WestVirginiaUniversity
THE RESEARCH REPOSITORY @ WVU

Graduate Theses, Dissertations, and Problem Reports

2023

# Longitudinal oxygen imaging in 3D (bio)printed models

Ryan Curtis O'Connell
rco0002@mix.wvu.edu

Follow this and additional works at: https://researchrepository.wvu.edu/etd

Part of the Biomedical Devices and Instrumentation Commons, and the Medical Biophysics Commons

## Recommended Citation

# LONGITUDINAL OXYGEN IMAGING IN 3D (BIO)PRINTED MODELS

Ryan C. O'Connell

Dissertation submitted to the
School of Medicine
at West Virginia University

in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in
Biochemistry and Molecular Medicine

Mark Tseytlin, Ph.D., Committee Chairperson
Timothy Eubank, Ph.D., Committee Co-Chair
Matthew Dietz, M.D.
Michael Gunther, Ph.D.
Ming Pei, M.D., Ph.D.
Elena Pugacheva, Ph.D.

Department of Biochemistry and Molecular Medicine

Morgantown, West Virginia
2023

# ABSTRACT

## Longitudinal Oxygen Imaging in 3D (Bio)printed Models

### by Ryan C. O'Connell

Electron paramagnetic resonance (EPR), and its molecular imaging modality, is a powerful tool to noninvasively map various biological and chemical markers within objects of interest. Reliable data acquisition is a major impeding factor for longitudinal hands-off measurements. Measurements are especially challenging in biomedical applications, as live objects are not static. Frequent changes occur that require constant fine recalibration of the EPR detection system, called the resonator. To enable longitudinal imaging, a technology permitting automatic digital control of resonator coupling, tuning, and EPR data acquisition was developed. Automation was achieved through the utilization of a microcontroller and digital peripheral components such as digitally controlled capacitors, a digital frequency source, and a printed circuit board resonator. Several applications of this technology have been suggested and tested, including *in vivo* EPR imaging. The first was to develop a tool for the optimization of light-based 3D printing, for which oxygen plays a major role. Towards this goal, an EPR oxygen-sensitive probe was incorporated into 3D printing resin. Oxygen depletion was measured during the 3D printing process as the polymerization front progressed. After printing, oxygen depletion was again measured during the post-curing process, proposed as a method to optimize post-curing light intensity, temperature, and duration in order to produce quality 3D printed constructs. The second application of longitudinal EPR imaging was directed toward resolving an important problem of oxygen delivery to thick ($>1$ cm) bioprinted models. Oxygen-sensitive EPR probes (water-soluble trityls or crystalline lithium octa-n-butoxynaphthalocyanine) were introduced into bioinks (liquid hydrogels containing cells, nutrients, and other biological factors) before printing. Bioinks become solid structures after printing due to crosslinking. EPR imaging was demonstrated to measure oxygen consumption by the cells embedded in the bioprints. As expected, an increase of oxygen depletion was observed by introducing a nutrient (pyruvate) to bioink. A numerical MATLAB simulation program was developed to predict rates of oxygen consumption by the cells in the bioprint. The input parameters for the mathematical model include the size and number of cells, the diffusion coefficient of the media, and rates of oxygen transfer through the cell membrane. The software is being further refined and optimized for computational speed. Future efforts will be aimed at improving the speed and scope of EPR automatic digital control, imaging the oxygen depletion process in commercial 3D printers, and applying EPR mapping of oxygen consumption rate to quantify the delivery of oxygen to cells deep inside bioprinted tissue models. Optimizing the delivery of oxygen to cells would overcome the challenge of the limit of diffusion, facilitating the development of larger and more complex bioprinted tissues and organs. These complex tissue and organ models are envisioned for use in drug testing, biomedical research, and, in the distant future, implants in humans.

*"Ludwig Boltzmann, who spent much of his life studying **EPR**, died in 1906, by his own hand. Paul Ehrenfest, carrying on the work, died similarly in 1933. Now it is our turn to study **EPR**."*

  -Adapted from: Goodstein, David L. *States of Matter*. Dover Books on Physics, 1975. Boltzmann and Ehrenfest studied statistical mechanics and thermodynamics, not EPR.

# Acknowledgments

It is through the work of many people that this dissertation was able to come to fruition, through countless hours of research, training, and support.

First and foremost, I would like to express deep appreciation to my advisor Dr. Mark Tseytlin for his support and guidance throughout the project. Starting in the lab not even knowing what EPR was or what I wanted to conduct this research on, you guided me through a myriad of research possibilities, code bugs, and frustrating results. Thank you for being my mentor.

Deep appreciation is again due to my co-advisor, Dr. Tim Eubank. Without your guidance on the biological side of things, this research would not have been complete. Thank you for always setting aside time and research space. I always felt welcomed in your lab and meetings.

Thank you to my wonderful committee: Drs. Dietz, Gunther, Pei, and Pugacheva. I could not have asked for a better team to guide this project. Thank you for the guidance through the first years of experiments and helping me to settle on a research plan, and for being forgiving when I showed up to the proposal and somehow didn't realize there was a defense after. Thank you for pushing me to include cells in the research project and providing the resources for me to do so. I am grateful for each of you.

Thank you to Vels for inspiration, knowledge, and encouragement. I had a blast trying experiments with you, and I always felt comfortable coming to you with questions. Thank you to Andrey for mentorship, guidance, and friendship. I am grateful to many faculty at WVU for their knowledge, guidance and support: Drs. Driesschaert, Webb, Gallagher, Ruppert, Khramtsov, Gencheva, Ivanov, Ramamurthy, Schaller, Stoilov, Leonardi, Raylman, Boone and Robart. The project would have been impossible without you.

Thank you to Oxana for teaching me how to use everything in the lab, for your expertise and stories. I really appreciate all the help, guidance, and mentorship. Thank you to Sajad and Faezeh for the wonderful food and company, I'll miss you both. Sajad, much of this research couldn't have been done without you. Thank you also to the undergraduate and high school students in the lab: Vignesh, Duncan, Kayna and Samuele. I am very grateful for the many friends on the fifth floor: Nicole, Kayla, Martin, Misa, Virat, Hannah, Duaa, Alyson, Hillary, Kristen, Teresa, Megan, Justin. Special thanks to Nicole, Tim, and Kayla for befriending me from the start. A quick shoutout to WVU ILL and LibGen for coming through when the journal paywalls wouldn't let me in. Also to the librarians who helped me get permission from the journals to reproduce my own work in this dissertation. Thank you also to the friends outside of the lab: Ernie, Sunshine, AJ, Kenya, Jimmy, Sisk, Swank, Nick, and many others.

