker2.RunWorkerCompleted += Worker2_RunWorkerCompleted;
        worker2.WorkerSupportsCancellation = true;

        worker3 = new BackgroundWorker();
        worker3.DoWork += Worker3_DoWork;

        worker4 = new BackgroundWorker();
        worker4.DoWork += Worker4_DoWork;
```

```csharp
            worker4.WorkerSupportsCancellation = true;

            worker6 = new BackgroundWorker();
            worker6.DoWork += Worker6_DoWork;


            gratingTimer = new DispatcherTimer();
            gratingTimer.Interval = TimeSpan.FromMilliseconds(500);
            gratingTimer.Tick += gratingTimer_Tick;
        }

        private void Window_Loaded(object sender, RoutedEventArgs e)
        {
            btnSetExpTime.IsEnabled = false;
            radioTen.IsChecked = true;
            checkBoxCool.IsEnabled = false;
            updateTime();
            Timers.TimerSetup();
            worker6.RunWorkerAsync();
                for (Int32 index = 0; index <Variables.maxNumberOfCameras; index++)
                    Variables.cameraImageData[index] = Variables.camImgData;
        }

        private void Window_Closing(object sender, System.ComponentModel.CancelEventArgs e)
        {
            if (Variables.numberOfCameras > 0)
            {
                for (Int32 index = 0; index >= Variables.numberOfCameras; index++)
                {
                    SxSDKAccess.sxClose(Variables.camHandles[index]);
                }
            }
            Variables.ms.Dispose();
            IntPtr windowHandle = new WindowInteropHelper(this).Handle;
            HwndSource source = HwndSource.FromHwnd(windowHandle);
            source.RemoveHook(new HwndSourceHook(this.WndProc));
        }


        private void updateTime()
        {
            Int32 multiplier = 1000; // assume milliseconds
            if (radioMin.IsChecked == true)
            {
                multiplier = 60000;
                lblSetTime.Content = timeSlider.Value.ToString() + " min.";
            }
            if (radioSec.IsChecked == true)
            {
                multiplier = 1000;
                lblSetTime.Content = timeSlider.Value.ToString() + " sec.";
            }
            if (radioTen.IsChecked == true)
            {
                multiplier = 100;
                lblSetTime.Content = (timeSlider.Value * 100).ToString() + " ms";
            }
            if (radioMs.IsChecked == true)
            {
                multiplier = 1;
                lblSetTime.Content = timeSlider.Value.ToString() + " ms";
            }
```

```csharp
        Variables.timerValue = (UInt32)(timeSlider.Value * multiplier);
    }

    private void radioMin_Checked(object sender, RoutedEventArgs e)
    {
        updateTime();
    }

    private void radioSec_Checked(object sender, RoutedEventArgs e)
    {
        updateTime();
    }

    private void radioTen_Checked(object sender, RoutedEventArgs e)
    {
        updateTime();
    }

    private void radioMs_Checked(object sender, RoutedEventArgs e)
    {
        updateTime();
    }

    private void timeSlider_ValueChanged(object sender,
    RoutedPropertyChangedEventArgs<double> e)
    {
        updateTime();
    }

    private void btnConnect_Click(object sender, RoutedEventArgs e)
    {

        txtBox.Text += "Raman & PL software\r\n";
        Variables.numberOfCameras = SxSDKAccess.sxOpen(Variables.camHandles);

        if (Variables.numberOfCameras > 0)
        {
            btnConnect.IsEnabled = false;
            btnConnect.Content = "Connected";
            txtBox.Text += "Number of cameras = " + Variables.numberOfCameras.ToString() +
"\r\n\r\n";

            for (Int32 index = 0; index < Variables.numberOfCameras; index++)
            {
                txtBox.Text += "Handle" + index.ToString() + " = " +
Variables.camHandles[index].ToString() + "\r\n";
                Variables.cameraInfo[index] =
sxInfo.readInfoPack(Variables.camHandles[index]);//, ref tempInfo);
                if (Variables.cameraInfo[index].result == 1)
                {
                    // Display some of the data
                    txtBox.Text += "Firmware = " +
Variables.cameraInfo[index].upperFirmwareVersion.ToString() + "."
                                        +
Variables.cameraInfo[index].lowerFirmwareVersion.ToString("D2") + "\r\n";
                    txtBox.Text += "Build = " +
Variables.cameraInfo[index].buildNumber.ToString() + "\r\n";
                    txtBox.Text += "Model = 0x" +
Variables.cameraInfo[index].cameraModel.ToString("X2")
                                        + ", Name = " +
Variables.cameraInfo[index].cameraName + "\r\n";
```

```csharp
                        txtBox.Text += "Width = " +
Variables.cameraInfo[index].imageWidth.ToString() + "\r\n";
                        txtBox.Text += "Height = " +
Variables.cameraInfo[index].imageHeight.ToString() + "\r\n";
                        txtBox.Text += "Fitted = ";
                        if (Variables.cameraInfo[index].hasStar2K)
                            txtBox.Text += "Star2K ";
                        if (Variables.cameraInfo[index].hasEEProm)
                            txtBox.Text += "EEprom ";
                        if (Variables.cameraInfo[index].hasGuider)
                            txtBox.Text += "Guider ";
                        if (Variables.cameraInfo[index].hasCooler)
                            txtBox.Text += "Cooler ";
                        if (Variables.cameraInfo[index].hasShutter)
                            txtBox.Text += "Shutter ";
                        if (Variables.cameraInfo[index].isInterlaced)
                            txtBox.Text += "Interlaced CCD ";
                        txtBox.Text += "\r\n\r\n";
                        comBoxSelCam.Items.Add(Variables.cameraInfo[index].cameraName);
                    }
                }
                lblSelCam.Visibility = Visibility.Visible;
                comBoxSelCam.Visibility = Visibility.Visible;
                Variables.firstMove = true;
                Variables.test = true;
                Variables.darkShutter = true;
            }
            else
                txtBox.Text += "No cammeras Found!\r\n";
        }

        private void comBoxSelCam_SelectionChanged(object sender, SelectionChangedEventArgs e)
        {
            btnSetExpTime.IsEnabled = true; //allow timer setting once camera selected

            Timers.timer1.Stop();           //assume no cooler
            checkBoxCool.IsChecked = false; //assume no cooler
            checkBoxCool.IsEnabled = false; //assume no cooler
            Variables.selectedCamera = comBoxSelCam.SelectedIndex;
            txtBox.Text += "selectedCamera = " + Variables.selectedCamera.ToString() + "\r\n";
            txtBox.Text += "hasShutter = " +
Variables.cameraInfo[Variables.selectedCamera].hasShutter.ToString() + "\r\n";
            txtBox.Text += "Name = " +
Variables.cameraInfo[Variables.selectedCamera].cameraName + "\r\n";
            if (Variables.cameraInfo[Variables.selectedCamera].hasShutter)
            {
                SxSDKAccess.sxSetShutter(Variables.camHandles[Variables.selectedCamera], 1);
            }
            if (Variables.cameraInfo[Variables.selectedCamera].hasCooler)
            {
                Timers.timer1.Start();
                checkBoxCool.IsEnabled = true;
                checkBoxCool.IsChecked = true;
                Variables.setPointChanged = true;
                btnSetExpTime.IsEnabled = true;
            }
        }

        private void checkBoxCool_Checked(object sender, RoutedEventArgs e)
        {
            if (checkBoxCool.IsChecked == true)
            {
```

```csharp
        checkBoxCool.Content = "ON";
        Variables.setPointChanged = true; // ensure set point is updated after switch
on
    }
    else
    {
        checkBoxCool.Content = "OFF";
    }

}

private void slidTemp_ValueChanged(object sender,
RoutedPropertyChangedEventArgs<double> e)
{
    Variables.normalisedTemperature = (UInt16)(slidTemp.Value + 2730);
    lblTempSet.Content = ((Single)((Variables.normalisedTemperature - 2730.0) /
10.0)).ToString();
    Variables.setPointChanged = true;
}


private void btnSetExpTime_Click(object sender, RoutedEventArgs e)
{
    if ((Variables.numberOfCameras > 0) && (Variables.timerValue > 0))    // must be
connected
    {
        Int32 Result =
(Int32)SxSDKAccess.sxSetTimer(Variables.camHandles[Variables.selectedCamera],
Variables.timerValue);
        Timers.timer2.Start();
        btnStopExposure.IsEnabled = true;
    }
}

private void btnOneExposure_Click(object sender, RoutedEventArgs e)
{
    proBarText.Text = "exposing";
    worker1.RunWorkerAsync();
    btnOneExposure.IsEnabled = false;
    btnConExposure.IsEnabled = false;
    Variables.continues = false;
}

private void btnConExposure_Click(object sender, RoutedEventArgs e)
{
    btnStopExposure.IsEnabled = true;
    btnConExposure.IsEnabled = false;
    btnOneExposure.IsEnabled = false;
    Variables.continues = true;
    worker1.RunWorkerAsync();
}

private void Worker1_DoWork(object sender, DoWorkEventArgs e)
{
        Boolean waitForExposureEnd = false;
        Exposure.getData();

        // For exposures of less than 1 second wait for the exposure to complete
        if (Variables.timerValue < 1000)
            waitForExposureEnd = true;

        if (Variables.cameraInfo[Variables.selectedCamera].hasShutter)
```

```csharp
            SxSDKAccess.sxSetShutter(Variables.camHandles[Variables.selectedCamera],
0); ;
        Variables.Result =
makeExposure(Variables.cameraImageData[Variables.selectedCamera], waitForExposureEnd);
        Dispatcher.Invoke(new Action(() =>
        {
            proBarText.Text = "exposing";
            lblRealTime.Content += "Result = " + Variables.Result.ToString() + ",
Time - " +
SxSDKAccess.sxGetTimer(Variables.cameraImageData[Variables.selectedCamera].camHandle).
ToString() + "\r\n";
        }));


        // Start the exposure checking timer
        if (!waitForExposureEnd)
        {
            worker4.RunWorkerAsync();
        }
        else        // Short exposure is already completed so read image now
        {
            if (Variables.cameraInfo[Variables.selectedCamera].hasShutter)

SxSDKAccess.sxSetShutter(Variables.camHandles[Variables.selectedCamera], 1);
            Exposure.getImage(Variables.Result);
        }
}

private void Worker2_RunWorkerCompleted(object sender, RunWorkerCompletedEventArgs e)
{
    if(Variables.continues)
    {
        worker1.RunWorkerAsync();
    }
    Dispatcher.Invoke(new Action(() =>
    {
        //img.Source = Variables.bitmap;
        if(Variables.continueDraw)
        {
            Backplane.Visibility = Visibility.Hidden;
            btnCanvas.Content = "Show";
        }
        else
        {
            Backplane.Visibility = Visibility.Visible;
            btnCanvas.Content = "Hide";
        }
        img.Source = Variables.bitmap;
        //img.ImageSource = Variables.bitmap;
        btnOneExposure.IsEnabled = true;
        btnConExposure.IsEnabled = true;
        proBar.Value = 0;
        proBar.IsIndeterminate = false;
        proBarText.Text = "done";
    }));
    Variables.abortExposure = false;
    if (Variables.continueDraw)
        selectedSpectra();
}

private void Worker2_DoWork(object sender, DoWorkEventArgs e)
{

```

```csharp
        DrawImage.panelImage_Paint();
}

private void Worker4_DoWork(object sender, DoWorkEventArgs e)
{
    Timers.expoTimer.Start();
}
// Takes exposure using information in sxExposurePack
private Int32 makeExposure(sxExposurePack exposureInfo, Boolean
waitForExposureCompleted)
{
    Int32 Result = 0;

    Result = SxSDKAccess.sxClearPixels(exposureInfo.camHandle, Variables.CCD_WIPE_ALL,
(UInt16)(exposureInfo.useGuider ? 1 : 0));
    SxSDKAccess.sxSetTimer(exposureInfo.camHandle, exposureInfo.exposureTime);

    if (waitForExposureCompleted)
    {
        Dispatcher.Invoke(new Action(() =>
        {
            proBarText.Text = "exposing";

        }));

        while (SxSDKAccess.sxGetTimer(exposureInfo.camHandle) > 0)
        {
            Dispatcher.Invoke(new Action(() =>
            {
                proBar.Value++;

            }));
        }
        Dispatcher.Invoke(new Action(() =>
        {
            proBarText.Text = "done";
            proBar.Value = 0;
        }));
    }

    return (Result);
}


private void btnStopExposure_Click(object sender, RoutedEventArgs e)
{
    proBarText.Text = "canceling";
    Variables.continues = false;
    Variables.abortExposure = true;
}

private void ComBoxBinY_SelectionChanged(object sender, SelectionChangedEventArgs e)
{
    switch (ComBoxBinY.SelectedIndex)
    {
        case 0:{Variables.yyBin = 1;
                break;}
        case 1:{Variables.yyBin = 2;
                break;}
        case 2:{Variables.yyBin = 3;
                break;}
        case 3:{Variables.yyBin = 4;
```

```csharp
                break;}
        case 4:{Variables.yyBin = 8;
                break;}
        case 5:{Variables.yyBin = 16;
                break;}

        default:
            break;
    }
}

private void ComBoxBinX_SelectionChanged_1(object sender, SelectionChangedEventArgs e)
{
    switch (ComBoxBinX.SelectedIndex)
    {
        case 0:{Variables.xxBin = 1;
                break;}
        case 1:{Variables.xxBin = 2;
                break;}
        case 2:{Variables.xxBin = 3;
                break;}
        case 3:{Variables.xxBin = 4;
                break;}
        case 4:{Variables.xxBin = 8;
                break;}
        case 5:{Variables.xxBin = 16;
                break;}
        default:
            break;
    }
}

private void Backplane_MouseLeftButtonDown(object sender, MouseButtonEventArgs e)
{
    double pixelWidth = img.Source.Width;
    double pixelHeight = img.Source.Height;
    Variables._anchorPoint.X = (int)(pixelWidth * e.GetPosition(Backplane).X /
Backplane.ActualWidth);//img.ActualWidth);
    Variables._anchorPoint.Y = (int)(pixelHeight * e.GetPosition(Backplane).Y /
Backplane.ActualHeight);//img.ActualHeight);
    Variables._drawPoint.X = (int)e.GetPosition(Backplane).X;
    Variables._drawPoint.Y = (int)e.GetPosition(Backplane).Y;
    Variables._isDragging = true;
}

private void Backplane_MouseLeftButtonUp(object sender, MouseButtonEventArgs e)
{

    double pixelWidth = img.Source.Width;
    double pixelHeight = img.Source.Height;
    Variables.x = (int)(pixelWidth * e.GetPosition(Backplane).X /
Backplane.ActualWidth);//img.ActualWidth);
    Variables.y = (int)(pixelHeight * e.GetPosition(Backplane).Y /
Backplane.ActualHeight);//img.ActualHeight);
    chcBoxCont.IsEnabled = true;
    Variables.matName = testBox.Text;
    selectedSpectra();
    ResetRect();
}

private void Backplane_MouseLeave(object sender, MouseEventArgs e)
{
```

```csharp
        ResetRect();
    }

    private void ResetRect()
    {
        Rect.Visibility = Visibility.Collapsed;
        Variables._isDragging = false;
    }

    private void Backplane_MouseMove(object sender, MouseEventArgs e)
    {
        if (Variables._isDragging)
        {
            Variables.x = e.GetPosition(Backplane).X;
            Variables.y = e.GetPosition(Backplane).Y;

            Rect.SetValue(Canvas.LeftProperty, Math.Min(Variables.x,
Variables._drawPoint.X));
            Rect.SetValue(Canvas.TopProperty, Math.Min(Variables.y,
Variables._drawPoint.Y));

            Rect.Width = Math.Abs(Variables.x - Variables._drawPoint.X);
            Rect.Height = Math.Abs(Variables.y - Variables._drawPoint.Y);