Thank you to my family for supporting me through the many long years of education and research. You're the best family I've ever had. I wouldn't have gotten here without you. Thank you for listening to all my frustrations and celebrating all my triumphs.

Thank you of course to my best friend and love, Marta Majewski. You uprooted yourself from AZ to WV to support me while finishing this project. Thank you for helping me to be human and enjoy life, and for supporting me in the hard parts.

# Contents

# List of Figures

# List of Tables

# Acronyms

| | |
|---|---|
| ABS | acrylonitrile butadiene styrene |
| ADiC | Automated Digital Control |
| AFC | Automatic Frequency Control |
| AI | Artificial Intelligence |
| ASIC | Application-Specific Integrated Circuit |
| AT | Acquisition Time |
| AWG | Arbitrary Waveform Generator |
| BLN | Balun |
| BOLD | Blood Oxygen Level-Dependent (MRI) |
| CCC | Critically Coupled Conditions |
| CF | Central Frequency |
| CPU | Central Processing Unit |
| CW | Continuous Wave (EPR) |
| DAC | Digital-Analog Converter |
| DCP | Digital Control Pin |
| DDS | Direct Digital Synthesis (radiofrequency source) |
| DFS | Digital Frequency Source |
| dFT | Deuterated Finland Trityl |
| DI | Decoupling Inductance |
| DIG | Fast FPGA Digitizer |
| DLP | Digital Light Processing |
| DTC | Digitally Tunable Capacitor |
| ECM | Extracellular Matrix |
| ENDOR | pulsed Electron Nuclear Double Resonance |
| EPR | Electron Paramagnetic Resonance |
| EPRI | Electron Paramagnetic Resonance Imaging |
| ESEEM | Electron Spin Echo Envelope Modulation |
| FCC | Fixed Coupling Capacitor |
| FPGA | Field-Programmable Gate Array |
| GUI | Graphical User Interface |
| I | Imaging |
| ID | Inner Diameter |
| iPSC | induced Pluripotent Stem Cell |
| I2C | Inter-Integrated Circuit |
| L | Resonator loop |
| LIFT | Laser-induced Forward Transfer |
| LiNc-BuO | Lithium Octa-n-butoxy Naphthalocyanine |
| LiPc | Lithium Phthalocyanine |
| MC | Microcontroller |
| MISO | Master-in, Slave-out |
| MOSI | Master-out, Slave-in |
| MRI | Magnetic Resonance Imaging |
| MSC | Mesenchymal Stem Cell |
| OCR | Oxygen Consumption Rate |
| OT | Overhead Time |

| | |
|---|---|
| PC | Personal Computer |
| PCB | Printed Circuit Board |
| PCBR | Printed Circuit Board Resonator |
| PET | Positron Emission Tomography |
| PETG | Polyethylene Terephthalate Glycol |
| PLA | Polylactic Acid |
| RC | Resonating Capacitor |
| RES | Digitally Controlled Resonator |
| RF | Radiofrequency |
| RFC | Radiofrequency Connector |
| RP | Reflection Probe |
| RS | Rapid Scan (EPR) |
| S | Spectroscopy |
| SCL | Serial Clock Line |
| SD | Secure Digital (SD card) |
| SDA | Serial Data Line |
| SLA | Stereolithography |
| SNR | Signal to Noise Ratio |
| SPECT | Single-photon Computed Tomography |
| SPI | Serial Peripheral Interface |
| SS | Slave Select |
| TC | Trimmer Capacitor |
| TTL | Transistor-transistor logic |
| USB | Universal Serial Bus |
| UV | Ultraviolet |
| VTC | Voltage-tunable Capacitor |
| 2D | Two-dimensional (usually x and y spatial dimensions) |
| 3D | Three-dimensional (x,y,z spatial dimensions) |
| 5D | Five-dimensional (x,y,z spatial dimensions, time, and EPR spectrum) |

## Nomenclature

| Symbol | Units | Description |
|---|---|---|
| $i$ | | Index |
| $s$ | | Spin quantum number |
| $\mu$ | | magnetic moment |

# Chapter 1

# Introduction

When a loved one has a life-threatening disease that requires an organ transplant, what are the steps towards recovery? Unfortunately, one of those first steps is being put on a transplant wait list, sometimes for many years. As of the time of writing, there are 104,000 people on the waiting list for an organ transplant in the United States, but only 42,800 transplants were performed in 2022.[1] To overcome this crisis, scientists are working towards developing a technology that can make new organs on demand. Advances in the field of tissue engineering have led to two main strategies to generate organs for tissue transplantation.

## 1.1 Making organs: decellularized extracellular matrix vs bioprinting

The first strategy to generate organs for tissue transplantation is the decellularization and recellularization of existing organs.[2] Decellularization is the process of removing the cells from an organ, leaving only the extracellular matrix (ECM).[3,4] After removal of the cells, the decellularized organ can be preserved through freeze-drying[5] or freezing,[6] followed by sterilization,[7] allowing for long-term storage. The decellularized organ is then recellularized, typically with stem cells such as induced pluripotent stem cells (iPSCs) that differentiate after initial recellularization.[8] Before transplantation, the final step is incubation in a bioreactor, which provides a flow of oxygen and nutrients to keep the cells alive and develops organ functionality through cell differentiation.[9] Bioreactors will be discussed in more detail in a later section.

Several artificial organs have been successfully transplanted in animals following recellularization.

To give an example, a decrease in blood glucose was seen in diabetic mice after transplantation of a recellularized pancreas seeded with human mesenchymal stem cells (MSCs).[10] Upon sectioning, these pancreata were found to contain both acini ducts and islets of Langerhans. In another model, rat intestine recellularized with human iPSC-derived cells showed transfer of glucose and fatty acids from the lumen of the intestine to the venous effluent.[8] Next, repopulation of decellularized mouse heart with human IPSC-derived cells resulted in transcription of cardiac-associated genes, spontaneous contractions of the heart, and increased contraction rate in response to isoproterenol treatment.[11]

Decellularized ECM without recellularization is already used in many clinical applications.[12] Acellular dermal matrix, a soft connective tissue graft, has been used to improve outcomes in Achilles tendon[13] and rotator cuff[14] repair, as well as to improve the healing rate for burns[15] and other wounds.[16] Outside of acellular dermal matrix, decellularized heart valves show an improvement in reoperation rates over standard valve replacement.[17]

As with any technology, decellularized ECM has benefits and limitations. One of the largest benefits to using decellularized ECM is that the vascular structure of the decellularized organ remains intact.[18] The existing vasculature is used for perfusion during the decellularization process to remove the cells,[19] and is also used during and after recellularization to keep the new cells alive by supplying oxygen and nutrients.[20] Another benefit to the use of decellularized ECM from animals or grown in cell culture is the potential large-scale availability of the organs for transplant compared to traditional organs harvested from humans.[21] In theory, decellularized ECM transplants should have lower incidence of rejection compared to typical organ transplants because the original cells are removed and replaced with the patient's own cells.[22] However, in practice there is still a significant immune response, perhaps resulting in part by changes to the ECM caused by the decellularization process.[23] Another limitation to decellularized ECM in contact with blood is the requirement to completely line the vasculature with endothelial cells.[24] Thrombosis, or the formation of blood clots, remains a challenge in decellularized ECM that is transplanted and in contact with blood.[25]

An alternative strategy to generate organs for tissue transplantation is the new and still developing field of bioprinting. Bioprinting is an additive manufacturing technique, often using a combination of cells and biomaterial, called bioink, to construct a tissue model.[26] The principles of bioprinting will be discussed more thoroughly in the next section. Briefly, bioprinting consists of two essential components: the bioink and the bioprinter. A bioink is a liquid containing cells and a biomaterial, usually a hydrogel.[27]

This bioink is polymerized using various strategies to form a solid structure, called the bioprint. Multiple bioinks can be combined to make more complex bioprints containing varying hydrogels and cell types.[28] Finally, the bioprinter is the machine that creates the three-dimensional (3D) bioprint using the bioink.

Like recellularized ECM-based tissue engineering, bioprinting has been used in preclinical studies. In one study, rats were anesthetized, and a hole drilled into their skull.[29] The hole was filled with a bone construct using intraoperative bioprinting, where the bioprinting took place directly on the site of the wound. The bioprinted bone construct also contained growth factor-encoded plasmid DNA to stimulate bone regeneration. After six weeks, the rats who received intraoperative bioprinting had 90% bone coverage compared to 25% for the control rats which healed naturally. In a different study, skin was bioprinted which had a distinct fibroblast network, basal layer, and a monolayer of keratinocytes which stratified after printing.[30] The skin was transplanted in mice and demonstrated improved wound-healing, healing faster and resulting in a thick (80µm after 14 days), stratified epithelial layer. The transplanted skin developed hair follicles after transplantation, similar to skin *in vivo*. In a fourth study, bioprinted cardiac patches composed of cardiac extracellular matrix and human progenitor cells were implanted in a rat model of right ventricular failure.[31] Rats that received the bioprinted cardiac patch showed better right ventricle function and tissue remodeling. Finally, yeast, including *Saccharomyces cerevisiae*, are commonly used in the lab to produce protein to treat certain diseases, such as the production of insulin for diabetes.[32] Introducing this yeast into a bioprint and implanting into a patient would result in long-term, stable protein release to the surrounding tissue. The release of protein from a bioprinted live biotherapeutic product has been shown over a period of ten days.[33]

Bioprinting can be used to create human-scale tissues[34] that are customized to match the shape of the original organ being replaced.[35] Bioprinting uses a liquid bioink, and as such does not require the original tissue structure used in many decellularized ECM applications. This overcomes the moral limitation of decellularized ECM: hydrogels to make bioink can come from non-animal sources in the cases of bacteria-based or algae-based alginate[36] and synthetic sources such as polyethylene glycol, poly-caprolactone and others.[37] Technology to generate recombinant human collagen is being developed,[38] but has not been commercialized. Collagen used in bioprinting still originates from animal sources, but is purified protein, eliminating the problem of immunologic rejection seen with decellularized ECM.[39] Like decellularized ECM, bioprinting promises large-scale availability of organs for transplantation.[40] At the same time, bioprinting has its own limitations. As will be discussed in more detail in a later section,

bioprinting does not retain the complex vascular system of decellularized ECM. Complex perfusion systems have been developed using bioprinting, but do not recapitulate the complex layered structure of vasculature and are large in comparison to the diffusion limit of oxygen, potentially resulting in hypoxic conditions for cells.[41]

Both decellularized ECM-based tissue engineering and bioprinting technologies hold promise as alternatives to human organ transplantation. In the meantime, both technologies also have important roles in biomedical research. The Environmental Protection Agency has announced plans to stop funding and conducting research on mammals by 2035, signalling a shift towards alternatives to animal research.[42] Tissue engineering is becoming an important alternative to animal research in several fields. Both bioprinting[43] and decellularized ECM-based tissue engineering[44] are used in high-throughput drug screening. The 3D spatial arrangement of cells has been important in understanding neurodegenerative diseases.[45] Finally, bioprinting has been shown to more closely recapitulate aspects of tumor microenvironment than other 3D culture technologies,[27] including interaction between the tumor and stromal cells,[46] tumor oxygen heterogeneity and necrosis,[47] perfusable vascular networks for metastatic models,[48] and infiltration of tumor-associated macrophages.[49] While decellularized ECM and bioprinting both have bright futures in medicine and biomedical research, the focus of this dissertation will be bioprinting, which is also the topic of the next section.

## 1.2    Bioprinting: bioprinters and bioinks

In the field of bioprinting, there are two required components: a bioprinter and bioink. When selecting the correct technology for the tissue model, biomaterial in the bioink and printing strategy for the bioprinter are important. Additionally, the bioink used for printing needs to be optimized for the bioprinter. A tabulated summary of this section is provided, see table 1.1.