            Variables.rWidth = (int)Rect.Width;
            Variables.rHeight = (int)Rect.Height;


            if (Rect.Visibility != Visibility.Visible)
                Rect.Visibility = Visibility.Visible;
        }
    }

    private void slidGrating_ValueChanged(object sender,
RoutedPropertyChangedEventArgs<double> e)
    {
        d = (UInt16)slidGrating.Value;
    }

    private void btnMove_Click(object sender, RoutedEventArgs e)
    {
    }

    private void btnCCD_Click(object sender, RoutedEventArgs e)
    {
        Variables.ToggleRedLEDPending = true;
    }

    private void Backplane_MouseRightButtonDown(object sender, MouseButtonEventArgs e)
    {

    }

    private void btnStopMove_Click(object sender, RoutedEventArgs e)
    {
        if (!Variables.btnStopGratingActive)
        {
            Variables.ToggleYellowLEDPending = true;
            btnStopMove.Content = "GO TO";
            btnStopMove.Background = (SolidColorBrush)(new
BrushConverter().ConvertFrom("#045757"));
            Variables.btnStopGratingActive = true;
```

```csharp
                proMoving.IsIndeterminate = false;

        }
        else
        {
            Variables.ToggleGratingPending = true;
            Variables.isMoving = true;
            btnStopMove.Content = "STOP";
            btnStopMove.Background = Brushes.Red;
            Variables.btnStopGratingActive = false;
            if (d < Variables.helpGov)
                proMoving.FlowDirection = FlowDirection.RightToLeft;
            else
                proMoving.FlowDirection = FlowDirection.LeftToRight;
            proMoving.IsIndeterminate = true;
        }

}
byte[] INBuffer = new byte[65];      //Allocate a memory buffer equal to the IN
endpoint size + 1
uint BytesRead = 0;
private void gratingTimer_Tick(object sender, EventArgs e)
{
    ReadFileManagedBuffer(ReadHandleToUSBDevice, INBuffer, 65, ref BytesRead,
IntPtr.Zero);
    UInt16 gov = (UInt16)((INBuffer[2] << 8) | INBuffer[1]);
    System.Diagnostics.Trace.WriteLine(gov);
}

private void btnCanvas_Click(object sender, RoutedEventArgs e)
{
    if(Backplane.IsVisible == true)
    {
        btnCanvas.Content = "Show";
        Backplane.Visibility = Visibility.Hidden;
    }
    else
    {
        btnCanvas.Content = "Hide";
        Backplane.Visibility = Visibility.Visible;
    }

}

private void btnDarkFrame_Click(object sender, RoutedEventArgs e)
{
    if(!Variables.darkFrame)
    {
        Variables.darkShutter = true;
        Variables.darkFrame = true;
        Variables.darkFrameSubtract = true;
        btnDarkFrame.Content = "Dark Frame is ON";
        btnDarkFrame.Background = Brushes.Red;
    }
    else
    {
        Variables.darkShutter = true;
        Variables.darkFrame = false;
        btnDarkFrame.Content = "Dark Frame is OFF";
        btnDarkFrame.Background = (SolidColorBrush)(new
BrushConverter().ConvertFrom("#045757"));
    }
```

```csharp
    }

    private void btnSaveData_Click(object sender, RoutedEventArgs e)
    {
        string sName = testBox.Text;
        string sLaser = "532";
        string sGrating = textGrating.Text;
        string sZF = "2";
        string sBin = Variables.xxBin.ToString() + "x" + Variables.yyBin.ToString();
        string sExpoTime = Variables.timerValue.ToString();
        string sImage = sName + ".tiff";
        Spectra spectra = new Spectra(sName, sLaser, sGrating, sZF, sBin, sExpoTime,
sImage);
        var dlg = new FolderBrowserForWPF.Dialog();
        dlg.Title = "Select folder";
        if (dlg.ShowDialog() == true)
        {

            Variables.saveFolder = dlg.FileName;
            Admin.saveSpectraToFile(spectra);
        }

    }

    private void chcBoxCont_Checked(object sender, RoutedEventArgs e)
    {
        Variables.continueDraw = true;
    }

    private void chcBoxCont_Unchecked(object sender, RoutedEventArgs e)
    {
        Variables.continueDraw = false;
    }

    private void chcRemoveDark_Checked(object sender, RoutedEventArgs e)
    {
        Variables.removeDark = true;
    }

    private void chcRemoveDark_Unchecked(object sender, RoutedEventArgs e)
    {
        Variables.removeDark = false;
    }

    private void Thumb_DragDelta(object sender,
System.Windows.Controls.Primitives.DragDeltaEventArgs e)
    {

        var thumb = sender as Thumb;
        var p = 0;
        Canvas.SetTop(thumb, Canvas.GetTop(thumb) + e.VerticalChange);
        p = (int)e.VerticalChange;

    }

    private void masterBiasFrame_Checked(object sender, RoutedEventArgs e)
    {
        Variables.masterBiasFrame = true;
    }

    private void masterBiasFrame_Unchecked(object sender, RoutedEventArgs e)
```

```csharp
{
    Variables.masterBiasFrame = false;
}

private void masterFlatFrame_Checked(object sender, RoutedEventArgs e)
{
    Variables.masterFlatFrame = true;
}

private void masterFlatFrame_Unchecked(object sender, RoutedEventArgs e)
{
    Variables.masterFlatFrame = false;
}

private void masterBiasGen_Checked(object sender, RoutedEventArgs e)
{
    Variables.masterBiasGen = true;
}

private void masterBiasGen_Unchecked(object sender, RoutedEventArgs e)
{
    Variables.masterBiasGen = false;
}

private void masterBiassubfromDark_Checked(object sender, RoutedEventArgs e)
{
    Variables.masterBiassubfromDark = true;
}

private void masterBiassubfromDark_Unchecked(object sender, RoutedEventArgs e)
{
    Variables.masterBiassubfromDark = false;
}

private void masterFlatMax_Checked(object sender, RoutedEventArgs e)
{
    Variables.masterFlatMax = true;
}

private void masterFlatMax_Unchecked(object sender, RoutedEventArgs e)
{
    Variables.masterFlatMax = false;
}

private void flatDivide_Checked(object sender, RoutedEventArgs e)
{
    Variables.flatDivide = true;
}

private void flatDivide_Unchecked(object sender, RoutedEventArgs e)
{
    Variables.flatDivide = false;
}

private void ComBoxLaserType_SelectionChanged(object sender, SelectionChangedEventArgs
e)
{
    switch (ComBoxLaserType.SelectedIndex)
    {
        case 0:
            slidGrating.Value = 47235;
            Variables.ToggleGratingPending = true;
```

```csharp
                Variables.isMoving = true;
                break;
            case 1:
                break;
        }
    }
}

public static void selectedSpectra()
{

    UInt32 suma = 0, k = 0;

    Variables.pixNumBuf2 = 0;
    Variables.bmpBuffer2 = new UInt32[(Variables.imageHeight / Variables.yyBin) *
(Variables.imageWidth / Variables.xxBin)];

    for (int i = (int)Variables._anchorPoint.X; i <= Variables.x; i++)
    {
        for (int j = (int)Variables._anchorPoint.Y; j <= Variables.y; j++)
        {
            suma = suma +
Variables.bmpBuffer[(j*(Variables.imageWidth/Variables.xxBin))+i];
            k++;
        }
        Variables.bmpBuffer2[Variables.pixNumBuf2++] = (UInt32)suma;
        suma = 0;
        k = 0;
    }
    DrawSpectra.drawGraph(Variables.pixNumBuf2);
    Variables.flagImage = true;
}
}
}
```

Draw Image CS file

```csharp
public class DrawImage
    {
        public static MainWindow mainWindow;

        public DrawImage(MainWindow window)
        {
            mainWindow = window;
        }
        public static void panelImage_Paint()
        {
            if (Variables.imageFlag)
            {
                if (!Variables.imageMemFlag)
                {
                    var width =
Variables.cameraInfo[Variables.selectedCamera].imageWidth / Variables.xxBin;
                    var height =
Variables.cameraInfo[Variables.selectedCamera].imageHeight / Variables.yyBin;
                    Variables.stride = 2 * width;
                    int i, j, k, index, l;
                    Int32 numBytes = width * height;
                    var pixelFormat = PixelFormats.Gray16;
                    var dpiX = 96d;
                    var dpiY = 96d;
```

```csharp
                    Variables.bmpBuffer = new UInt16[numBytes];
                    Variables.bmpBuffer3 = new UInt16[numBytes];


                    ///////////////master bias frame/////////////////////////
                    /// expo lowest, average of 20 shoots
                    ///////////////////////////////////////////////
                    if (Variables.masterBiasFrame)
                    {
                        index = 0;      // index to Int16 pixels in pixelBuffer
                        k = 0;
                        l = 0;
                        for (i = 0; i < height; i++)
                        {
                            for (j = 0; j < width; j++)
                            {

Variables.bpmBufferMasterBias[l++]=Variables.bpmBufferMasterBias[k++] +
(Variables.pixelBuffer[index++]);
                            }
                        }
                        Variables.biasMasterControl++;
                    }
                    ///////////////master dark frame/////////////////////////////
                    // expo as real expo, average of 20 shoots
                    ///////////////////////////////////////////////////

                    ///////////////master substract dark frame///////////////
                    //// master dark - master bias
                    ///////////////////////////////////////////////////

                    ///////////////master flat frame/////////////////////////
                    //// expo 1 second, average of 20 shoots
                    //////////////////////////////////////
                    ///
                    else if (Variables.masterFlatFrame)
                    {
                        index = 0;      // index to Int16 pixels in pixelBuffer
                        k = 0;
                        l = 0;
                        for (i = 0; i < height; i++)
                        {
                            for (j = 0; j < width; j++)
                            {
                                Variables.bpmBufferMasterFlat[l++] =
Variables.bpmBufferMasterFlat[k++] + (Variables.pixelBuffer[index++]);
                            }
                        }
                        Variables.flatMasterControl++;
                    }

                    else if (Variables.darkFrame)
                    {
                        Variables.bmpBufferDark = new UInt16[numBytes];
                        index = 0;      // index to Int16 pixels in pixelBuffer
                        k = 0;

                        for (i = 0; i < height; i++)
                        {
                            for (j = 0; j < width; j++)
                            {
```

```csharp
                                Variables.bmpBufferDark[k++] =
(Variables.pixelBuffer[index++]);

                        }
                    }

                    if (Variables.masterBiasGen)
                    {
                        index = 0;        // index to Int16 pixels in pixelBuffer
                        k = 0;
                        l = 0;
                        for (i = 0; i < height; i++)
                        {
                            for (j = 0; j < width; j++)
                            {
                                Variables.bpmBufferMasterBias[l++] =
Variables.bpmBufferMasterBias[k++] / Variables.biasMasterControl;
                            }
                        }
                    }

                    if (Variables.masterBiassubfromDark)
                    {
                        index = 0;        // index to Int16 pixels in pixelBuffer
                        k = 0;
                        l = 0;
                        for (i = 0; i < height; i++)
                        {
                            for (j = 0; j < width; j++)
                            {
                                Variables.bmpBufferDark[k++] =
(UInt16)(Variables.bmpBufferDark[index++] - Variables.bpmBufferMasterBias[l++]);
                            }
                        }
                    }

                    Variables.imageFlag = false;
                    mainWindow.Dispatcher.Invoke(() =>
                    {

                        mainWindow.chcRemoveDark.IsChecked = true;
                    });

                }
                else
                {
                    if(!Variables.removeDark)
                    {
                        Array.Clear(Variables.bmpBufferDark, 0,
Variables.bmpBufferDark.Length);
                    }
                    index = 0;        // index to Int16 pixels in pixelBuffer
                    k = 0;
                    l = 0;
                    for (i = 0; i < height; i++)
                    {
                        for (j = 0; j < width; j++)
                        {

                            if(Variables.darkFrameSubtract)
                            {
```

```csharp
                                    if (Variables.pixelBuffer[index++] >=
Variables.bmpBufferDark[l++])
                                        Variables.bmpBuffer[k++] =
(UInt16)(Variables.pixelBuffer[index-1] - Variables.bmpBufferDark[l-1]);
                                    else
                                        Variables.bmpBuffer[k++] = 0;
                                }
                                else
                                {
                                    Variables.bmpBuffer[k++] =
(Variables.pixelBuffer[index++]);
                                }
                            }
                        }

                        //Master flat Maximum
                        if (Variables.masterFlatMax)
                        {
                            index = 0;      // index to Int16 pixels in pixelBuffer
                            k = 0;
                            l = 0;
                            double max = Variables.bpmBufferMasterFlat[0];
                            for (i = 0; i < height; i++)
                            {
                                for (j = 0; j < width; j++)
                                {
                                    if (Variables.bpmBufferMasterFlat[index++] > max)
                                    {
                                        max = Variables.bpmBufferMasterFlat[k++];
                                    }
                                }
                            }

                            index = 0;      // index to Int16 pixels in pixelBuffer
                            k = 0;
                            l = 0;
                            for (i = 0; i < height; i++)
                            {
                                for (j = 0; j < width; j++)
                                {

                                    Variables.bpmBufferMasterFlat[index++] =
Variables.bpmBufferMasterFlat[k++] / max;

                                }
                            }
                        }

                        // divide flat from Image
                        if (Variables.flatDivide)
                        {
                            index = 0;      // index to Int16 pixels in pixelBuffer
                            k = 0;
                            l = 0;
                            for (i = 0; i < height; i++)
                            {
                                for (j = 0; j < width; j++)
                                {

                                    Variables.bmpBuffer[k++] =
(UInt16)(Variables.bmpBuffer[l++] / Variables.bpmBufferMasterFlat[index++]);
```

```
                    }
                }
            }

            Variables.bitmap = BitmapSource.Create(width, height, dpiX,
dpiY, pixelFormat, null, Variables.bmpBuffer, Variables.stride);
            Variables.bitmap.Freeze();


            using (var fileStream = new FileStream(@Variables.privremeni +
"image.tiff", FileMode.Create, FileAccess.ReadWrite))
            {
                BitmapEncoder encoder = new TiffBitmapEncoder();
                encoder.Frames.Add(BitmapFrame.Create(Variables.bitmap));
                encoder.Save(fileStream);
            }

            mainWindow.Dispatcher.Invoke(() =>
            {

                mainWindow.proBar.Value = 100;
                //Variables.winImage.img.ImageSource = Variables.bitmap;
            });
            Variables.imageFlag = false;

            if (Variables.flagImage)
            {
                // DrawSpectra.drawGraphFull();
            }
        }
        else
        {
        }

    }
  }
 }
```

Draw Spectra CS file

```
class Spectra
{
        string sName, sLaser, sGrating, sZF, sBin, sExpoTime, sX, sY, sImage;

        public string SName
        {
            get { return sName; }
        }
        public string SLaser
        {
            get { return sLaser; }
        }
        public string SGrating
        {
            get { return sGrating; }
        }
        public string SZF
        {
            get { return sZF; }
        }
        public string SBin
        {
```

```csharp
            get { return sBin; }
        }
        public string SExpoTime
        {
            get { return sExpoTime; }
        }

        public string SImage
        {
            get { return sImage; }
        }


    public Spectra (string sname, string slaser, string sgrating, string szf, string sbin,
    string sexpotime, string simage)
        {
            sName = sname;
            sLaser = slaser;
            sGrating = sgrating;
            sZF = szf;
            sBin = sbin;
            sExpoTime = sexpotime;
            sImage = simage;
        }