There are currently about a dozen main bioprinter types (see figure 1.1 for a basic schematic of four of the most popular bioprinting strategies).[50] Many of these bioprinting strategies involve extrusion or mechanical movement of bioink to the build plate. These include inkjet, single-nozzle extrusion, multi-nozzle extrusion, co-axial extrusion, and embedded extrusion (see figure 1.1 a and b). The inkjet bioprinting style was arguably the first modern bioprinter developed in 2003, with cells deposited at the same time as the scaffold.[51] Within inkjet printing there are several methods to mechanically separate

the bioink from the printing head towards the printing bed, but the general process remains the same: microdroplets about 50µm in size are released at frequencies up to 10kHz from one of many nozzles (one bioprinter can have hundreds).[52] Next, single-nozzle extrusion is the most popular bioprinting technique due to ease-of-use, low cost and compatible bioinks.[53] At its simplest, the bioink is pushed through a syringe directly onto the printing bed. Multi-nozzle extrusion is identical to single-nozzle extrusion with the addition of multiple nozzles that can contain different bioinks to construct more complex tissues.[54] Co-axial extrusion uses a single complex nozzle, where a first bioink, called the core, is surrounded by a second bioink called the sheath.[55] By setting the core to extrude a sacrificial bioink or crosslinking solution, small (several hundred micrometers) to large (1cm) vasculature can be generated.[56] The last type of extrusion is embedded extrusion, used by our lab (see chapter 4). Similar to other extrusion methods, bioink is pushed out of a nozzle into a support bath, which holds the bioink in place while polymerization occurs.[57] This printing method allows for complex, multimaterial printing without the deformation due to the weight of the bioprint seen in other mechanical bioprinting methods.

Light-based bioprinting uses light to deposit the bioink on the printing bed (see figure 1.1 c and d). In general light-based bioprinting has the benefit of high speed and resolution, but there are less bioinks available.[58] The main bioprinting types in this category are single-vat, multi-vat, sequential injection, liquid deposition, multi-wavelength, and laser-induced forward transfer (LIFT) bioprinting.[50] In the vat bioprinting methods, another two categories exist: stereolithography (SLA) bioprinting and digital light processing (DLP) bioprinting.[59,60] Single and multi-vat bioprinting use laser light in the ultraviolet (UV) or visible range to photopolymerize the bioink onto the printing bed, which is in direct contact with a vat filled with bioink.[61] The SLA method uses point-by-point laser light to polymerize bioink to the printing bed with approximately one micrometer resolution, limited by the diameter of the laser beam.[60,62] The DLP method (used by our lab, see chapter 4) makes projections, or 2D images, which polymerize the bioink layer by layer. This layer-by-layer instead of point-by-point polymerization makes DLP faster in general than the SLA method, while retaining most of the resolution.[60] Sequential injection bioprinting consists of injecting a microfluidics chamber with bioink, polymerizing the bioink via SLA, then injecting a different bioink and repeating the procedure to produce high-fidelity, complex microstructures.[63] Multi-vat and liquid deposition bioprinting are similar in method: in multi-vat bioprinting, multiple vats are used and the bioprint is moved between vats to utilize a different bioink, while in liquid deposition, the different bioinks are placed in droplets on a glass slide instead of in a vat.[64,65] In this way, less

bioink has to be used to create the print; in vat bioprinting, a lot of the bioink is not used for the print, but is used to submerge the printing bed. In multi-wavelength bioprinting, bioink containing one cell type and multiple biomaterials are mixed together.[66] The biomaterials are designed in a way that they can be polymerized at different wavelengths, so one biomaterial is printed with UV light while another biomaterial is printed with blue light. The last light-based bioprinting method discussed here, LIFT, uses a gel bioink which is liquefied by light, forming a droplet deposited on the printing bed. This droplet crosslinks in a second bath of bioink or biomaterial on the printing bed, forming a bioprint consisting of a mosaic of gel droplets interspersed in a second gel.[67] Bioprinting technologies are selected to match the particular tissue model being printed. Bioprinter methods have different cell viability, printing speed, resolution, ease-of-use, cost, and available bioinks.

As mentioned previously, bioink consists of cells and biomaterial. Virtually any cell type can be used in bioprinting, including plant cells,[68] but to survive they must be supported by a compatible biomaterial. For this purpose, the bioinks are based on cell culture media (Ham's F12 media for culturing hepatocytes,[69] endothelial growth media 2 for endothelial cells,[70] etc.). In some cases the cells and culture media are all that is required for bioprinting, for example in the case of scaffold-free bioink.[71] The next component of the bioink is the structural material, usually a hydrogel. The structural material can be natural (agarose, alginate, chitosan, collagen, fibrin, gelatin, hyaluronic acid, Matrigel™, silk, and dECM) or synthetic (gelatin methacrylate, Pluronic F-127Ⓡ and poly(ethylene glycol)).[58] Most light-based bioprinting uses synthetic hydrogel because it needs to be photopolymerizable.[72]

A final component of bioink is the polymerizing agent, an ion, molecule, or protein added to the bioink that causes polymerization or crosslinking of the structural material. In general, all light-based bioinks contain a photoinitiator, which facilitates polymerization of the structural material in the presence of light of a certain wavelength (usually UV or blue light).[73] Photoinitiators are also present in 3D printing with resin, a technology similar to light-based bioprinting.[74] Photoinitiators react with light to form radicals and consume oxygen, discussed and measured in chapter 2.[75] Besides photopolymerization, thermal, ionic, enzymatic, and pH mediated crosslinking methods can be used.[58] Ionic crosslinking typically uses calcium cations to crosslink two molecules of structural material that contain negatively-charged groups. An example of an ionic crosslinking structural material is alginate, used by our lab. Next, an example of enzymatic crosslinking is fibrinogen and thrombin. The protein fibrinogen in the bioink is brought into contact with the enzyme thrombin, usually in the support media, and is converted

by thrombin into fibrin, which forms a fibrous matrix.[76] Finally, lowering pH causes rapid precipitation of chitosan to form a gel,[77] while collagen crosslinks (slowly) when the pH is brought to between 6.5 and 8.5 with temperature between 20-37°C.[78]

## 1.3  Technology in development today for tomorrow's bioprinted transplants

To enable bioprinting of resilient, human-sized, functional organs, two central challenges are still present. The first of these is the delivery of oxygen and nutrients and the removal of waste to all cells in the bioprint. One way to deliver oxygen to bioprinted tissue is to introduce an oxygen-releasing chemical or cell to the bioink. Calcium peroxide and sodium percarbonate have been added to bioinks and show oxygen release for at least twelve days.[79] However, catalase had to be added to improve cell viability. Calcium peroxide produces hydrogen peroxide as a byproduct, which increases production of free radicals that reduce cell viability.[80] Catalase solves this problem by breaking down hydrogen peroxide into water and oxygen.[81] A second potential issue with oxygen generation from calcium peroxide is the presence of a sudden spike of oxygen directly after printing followed by a slow decay in oxygen production, as seen by Ke et al.[79] Oxygen concentration was found to increase to 30mg/L one hour after 1mg/mL calcium peroxide addition to culture media, followed by a decay to 5mg/L oxygen concentration. For reference, saturated water oxygen concentration at room temperature is about 8.5mg/L.[82] Like hydrogen peroxide, high oxygen levels are also toxic to cells through the formation of reactive oxygen species.[83] As an alternative, oxygen-producing cells such as algae and cyanobacteria have also been used to supplement oxygen in bioprinting.[84] An algae, *C. reinhardtii*, bioprinted in co-culture with HepG2, a hepatic tumor cell line, showed steady release of oxygen and increased cell viability of the HepG2 cells.[85]

Another developing method to deliver oxygen and nutrients and remove waste for the cells in the bioprints is through vascularization. Similar to natural tissue, bioprinted tissue models can include vasculature, sometimes complex and containing endothelial cells, smooth muscle cells, and fibroblasts.[86] *In vivo*, blood vessels contain a single layer of endothelial cells, the barrier between the blood and the body, then a layer of smooth muscle cells that compress and relax to help the blood flow, and finally a layer of connective tissue containing fibroblasts, which generate ECM.[87] Each of these have been incorporated into bioprinted vasculature models. Endothelial cells are added into the bioprint by printing

with them[88] or by introducing them after printing.[89] A gap in the field of bioprinting with endothelial cells, is that, with few exceptions,[90] studies use human umbilical vein endothelial cells, which are easy to grow, while endothelial cells are in fact organ-specific in function and morphology.[91] Smooth muscle cells can be printed into vasculature and show orientation and contractile marker expression.[92] Finally, fibroblasts have also been included and result in better shape stabilization of the vessels.[93] In the body, there is a spectrum of vasculature size. The aorta is 2-3cm in diameter, arteries 2-5mm, arterioles 100-500µm, and capillaries 5-20µm.[94] The size of each vessel is important, as they carry blood from the heart to the cells in the tissue, or in this case, from the media to the cells in the bioprint. Maybe most importantly are the capillaries, where the oxygen, nutrient, and waste transfer takes place.[95] However, most bioprinted vasculature models are between the artery-arteriole size range (500µm-2mm).[96] While acellular capillary bioprints of inner diameter 5µm have been shown,[97–99] no capillary-sized vessels have yet been incorporated into a large functional bioprint. The smallest cell-laden vasculature bioprinting has achieved is about 150µm in diameter, many times larger than the size of capillaries *in vivo*.[100]

Although bioprinted vasculature incorporated into tissue models is wider than *in vivo* capillaries, it has been shown to increase cell viability in these models.[101] In an early study in 2009, channels were made using sacrificial gelatin in a collagen bioprint, and showed improvement of fibroblast cell viability from 65% to 90%.[102] More recently, Pi et al showed 90% cell viability for human umbilical vein endothelial cells over a period of 14 days in a perfusable bioprinted filament 1mm in diameter from a coaxial bioprinter.[103] Using fluorescence microscopy, Kolesky et al demonstrate the dependence of cell viability on cell density and distance from vasculature in bioprints.[96] Much research is continually being devoted towards the need to sufficiently vascularize bioprints to enable thick tissue models.[104–106]

Furthermore, the scientific community has not yet decided on an optimal layout for vasculature. Resembling the city streets of the United States, some vasculature and microchannels are outlaid in a grid pattern,[34] although zig-zags, spirals, and other patterns can be made.[100,107] To determine optimal layout for vasculature, in-depth 3D oxygen mapping will be required, along with mathematical modelling, to predict which areas of the bioprint will need oxygen, delivery of nutrients, and removal of waste.

The requirement for small capillaries and oxygen-releasing chemicals or cells is due to the "limit of diffusion" of oxygen, nutrients, and waste. In other words, the oxygen limit of diffusion refers to the depth oxygen can reach before being consumed by cells in the bioprint. Depending on the cell density and type of materials used, the diffusion limit of oxygen in 3D culture is between 40µm for cell-only

structures,[108] to 350µm for acellular structures, limiting the size of diffusion-only bioprints (with cells) to less than 3mm.[109] Complex, optimized vasculature in bioprinting needs to be developed to transfer oxygen, nutrients, and waste before the printing of functional, full-size organs is possible.[110] These complex vascular systems are possible, and have been recently modeled.[111] To confirm the success and to optimize these strategies for transferring oxygen, nutrients, and waste to and from cells in bioprints, the mapping of oxygen, pH, and other parameters will be necessary. Measuring the viability of cells in bioprints is a good start, but does not give data regarding the parameters that affect the cell viability.