    }
```

APPENDIX F: C FIRMWARE SOURCE CODE

Main system configuration

```
/** CONFIGURATION Bits **********************************************/

#pragma config PLLSEL = PLL3X // PLL Selection (3x clock multiplier)
#pragma config CFGPLLEN = OFF // PLL Enable Configuration bit (PLL Disabled (firmware
controlled))
#pragma config CPUDIV = NOCLKDIV // CPU System Clock Postscaler (CPU uses system clock
(no divide))
#pragma config LS48MHZ = SYS48X8 // Low Speed USB mode with 48 MHz system clock (System
clock at 48 MHz, USB clock divider is set to 8)
#pragma config FOSC = HSH //INTOSCIO  // Oscillator Selection (Internal oscillator)
#pragma config PCLKEN = OFF // Primary Oscillator Shutdown (Primary oscillator shutdown
firmware controlled)
#pragma config FCMEN = OFF // Fail-Safe Clock Monitor (Fail-Safe Clock Monitor disabled)
#pragma config IESO = OFF // Internal/External Oscillator Switchover (Oscillator
Switchover mode disabled)
#pragma config nPWRTEN = OFF // Power-up Timer Enable (Power up timer disabled)
#pragma config BOREN = SBORDIS // Brown-out Reset Enable (BOR enabled in hardware (SBOREN
is ignored))
#pragma config BORV = 190 // Brown-out Reset Voltage (BOR set to 1.9V nominal)
#pragma config nLPBOR = ON // Low-Power Brown-out Reset (Low-Power Brown-out Reset
enabled)
#pragma config WDTEN = SWON // Watchdog Timer Enable bits (WDT controlled by firmware
(SWDTEN enabled))
#pragma config WDTPS = 32768 // Watchdog Timer Postscaler (1:32768)
#pragma config CCP2MX = RC1 // CCP2 MUX bit (CCP2 input/output is multiplexed with RC1)
#pragma config PBADEN  = OFF // PORTB A/D Enable bit (PORTB<5:0> pins are configured
as digital I/O on Reset)
#pragma config T3CMX = RC0 // Timer3 Clock Input MUX bit (T3CKI function is on RC0)
#pragma config SDOMX = RC7 // SDO Output MUX bit (SDO function is on RC7)
#pragma config MCLRE = ON // Master Clear Reset Pin Enable (MCLR pin enabled; RE3 input
disabled)
#pragma config STVREN = ON // Stack Full/Underflow Reset (Stack full/underflow will
cause Reset)
#pragma config LVP = OFF // Single-Supply ICSP Enable bit (Single-Supply ICSP disabled)
#pragma config ICPRT = OFF // Dedicated In-Circuit Debug/Programming Port Enable (ICPORT
disabled)
#pragma config XINST = OFF // Extended Instruction Set Enable bit (Instruction set
extension and Indexed Addressing mode disabled)
#pragma config CP0 = OFF // Block 0 Code Protect (Block 0 is not code-protected)
#pragma config CP1 = OFF // Block 1 Code Protect (Block 1 is not code-protected)
#pragma config CP2 = OFF // Block 2 Code Protect (Block 2 is not code-protected)
#pragma config CP3 = OFF // Block 3 Code Protect (Block 3 is not code-protected)
#pragma config CPB = OFF // Boot Block Code Protect (Boot block is not code-protected)
#pragma config CPD = OFF  // Data EEPROM Code Protect (Data EEPROM is not code-protected)
#pragma config WRT0 = OFF // Block 0 Write Protect (Block 0 (0800-1FFFh) is not write-
protected)
#pragma config WRT1 = OFF // Block 1 Write Protect (Block 1 (2000-3FFFh) is not write-
protected)
#pragma config WRT2 = OFF // Block 2 Write Protect (Block 2 (04000-5FFFh) is not write-
protected)
#pragma config WRT3 = OFF // Block 3 Write Protect (Block 3 (06000-7FFFh) is not write-
protected)
#pragma config WRTC = OFF // Configuration Registers Write Protect (Configuration
registers (300000-3000FFh) are not write-protected)
#pragma config WRTB    = OFF // Boot Block Write Protect (Boot block (0000-7FFh) is not
write-protected)
```

```
#pragma config WRTD = OFF // Data EEPROM Write Protect (Data EEPROM is not write-
protected)
#pragma config EBTR0 = OFF // Block 0 Table Read Protect (Block 0 is not protected from
table reads executed in other blocks)
#pragma config EBTR1  = OFF // Block 1 Table Read Protect (Block 1 is not protected from
table reads executed in other blocks)
#pragma config EBTR2 = OFF // Block 2 Table Read Protect (Block 2 is not protected from
table reads executed in other blocks)
#pragma config EBTR3 = OFF  // Block 3 Table Read Protect (Block 3 is not protected from
table reads executed in other blocks)
#pragma config EBTRB = OFF // Boot Block Table Read Protect (Boot block is not protected
from table reads executed in other blocks)
```

PIN configuration

```
#define PIN_M1_LAT LATEbits.LATE0
#define PIN_M2_LAT LATEbits.LATE1
#define PIN_M3_LAT LATEbits.LATE2
#define PIN_LED_1_LAT LATDbits.LATD0
#define PIN_LED_2_LAT LATDbits.LATD1
#define PIN_LED_3_LAT LATDbits.LATD2
#define PIN_LED_4_LAT LATDbits.LATD3
#define PIN_LED_5_LAT LATDbits.LATD4
#define PIN_P1B_LAT LATDbits.LATD5
#define PIN_P1C_LAT LATDbits.LATD6
#define PIN_P1D_LAT LATDbits.LATD7
#define PIN_RC0_LAT LATCbits.LATC0
#define PIN_CLK_LAT LATCbits.LATC1
#define PIN_P1A_LAT LATCbits.LATC2
#define PIN_SENS_LAT LATBbits.LATB0
#define PIN_RESET_LAT LATBbits.LATB1
#define PIN_SYNC_LAT LATBbits.LATB2
#define PIN_DIR_LAT LATBbits.LATB3
#define PIN_RB4_LAT LATBbits.LATB4
#define PIN_RB5_LAT LATBbits.LATB5
#define PIN_RA0_LAT LATAbits.LATA0
#define PIN_RA1_LAT LATAbits.LATA1
#define PIN_RA2_LAT LATAbits.LATA2
#define PIN_RA3_LAT LATAbits.LATA3
#define PIN_RA4_LAT LATAbits.LATA4
#define PIN_RA5_LAT LATAbits.LATA5

#define PIN_M1_TRIS TRISEbits.TRISE0
#define PIN_M2_TRIS TRISEbits.TRISE1
#define PIN_M3_TRIS TRISEbits.TRISE2
#define PIN_LED_1_TRIS TRISDbits.TRISD0
#define PIN_LED_2_TRIS TRISDbits.TRISD1
#define PIN_LED_3_TRIS TRISDbits.TRISD2
#define PIN_LED_4_TRIS TRISDbits.TRISD3
#define PIN_LED_5_TRIS TRISDbits.TRISD4
#define PIN_P1B_TRIS TRISDbits.TRISD5
#define PIN_P1C_TRIS TRISDbits.TRISD6
#define PIN_P1D_TRIS TRISDbits.TRISD7
#define PIN_RC0_TRIS TRISCbits.TRISC0
#define PIN_CLK_TRIS TRISCbits.TRISC1
#define PIN_P1A_TRIS TRISCbits.TRISC2
#define PIN_SENS_TRIS TRISBbits.TRISB0
#define PIN_RESET_TRIS TRISBbits.TRISB1
#define PIN_SYNC_TRIS TRISBbits.TRISB2
#define PIN_DIR_TRIS TRISBbits.TRISB3
#define PIN_RB4_TRIS TRISBbits.TRISB4
```

```c
#define PIN_RB5_TRIS TRISBbits.TRISB5
#define PIN_RA0_TRIS TRISAbits.TRISA0
#define PIN_RA1_TRIS TRISAbits.TRISA1
#define PIN_RA2_TRIS TRISAbits.TRISA2
#define PIN_RA3_TRIS TRISAbits.TRISA3
#define PIN_RA4_TRIS TRISAbits.TRISA4
#define PIN_RA5_TRIS TRISAbits.TRISA5