After bioprinting, the structure needs to mature into a functional tissue. This process is here called 4D bioprinting, the fourth dimension referring to time.[112] To enable tissue maturation over time, a device called a bioreactor is necessary to keep the tissue alive.[113] There are currently four main types of bioreactors: spinner flask, rotating wall, compression, and perfusion. The spinner flask is a flask with a magnetic stir bar at the bottom. Bioprints can be hung from the cap of the flask and suspended in cell culture media which is spun by the stir bar,[114] though this bioreactor seems to be mostly used to generate spheroids for use in the bioink.[71,115] Rotating wall bioreactors are like media-filled hamster wheels: the media and bioprint are held in a spinning cylinder which suspends the bioprints in a microgravity environment. They have been used in several 3D culture models including skin construct[116] and cardiac tissue.[117] A related bioreactor is the Taylor-Couette bioreactor, where an inner cylinder spins and the outer wall does not move. The Taylor-Couette bioreactor has been applied in bioprinted human dermal tissue, and resulted in improved maturation of fibroblasts compared with static culture.[118] Next is the compression bioreactor, where a piston is used to put cyclical pressure on the scaffold. This type of bioreactor is important to stimulate mineral density and osteoblast differentiation in bone 3D culture.[119] The last, and most used in bioprinting, is the perfusion bioreactor. A perfusion bioreactor works by perfusing culture media through microchannels or vasculature in the bioprint.[120] A bioprinted bone scaffold was perfused for two weeks and showed phosphate deposition and improved stiffness.[121] Another study created a successful *in vitro* model of trophoblast and endothelial cell signaling using perfusion bioreactor incubation of a bioprint.[122] Trophoblasts are important cells in early pregnancy that develop into part of the placenta and provide nutrients to the embryo. Compared to the standard 2D culture model, increased angiogenic markers were found in the bioprinted model along with colocalization of the trophoblasts with endothelial cells. Bioreactors can monitor oxygen, pH, carbon dioxide, total carbon, and temperature in the culture media.[123,124] What these bioreactors lack, however, is monitoring

of the microenvironment (oxygen, pH, etc.) of the bioprint or 3D culture. Optimization of bioreactor function through mapping of the bioprint microenvironment while inside the bioreactor would facilitate the development of long-term organ models for research or transplant.

A caveat to current studies of oxygen delivery to cells in bioprints is that the oxygen was measured only at a single point[79] or on average from oxygen released by the bioprint.[85] Tissue oxygenation, whether in a bioprint model or *in vivo*, can be heterogenous. Mapping oxygen concentration throughout bioprints will become important to determine the optimal amount of oxygen-generating chemical or cells. Strategies to measure parameters of the microenvironment in bioprints will be discussed in the next section.

## 1.4    Quantitative, noninvasive mapping of the microenvironment

As discussed in the previous section, there is a need for measurement of the microenvironment of bioprints to optimize models and enable human-scale, functional organs for research and transplantation. There are several instruments that can be used for mapping of the microenvironment. Biosensors, microscopy, magnetic resonance imaging (MRI), positron emission tomography (PET), ultrasound, single-photon computed tomography (SPECT), and electron paramagnetic resonance imaging (EPRI) are all methodologies used to measure or map tissue microenvironment. Their application or potential for application towards bioprint microenvironment measurement is discussed in this section, and these technologies are summarized in table 1.2.

Biosensors are portable, rapid, and highly sensitive.[125] Encompassing a wide variety of devices, biosensors are used for medical diagnostics, chemical sensing, temperature sensing, environmental monitoring, pressure sensing, biomolecule sensing and force sensing.[126] From household glucose monitors[127] to cancer diagnostic tools,[128] biosensors have had wide-ranging and important impact in medical research, and bioprinting is no exception. Biosensors have the ability to measure pH, oxygen, and carbon dioxide, among many other parameters useful to monitor cells in bioprints.[129] Oxygen biosensors have been used to show and calculate the rate of oxygen release from calcium peroxide embedded in hydrogel.[130] The effect of perfusable microchannels on oxygen concentration in the middle of a cell-laden bioprint has also been measured using an oxygen biosensor.[131] By tracking changes in oxygen over time, biosensors have also been used to measure metabolic flux in 2D and 3D culture.[132] As one example,

metabolic flux has been recorded in bioprinted model of skeletal muscle.[133] Here, the authors treated bioprinted skeletal muscle with electron transport chain inhibitors and observed the resulting effects on the oxygen consumption rate by the cells using the Seahorse, a medical research device that utilizes an oxygen biosensor. Although biosensors are able to measure many important parameters that will be useful in bioprinting, they provide data at only one point, providing an average of the system. Additionally, to measure the microenvironment of the bioprint, the sensors have to be invasively embedded or inserted into the bioprint, which itself introduces changes in the microenvironment.[131] To map spatial data from bioprints, another method will have to be used.

Moving from single-point measurements into imaging, microscopy can be used to quantitatively and noninvasively map the microenvironment. Besides viewing the physical structure of the bioprint or cell morphology, microscopy techniques such as near infrared fluorescent microscopy have the ability to quantitatively map proteins, cell structure, oxygen,[134] reactive oxygen species,[135] pH,[136] viscosity, etc. with up to 3mm depth.[137,138] One particularly useful microscopy measurement for bioprinting is live/dead imaging, where one dye indicates live cells and another dead cells.[139] This is used to measure the ratio of cells that survive the bioprinting process or to measure the length of time cells remain viable in a bioprint. In the application of oxygen mapping, one study demonstrated oxygen microscopy of multicellular spheroids which were bioprinted.[140] Another study demonstrated oxygen mapping and metabolic flux of microalgae and human mesenchymal stem cells in a bioprinted model.[141] Out of all imaging techniques discussed here, light microscopy has by far the best resolution,[142] but is limited in depth. In addition, with increasing imaging depth, resolution loss occurs because of light scattering.[143] For imaging of thick (<1cm), human-scale, or transplanted bioprints, another method will have to be used.

MRI, the imaging implementation of nuclear magnetic resonance, is used to generate anatomic images, but can also be used for oxygen mapping through oxygen-enhanced MRI and blood oxygen level-dependent (BOLD) techniques,[144] pH,[145] and reactive oxygen species,[146] among others. MRI has been used as a quality control tool for bioprinting: after bioprinting, MRI can be used to determine porosity, hole areas, and deviations in the bioprint when compared to a reference.[147] T2*-weighted MRI was used to track cells that were labeled with micron-sized superparamagnetic iron oxide particles after *in situ* bioprinting into mouse calvarial defect.[148] Manganese-based contrast agent was used for MRI tracking of bioprinted constructs after implant, to enable postimplant intervention if necessary.[149]

Dynamic contrast-enhanced MRI and diffusion-weighted MRI were used to measure vasculogenesis, the process of new blood vessel generation, in fat graft bioprints and chondrocyte-seeded bioprints after they were implanted into mice.[150,151] Dynamic contrast-enhanced MRI and diffusion-weighted MRI techniques measure tissue perfusion and diffusion coefficient, respectively. Through the use of $T_1$-weighted or $R_1$ relaxation rate measurements, tissue oxygen level dependent MRI can measure relative tissue oxygenation,[144,152–154] and in some cases, quantitative tissue oxygenation.[155] MRI has enormous potential in the noninvasive imaging of components of the bioprint microenvironment.

PET works through the use of a positron emitter such as $^{11}$C, $^{15}$O, $^{18}$F.[156] A positron is an antimatter electron - it has the same mass as an electron, but a positive charge. When the positron comes in contact with an electron they are annihilated, resulting in the formation of two high-energy photons that travel almost perfectly 180° in respect to the other, which is what the PET machine detects. Multiple hypoxia imaging radiotracers have been developed for PET.[157] PET can map metabolic rate through the use of labeled glucose, fluorodeoxyglucose.[156] The consumption rate of oxygen can be measured by PET through the use of $^{15}$O radiotracers.[158] At the time of writing, PET has not yet been used in bioprints. Using positronium, an atom containing a positron and an electron, can be used in PET to quantitatively measure oxygen.[159] However, its low oxygen concentration resolution of 10mmHg is currently limiting in the case of measuring fine changes in oxygen in bioprints. For this, another method will have to be used

SPECT is similar to PET in the use of radioisotopes, but in SPECT gamma radiation is released and measured directly.[160] SPECT is often used to measure perfusion through blood vessels. SPECT was used to measure changes in pulmonary vasculature before and after COVID-19 infection in humans, and is regarded as the gold-standard screening test in assessment of chronic thromboembolism (clogged or narrowed blood vessels in the lungs).[161] It is also used to measure changes in perfusion during the course of Moyamoya disease (causes blood flow restriction and strokes in the brain)[162] and to measure functional blood flow in the brain.[163] Moving towards bioprinting, SPECT has been used to show bone regeneration in a poly(ethylene-glycol) hydrogel implanted in a calvarial bone defect.[164] One limitation of SPECT is resolution, which varies greatly between machines, with a range between 1.4mm to 20mm.[165,166] This limitation currently restricts SPECT to mapping general changes in the tissue rather than mapping of the microenvironment.

Ultrasound is a technique where sound waves are transferred to the object being imaged, and

the reflection is measured.[167] Doppler-ultrasound can rapidly map changes in blood flow.[168] Functional bioprints are not static, but need to change over time as the cells differentiate and excrete ECM and other molecules. Grayscale, double-transmission and pulse-echo ultrasound methods have been used to track collagen deposition over time by myofibroblasts and chondrocytes embedded in fibrin and poly(ethylene glycol) gels.[169,170] After bioprinting, cells can die due to the stresses of the printing process or because of hypoxia, or proliferate (stem cells, cancer cells). As discussed previously, live/dead microscopy or assays can be performed. As an alternative, cell concentration in bioprints can be estimated using high frequency quantitative ultrasound.[171] Finally, microscale viscoelasticity in hydrogel constructs can be mapped.[172] Photoacoustic microscopy, a hybrid modality utilizing both ultrasound and microscopy principles, has been used to image microchannel structures and blood oxygen saturation of blood moving through the microchannel structures in a bioprinted model.[173,174]

Last but not least, EPRI can be used to quantitatively and non-invasively measure components of the microenvironment in bioprints. As discussed later in detail, EPRI can be used to quantitatively map oxygen with 100μm resolution (see chapters 2,3,4). EPR imaging probe can be incorporated into resin (chapter 2) and bioink (chapter 4) to enable imaging of the microenvironment. EPR probes have high sensitivity and minimal toxicity.[175] While oxygen mapping is currently at the center of EPRI *in vivo*, EPRI can map other parameters of the microenvironment, such as inorganic phosphate,[176] glutathione,[177] pH,[178] redox status,[179] viscosity,[180] reactive oxygen species,[181] nitroxyl,[182] antioxidant capacity,[183] and hydrogen peroxide.[184] In light-based 3D printing and bioprinting, EPR has been used to measure free radical formation.[185–187] Lastly, EPR can monitor molecular viscosity and folding through spin-labeling in liquid and polymerized hydrogel.[188] While each of these parameters of the microenvironment are important on their own, the power of EPR is that multiple parameters can be measured through the use of one EPR probe. *In vivo* simultaneous measurement of oxygen partial pressure, pH, and phosphate concentration has been shown.[189] Additionally, pH and redox status can be measured simultaneously, as well as pH and glutathione.[177,179] The development of EPR probes for measurement of new parameters is under constant research. However, the use of these probes towards bioprinting is lacking, outside of measuring free radical formation. The purpose of the research in this dissertation is to demonstrate the capability of EPRI to quantitatively and noninvasively measure parameters of the microenvironment in bioprints, towards optimizing bioprint design to enable complex and human-size organ models. As a first step towards this goal, oxygen concentration will be mapped in a bioprinted model (chapter 4).

## 1.5 5D Rapid Scan Electron Paramagnetic Resonance Imaging

While EPR and NMR are very similar, EPR fundamentally utilizes electronic magnetic moments while NMR utilizes nuclear magnetic moments.[190,191] The electron is a quantum mechanical object that possesses discrete angular momentum, called spin. Associated with the spin is a quantum number, $s = \frac{1}{2}$. When a sample containing unpaired electrons is placed in a magnetic field, two energy levels become separated that correspond to the parallel and antiparallel orientation of the spin relative to the magnetic field. In the absence of a magnetic field, the magnetic moment is randomly oriented. When the energy of a photon matches the energy difference in spin states (resonance conditions), the electron switches its state by absorbing the photon. This process can be measured by the EPR detector. The resulting observation is a spectrum of energy absorption (sometimes of dispersion) as a function of the excitation frequency. Two types of excitation/detection approaches have been utilized: continuous wave (CW) and pulse. In the latter method, a broad range of frequencies is excited simultaneously. In CW detection, as the name suggests, only a single frequency is generated. In this case, EPR spectrum is observed by changing the external magnetic field.

The standard CW EPR method utilizes magnetic field modulation with a small amplitude compared to the EPR spectral width.[190] By modulating the magnetic field strength, a sinusoidal wave with an amplitude proportional to the slope of the EPR signal is formed, giving a first derivative absorption spectrum. Phase-sensitive detection at the modulation frequency is used to accurately measure the EPR spectrum.[192] In this procedure, alternating current (AC) converts the oscillating signal to a direct current (DC) EPR spectrum. A more advanced multi-harmonic detection was proposed that increased signal-to-noise ratios of the spectra by performing digital phase-sensitive detection at several harmonics of the modulation frequency.[193,194].