#define PIN_M1_ANSEL ANSELEbits.ANSE0
#define PIN_M2_ANSEL ANSELEbits.ANSE1
#define PIN_M3_ANSEL ANSELEbits.ANSE2
#define PIN_LED_1_ANSEL ANSELDbits.ANSD0
#define PIN_LED_2_ANSEL ANSELDbits.ANSD1
#define PIN_LED_3_ANSEL ANSELDbits.ANSD2
#define PIN_LED_4_ANSEL ANSELDbits.ANSD3
#define PIN_LED_5_ANSEL ANSELDbits.ANSD4
#define PIN_P1B_ANSEL ANSELDbits.ANSD5
#define PIN_P1C_ANSEL ANSELDbits.ANSD6
#define PIN_P1D_ANSEL ANSELDbits.ANSD7
#define PIN_P1A_ANSEL ANSELCbits.ANSC2
#define PIN_SENS_ANSEL ANSELBbits.ANSB0
#define PIN_RESET_ANSEL ANSELBbits.ANSB1
#define PIN_SYNC_ANSEL ANSELBbits.ANSB2
#define PIN_DIR_ANSEL ANSELBbits.ANSB3
#define PIN_RB4_ANSEL ANSELBbits.ANSB4
#define PIN_RB5_ANSEL ANSELBbits.ANSB5
#define PIN_RA0_ANSEL ANSELAbits.ANSA0
#define PIN_RA1_ANSEL ANSELAbits.ANSA1
#define PIN_RA2_ANSEL ANSELAbits.ANSA2
#define PIN_RA3_ANSEL ANSELAbits.ANSA3
#define PIN_RA5_ANSEL ANSELAbits.ANSA5

#define PIN_ON   1
#define PIN_OFF 0

#define PIN_INPUT           1
#define PIN_OUTPUT          0

#define PIN_DIGITAL         0
#define PIN_ANALOG          1

void PIN_On(PIN pin)
{
    switch(pin)
    {
        case PIN_M1:
            PIN_M1_LAT = PIN_ON;
            break;

        case PIN_M2:
            PIN_M2_LAT = PIN_ON;
            break;

        case PIN_M3:
            PIN_M3_LAT = PIN_ON;
            break;

        case PIN_LED_1:
            PIN_LED_1_LAT = PIN_ON;
            break;

        case PIN_LED_2:
```

```
        PIN_LED_2_LAT = PIN_ON;
        break;

    case PIN_LED_3:
        PIN_LED_3_LAT = PIN_ON;
        break;

    case PIN_LED_4:
        PIN_LED_4_LAT = PIN_ON;
        break;

    case PIN_LED_5:
        PIN_LED_5_LAT = PIN_ON;
        break;

    case PIN_P1A:
        PIN_P1A_LAT = PIN_ON;
        break;

    case PIN_P1B:
        PIN_P1B_LAT = PIN_ON;
        break;

    case PIN_P1C:
        PIN_P1C_LAT = PIN_ON;
        break;

    case PIN_P1D:
        PIN_P1D_LAT = PIN_ON;
        break;

    case PIN_RC0:
        PIN_RC0_LAT = PIN_ON;
        break;

    case PIN_CLK:
        PIN_CLK_LAT = PIN_ON;
        break;

    case PIN_RB4:
        PIN_RB4_LAT = PIN_ON;
        break;

    case PIN_RB5:
        PIN_RB5_LAT = PIN_ON;
        break;

    case PIN_DIR:
        PIN_DIR_LAT = PIN_ON;
        break;

    case PIN_SYNC:
        PIN_SYNC_LAT = PIN_ON;
        break;

    case PIN_RESET:
        PIN_RESET_LAT = PIN_ON;
        break;

    case PIN_SENS:
        PIN_SENS_LAT = PIN_ON;
        break;
```

```c
        case PIN_RA0:
            PIN_RA0_LAT = PIN_ON;
            break;

        case PIN_RA1:
            PIN_RA1_LAT = PIN_ON;
            break;

        case PIN_RA2:
            PIN_RA2_LAT = PIN_ON;
            break;

        case PIN_RA3:
            PIN_RA3_LAT = PIN_ON;
            break;

        case PIN_RA4:
            PIN_RA4_LAT = PIN_ON;
            break;

        case PIN_RA5:
            PIN_RA5_LAT = PIN_ON;
            break;
    }
}

void PIN_Off(PIN pin)
{
    switch(pin)
    {
        case PIN_M1:
            PIN_M1_LAT = PIN_OFF;
            break;

        case PIN_M2:
            PIN_M2_LAT = PIN_OFF;
            break;

        case PIN_M3:
            PIN_M3_LAT = PIN_OFF;
            break;

        case PIN_LED_1:
            PIN_LED_1_LAT = PIN_OFF;
            break;

        case PIN_LED_2:
            PIN_LED_2_LAT = PIN_OFF;
            break;

        case PIN_LED_3:
            PIN_LED_3_LAT = PIN_OFF;
            break;

        case PIN_LED_4:
            PIN_LED_4_LAT = PIN_OFF;
            break;

        case PIN_LED_5:
            PIN_LED_5_LAT = PIN_OFF;
            break;
```

```
case PIN_P1A:
    PIN_P1A_LAT = PIN_OFF;
    break;

case PIN_P1B:
    PIN_P1B_LAT = PIN_OFF;
    break;

case PIN_P1C:
    PIN_P1C_LAT = PIN_OFF;
    break;

case PIN_P1D:
    PIN_P1D_LAT = PIN_OFF;
    break;

case PIN_RC0:
    PIN_RC0_LAT = PIN_OFF;
    break;

case PIN_CLK:
    PIN_CLK_LAT = PIN_OFF;
    break;

case PIN_RB4:
    PIN_RB4_LAT = PIN_OFF;
    break;

case PIN_RB5:
    PIN_RB5_LAT = PIN_OFF;
    break;

case PIN_DIR:
    PIN_DIR_LAT = PIN_OFF;
    break;

case PIN_SYNC:
    PIN_SYNC_LAT = PIN_OFF;
    break;

case PIN_RESET:
    PIN_RESET_LAT = PIN_OFF;
    break;

case PIN_SENS:
    PIN_SENS_LAT = PIN_OFF;
    break;

case PIN_RA0:
    PIN_RA0_LAT = PIN_OFF;
    break;

case PIN_RA1:
    PIN_RA1_LAT = PIN_OFF;
    break;

case PIN_RA2:
    PIN_RA2_LAT = PIN_OFF;
    break;

case PIN_RA3:
```

```c
                PIN_RA3_LAT = PIN_OFF;
                break;

            case PIN_RA4:
                PIN_RA4_LAT = PIN_OFF;
                break;

            case PIN_RA5:
                PIN_RA5_LAT = PIN_OFF;
                break;
        }
    }

    void PIN_Toggle(PIN pin)
    {
        switch(pin)
        {
            case PIN_M1:
                PIN_M1_LAT ^= 1;
                break;

            case PIN_M2:
                PIN_M2_LAT ^= 1;
                break;

            case PIN_M3:
                PIN_M3_LAT ^= 1;
                break;

            case PIN_LED_1:
                PIN_LED_1_LAT ^= 1;
                break;

            case PIN_LED_2:
                PIN_LED_2_LAT ^= 1;
                break;

            case PIN_LED_3:
                PIN_LED_3_LAT ^= 1;
                break;

            case PIN_LED_4:
                PIN_LED_4_LAT ^= 1;
                break;

            case PIN_LED_5:
                PIN_LED_5_LAT ^= 1;
                break;

            case PIN_P1A:
                PIN_P1A_LAT ^= 1;
                break;

            case PIN_P1B:
                PIN_P1B_LAT ^= 1;
                break;

            case PIN_P1C:
                PIN_P1C_LAT ^= 1;
                break;

            case PIN_P1D:
```

```c
            PIN_P1D_LAT ^= 1;
            break;

        case PIN_RC0:
            PIN_RC0_LAT ^= 1;
            break;

        case PIN_CLK:
            PIN_CLK_LAT ^= 1;
            break;

        case PIN_RB4:
            PIN_RB4_LAT ^= 1;
            break;

        case PIN_RB5:
            PIN_RB5_LAT ^= 1;
            break;

        case PIN_DIR:
            PIN_DIR_LAT ^= 1;
            break;

        case PIN_SYNC:
            PIN_SYNC_LAT ^= 1;
            break;

        case PIN_RESET:
            PIN_RESET_LAT ^= 1;
            break;

        case PIN_SENS:
            PIN_SENS_LAT ^= 1;
            break;

        case PIN_RA0:
            PIN_RA0_LAT ^= 1;
            break;

        case PIN_RA1:
            PIN_RA1_LAT ^= 1;
            break;

        case PIN_RA2:
            PIN_RA2_LAT ^= 1;
            break;

        case PIN_RA3:
            PIN_RA3_LAT ^= 1;
            break;

        case PIN_RA4:
            PIN_RA4_LAT ^= 1;
            break;

        case PIN_RA5:
            PIN_RA5_LAT ^= 1;
            break;
    }
}
```

```c
bool PIN_Get(PIN pin)
{
    switch(pin)
    {
        case PIN_M1:
            return ( (PIN_M1_LAT == PIN_ON) ? true : false );

        case PIN_M2:
            return ( (PIN_M2_LAT == PIN_ON) ? true : false );

        case PIN_M3:
            return ( (PIN_M3_LAT == PIN_ON) ? true : false );

        case PIN_LED_1:
            return ( (PIN_LED_1_LAT == PIN_ON) ? true : false );

        case PIN_LED_2:
            return ( (PIN_LED_2_LAT == PIN_ON) ? true : false );

        case PIN_LED_3:
            return ( (PIN_LED_3_LAT == PIN_ON) ? true : false );

        case PIN_LED_4:
            return ( (PIN_LED_4_LAT == PIN_ON) ? true : false );

        case PIN_LED_5:
            return ( (PIN_LED_5_LAT == PIN_ON) ? true : false );

        case PIN_P1A:
            return ( (PIN_P1A_LAT == PIN_ON) ? true : false );

        case PIN_P1B:
            return ( (PIN_P1B_LAT == PIN_ON) ? true : false );

        case PIN_P1C:
            return ( (PIN_P1C_LAT == PIN_ON) ? true : false );

        case PIN_P1D:
            return ( (PIN_P1D_LAT == PIN_ON) ? true : false );

        case PIN_RC0:
            return ( (PIN_RC0_LAT == PIN_ON) ? true : false );

        case PIN_CLK:
            return ( (PIN_CLK_LAT == PIN_ON) ? true : false );

        case PIN_RB4:
            return ( (PIN_RB4_LAT == PIN_ON) ? true : false );

        case PIN_RB5:
            return ( (PIN_RB5_LAT == PIN_ON) ? true : false );

        case PIN_DIR:
            return ( (PIN_DIR_LAT == PIN_ON) ? true : false );

        case PIN_SYNC:
            return ( (PIN_SYNC_LAT == PIN_ON) ? true : false );

        case PIN_RESET:
            return ( (PIN_RESET_LAT == PIN_ON) ? true : false );

        case PIN_SENS:
```

```c
                return ( (PIN_SENS_LAT == PIN_ON) ? true : false );

        case PIN_RA0:
            return ( (PIN_RA0_LAT == PIN_ON) ? true : false );

        case PIN_RA1:
            return ( (PIN_RA1_LAT == PIN_ON) ? true : false );

        case PIN_RA2:
            return ( (PIN_RA2_LAT == PIN_ON) ? true : false );

        case PIN_RA3:
            return ( (PIN_RA3_LAT == PIN_ON) ? true : false );

        case PIN_RA4:
            return ( (PIN_RA4_LAT == PIN_ON) ? true : false );

        case PIN_RA5:
            return ( (PIN_RA5_LAT == PIN_ON) ? true : false );
    }

    return false;
}

void PIN_Enable(PIN pin)
{
    switch(pin)
    {
        case PIN_M1:
            PIN_M1_TRIS = PIN_OUTPUT;
            PIN_M1_ANSEL = PIN_DIGITAL;
            break;

        case PIN_M2:
            PIN_M2_TRIS = PIN_OUTPUT;
            PIN_M2_ANSEL = PIN_DIGITAL;
            break;

        case PIN_M3:
            PIN_M3_TRIS = PIN_OUTPUT;
            PIN_M3_ANSEL = PIN_DIGITAL;
            break;

        case PIN_LED_1:
            PIN_LED_1_TRIS = PIN_OUTPUT;
            PIN_LED_1_ANSEL = PIN_DIGITAL;
            break;

        case PIN_LED_2:
            PIN_LED_2_TRIS = PIN_OUTPUT;
            PIN_LED_2_ANSEL = PIN_DIGITAL;
            break;

        case PIN_LED_3:
            PIN_LED_3_TRIS = PIN_OUTPUT;
            PIN_LED_3_ANSEL = PIN_DIGITAL;
            break;

        case PIN_LED_4:
            PIN_LED_4_TRIS = PIN_OUTPUT;
            PIN_LED_4_ANSEL = PIN_DIGITAL;
            break;
```

```c
case PIN_LED_5:
    PIN_LED_5_TRIS = PIN_OUTPUT;
    PIN_LED_5_ANSEL = PIN_DIGITAL;
    break;

case PIN_P1A:
    PIN_P1A_TRIS = PIN_OUTPUT;
    PIN_P1A_ANSEL = PIN_DIGITAL;
    break;

case PIN_P1B:
    PIN_P1B_TRIS = PIN_OUTPUT;
    PIN_P1B_ANSEL = PIN_DIGITAL;
    break;

case PIN_P1C:
    PIN_P1C_TRIS = PIN_OUTPUT;
    PIN_P1C_ANSEL = PIN_DIGITAL;
    break;

case PIN_P1D:
    PIN_P1D_TRIS = PIN_OUTPUT;
    PIN_P1D_ANSEL = PIN_DIGITAL;
    break;

case PIN_RC0:
    PIN_RC0_TRIS = PIN_OUTPUT;
    break;

case PIN_CLK:
    PIN_CLK_TRIS = PIN_OUTPUT;
    break;

case PIN_RB4:
    PIN_RB4_TRIS = PIN_OUTPUT;
    PIN_RB4_ANSEL = PIN_DIGITAL;
    break;

case PIN_RB5:
    PIN_RB5_TRIS = PIN_OUTPUT;
    PIN_RB5_ANSEL = PIN_DIGITAL;
    break;

case PIN_DIR:
    PIN_DIR_TRIS = PIN_OUTPUT;
    PIN_DIR_ANSEL = PIN_DIGITAL;
    break;

case PIN_SYNC:
    PIN_SYNC_TRIS = PIN_OUTPUT;
    PIN_SYNC_ANSEL = PIN_DIGITAL;
    break;

case PIN_RESET:
    PIN_RESET_TRIS = PIN_OUTPUT;
    PIN_RESET_ANSEL = PIN_DIGITAL;
    break;

case PIN_SENS:
    PIN_SENS_TRIS = PIN_INPUT;
    PIN_SENS_ANSEL = PIN_DIGITAL;
```

```
                break;

        case PIN_RA0:
            PIN_RA0_TRIS = PIN_OUTPUT;
            PIN_RA0_ANSEL = PIN_DIGITAL;
            break;

        case PIN_RA1:
            PIN_RA1_TRIS = PIN_OUTPUT;
            PIN_RA1_ANSEL = PIN_DIGITAL;
            break;

        case PIN_RA2:
            PIN_RA2_TRIS = PIN_OUTPUT;
            break;

        case PIN_RA3:
            PIN_RA3_TRIS = PIN_OUTPUT;
            break;

        case PIN_RA4:
            PIN_RA4_TRIS = PIN_OUTPUT;
            break;

        case PIN_RA5:
            PIN_RA5_TRIS = PIN_OUTPUT;
            PIN_RA5_ANSEL = PIN_DIGITAL;
            break;

    }
}
```

Main file code

```
MAIN_RETURN main(void)
{
    SYSTEM_Initialize(SYSTEM_STATE_USB_START);
    TMR0_Initialize();
    TMR1_Initialize();
    USBDeviceInit();
    USBDeviceAttach();
    INTCONbits.GIE=1;//enable global interrupts
    INTCONbits.PEIE=1;//enable peripheral inter

    PWM1 = 54;
    PIN_Enable(PIN_LED_4);
    PIN_Enable(PIN_LED_5);
    PIN_Enable(PIN_M1);
    PIN_Enable(PIN_M2);
    PIN_Enable(PIN_M3);
    PIN_Enable(PIN_DIR);
    PIN_Enable(PIN_SYNC);
    PIN_Enable(PIN_RESET);
    PIN_Enable(PIN_CLK);
    PIN_Enable(PIN_SENS);
    PIN_Enable(PIN_RA0);
    PIN_Off(PIN_RA0);
    PIN_Enable(PIN_P1D);//SERVO III
    PIN_Enable(PIN_P1C); //SERVO II
    PIN_Enable(PIN_P1B); // SERVO I
    PWM1 = 125; // 10 i 54
    PWM2 = 125; // 10 i 54
```

```
    PWM3 = 125; // 10 i 54
    uint8_t i = 0;
    uint16_t stepp=0;

    PIN_On(PIN_RESET);
    PIN_Off(PIN_RESET);// reset sla
    PIN_Off(PIN_SYNC); // pwm sync sla
    PIN_Off(PIN_M1); //M1
    PIN_Off(PIN_M2); //M2
    PIN_Off(PIN_M3); //M3
    PIN_On(PIN_DIR); // direction

    while(PORTBbits.RB0==0)
    {

        PIN_Toggle(PIN_CLK);
          delayTMRs(1);//__delay_ms(1);//delayTMRs(2);
    }

    PIN_Off(PIN_DIR); // direction
    for(stepp=0; stepp<32767; stepp++)
    {
        PIN_Toggle(PIN_CLK);
        delayTMRs(1);//__delay_ms(1);//delayTMRs(2);
    }
    PIN_On(PIN_M1);//M1
    PIN_On(PIN_M2);//M2
    PIN_On(PIN_M3);//M3


    while(1)
    {
        SYSTEM_Tasks();

        #if defined(USB_POLLING)
            USBDeviceTasks();
        #endif

        //Application specific tasks
        APP_DeviceCustomHIDTasks();

    }//end while
}//end main
```

Interrupt code for motor movement

```
void INTERRUPT SYS_InterruptHigh(void)
{
    #if defined(USB_INTERRUPT)
        USBDeviceTasks();
    #endif
    // interrupt handler
    if(INTCONbits.TMR0IE == 1 && INTCONbits.TMR0IF == 1)
    {
        INTCONbits.TMR0IF = 0;
        TMR0L = 0x01;
         p++;                // Count pieces of seconds
           if(p == 12 )
           {
                p=0;
                brojac++;
                brojac2++;
```

```c
            if(brojac2 == 2)
            {
                if(check==1 && zbroj<kkk)
                {
                    PIN_Toggle(PIN_CLK);
                    kkk--;
                }
                else if (check == 2 && zbroj>kkk)
                {
                    PIN_Toggle(PIN_CLK);
                    kkk++;
                }
                else if (check == 0)
                {
                    PIN_On(PIN_M1);//M1
                    PIN_On(PIN_M2);//M2
                    PIN_On(PIN_M3);//M3
                }
                brojac2 = 0;
            }
            if(brojac==1000)
            {
               PIN_Toggle(PIN_LED_5);
                brojac = 0;
            }
        }
    }
}
else if(INTCONbits.PEIE == 1)
{
    if(PIE1bits.TMR1IE == 1 && PIR1bits.TMR1IF == 1)
    {
        TMR1 = 65067;
        PIR1bits.TMR1IF = 0;
        k++;
        if (k>=PWM1)
        {
            PIN_Off(PIN_P1B);
        }
        if (k>=PWM2)
        {
            PIN_Off(PIN_P1C);
        }
        if (k>=PWM3)
        {
            //PIN_Off(PIN_P1D);
        }
        if(k==512)
        {
            PIN_On(PIN_P1B);
            PIN_On(PIN_P1C);
            PIN_On(PIN_P1D);
            k = 0;
        }
    }
    else
    {
        //Unhandled Interrupt
    }
}
else
{
}}
```

Function for receiving data from USB

```c
void APP_DeviceCustomHIDTasks()
{
    if( USBGetDeviceState() < CONFIGURED_STATE )
    {
        return;
    }
    if( USBIsDeviceSuspended()== true )
    {
        return;
    }

    if(HIDRxHandleBusy(USBOutHandle) == false)
    {
        switch(ReceivedDataBuffer[0])
        {
            case 0x32:
                PIN_Toggle(PIN_RA0);
                break;
            case 0x90:
                if(!HIDTxHandleBusy(USBInHandle))
                {
                    ToSendDataBuffer[0] = 0x22;
                    ToSendDataBuffer[1] = (uint8_t)kkk; //LSB
                    ToSendDataBuffer[2] = kkk >> 8;     //MSB
                    USBInHandle = HIDTxPacket(CUSTOM_DEVICE_HID_EP,
                                          (uint8_t*)&ToSendDataBuffer[0],64);
                }
                USBInHandle = HIDTxPacket(CUSTOM_DEVICE_HID_EP,
                                          (uint8_t*)&ToSendDataBuffer[0],64);
                break;

            case 0x64:
                check = 0;
                break;

            case COMMAND_TOGGLE_LED:  //Toggle LEDs command
                PIN_Toggle(LED_USB_DEVICE_HID_CUSTOM);
                PWM1--;
                if(PWM1==10)
                {
                    PWM1=54;
                }
                break;

            case COMMAND_GET_BUTTON_STATUS:
                if(!HIDTxHandleBusy(USBInHandle))
                {
                    ToSendDataBuffer[0] = 0x81;
                    if(BUTTON_IsPressed(BUTTON_USB_DEVICE_HID_CUSTOM) == false)
                    {
                        ToSendDataBuffer[1] = 0x01;
                    }
                    else
                    {
                        ToSendDataBuffer[1] = 0x00;
                    }

                    USBInHandle = HIDTxPacket(CUSTOM_DEVICE_HID_EP,
                                          (uint8_t*)&ToSendDataBuffer[0],64);
```

```c
                }
                break;

            case COMMAND_READ_POTENTIOMETER:
                {
                    uint16_t pot;

                    if(!HIDTxHandleBusy(USBInHandle))
                    {
                        pot = ADC_Read10bit(ADC_CHANNEL_POTENTIOMETER);

                        ToSendDataBuffer[0] = 0x37;
                        ToSendDataBuffer[1] = (uint8_t)pot; //LSB
                        ToSendDataBuffer[2] = pot >> 8;      //MSB


                        USBInHandle = HIDTxPacket(CUSTOM_DEVICE_HID_EP,
                                            (uint8_t*)&ToSendDataBuffer[0],64);
                    }
                }
                break;

            case COMMAND_GRATTING:
                {

                    zbroj = (ReceivedDataBuffer[5]<<8)|ReceivedDataBuffer[4];
                    if(zbroj < kkk)
                    {
                        PIN_On(PIN_M1);//M1
                        PIN_On(PIN_M2);//M2
                        PIN_Off(PIN_M3);//M3
                        PIN_On(PIN_DIR); // direction
                        check = 1;
                    }
                    else
                    {
                        PIN_On(PIN_M1);//M1
                        PIN_On(PIN_M2);//M2
                        PIN_Off(PIN_M3);//M3
                        PIN_Off(PIN_DIR); // direction
                        check = 2;
                    }


                }
                break;
        }
        USBOutHandle = HIDRxPacket(CUSTOM_DEVICE_HID_EP,
                            (uint8_t*)&ReceivedDataBuffer[0], 64);
    }

}
```