In our laboratory, we have developed a next-generation CW EPR technology called rapid scan (RS). This technology permits up to two orders of magnitude signal-to-noise ratio (SNR) enhancement compared to the traditional CW EPR. RS sensitivity gain stems from the phenomenon called relaxation. In the standard CW method, the spin system experiences power saturation (analogous to reversible photobleaching in optics). Relaxation, especially longitudinal $T_1$, is the time required for the spin system to recover. The result is suboptimal data acquisition, slow and limited by the maximum power. In RS EPR, the magnetic field or frequency is changed quickly relative to the characteristic relaxation rate of

the electrons in the sample being measured.[195] Spin system saturation occurs at higher powers.[196] Also, in this method, the entire spectrum is collected at once instead of measuring point by point with field modulation as in CW EPR.[196] As a result of the scan being faster than the relaxation rate, distortion of the spectrum occurs. The distortion is resolved using deconvolution.[197] In CW EPR, field modulation must be small relative to the linewidth or else line broadening will occur, but this is not necessary in RS EPR.[195] Lastly, because of rapid spectra collection, image acquisition is several times faster than in CW EPRI, which is crucial for imaging the changing microenvironment over time (as well as live animals).

From the engineering standpoint, the EPR spectrometer is somewhat analogous to optical instruments. Radiofrequency (RF) waves are directed toward the sample, interact with the media, and reflect back toward a detector. The crucial difference with optical devices is the use of a resonator that amplifies RF up to several orders of magnitude, both the excitation and detection waves. As the name suggests, this critical unit of an EPR instrument requires fine-tuning to the resonance frequency (similar to tuning your radio to a favorite channel). Two strategies have been used: change the resonator frequency to match the RF source, or vice-versa. The latter, changing frequency source, is a faster and less challenging option. The resonator efficiency is quantified as a quality factor (Q-factor), defines as the ratio between resonance frequency and full-width half maximum excitation bandwidth. Q-factor depends on the resonator design and energy absorption by the sample. RS EPR permits the matching of the signal bandwidth to that of a resonator with a given Q-factor. The standard CW method is suboptimal in this regard. The signal bandwidth is much less compared to the resonator bandwidth. The resonance frequency can be changed through the use of a tuning capacitor on the resonator, a process called tuning. To take full advantage of the resonator, source energy has to be delivered to the sample with very little reflection. This is achieved by a technique called coupling. Fine coupling is achieved by the use of trimmer inductors and/or capacitors. Our laboratory pioneered the design and use of digitally controlled capacitors for EPR permitting unparalleled speed and reliability (see Chapter 2).

Different types of resonators have been designed and used that are tailored to specific applications, including surface coil resonators, implantable resonators, and volume resonators.[198] In simple terms, the surface coil is placed on the surface of the sample,[199] an implantable resonator is implanted into the sample,[200] and the sample is placed into the volume resonator.[201] Our EPR imager currently utilizes a loop-gap volume resonator for imaging of bioprinted constructs and a surface resonator for *in vivo* measurements of breast tumors (see chapter 2).[202]

Generation of low noise excitation RF is crucially important for CW EPR detection, as phase noise that has a bandwidth of about 100kHz to 1MHz is reflected from the resonator contaminating spin responses. Various types of RF sources have been used, such as the klystron[203] or Gunn diode[204] for high frequency EPR. In lower RF, less than 1Ghz, voltage controlled oscillators (VCO) are commonly used. VCO has the disadvantage of "drifting" over time. The frequency that depends on the applied voltage would slowly change. Also any noise in the applied voltage can result in unwanted changes in frequency. The novel approach in our design is to utilize digital frequency sources that are locked to a very stable reference. The frequency can be defined down to a fraction of one Hz and does not drift over time.

With the addition of gradient coils, EPR images can be collected. Gradient coils generate additional magnetic fields that are spatially encoded.[205] By collecting spectra with different gradient field magnitudes and performing reconstruction as in computed tomography, an image can be computed.[206] The spectra containing different gradient field magnitudes are called projections.

In addition to CW and CW RS EPRI, other EPR modalities including pulse EPR, multiharmonic EPR, and time-resolved EPR can be used to give important information about the microenviornment. Without getting too far into the details of pulse EPR theory, magnetic moment ($\mu$) is another property of electrons.[207] The sum of the electron magnetic moments in a sample results in the net magnetization vector. Through the use of pulsed RF, the net magnetization vector can be perturbed, and the resulting signal measured by the resonator.[208] In this way, different characteristics of the sample can be measured through the use of different types of pulses, as in electron spin echo envelope modulation spectroscopy (ESEEM)[209] and electron nuclear double resonance spectroscopy (ENDOR).[210] Pulse EPR oxymetry is rapidly developing, with applications towards preclinical models.[211] Next, multiharmonic EPR uses magnetic field modulation greater than the measured linewidth. By extracting information from multiple harmonics of the modulation frequency, SNR of the spectrum can be increased.[193] Like RS and pulse EPRI, multiharmonic EPRI has been used to provide superior imaging over CW EPRI, especially in the case of lower signal.[194] Finally, time-resolved EPR works by detecting EPR absorption and emission signals after a brief pulse of laser irradiation.[212,213] Time-resolved EPRI has been applied to measuring diffusion in aerogels.[214]

Although in theory any paramagnetic atom or molecule can be studied with EPR, chemical probes are most often used *in vivo* to measure the microenvironment. Melanin in skin[215] and tooth enamel[216]

are two exceptions where endogenous EPR signal is measured *in vivo*. Linewidth, solubility, toxicity, and sensitivity are important in determining suitable EPR probe for imaging. Resolution for EPRI is dependent on the characteristic linewidth of the probe and on gradient strength, among other factors.[217] EPR probe is either crystalline or soluble. Crystalline probe such as lithium phthalocyanine (LiPc) and lithium octa-n-butoxynaphthalocyanine (LiNc-BuO) can be injected into an animal[218] or encapsulated in resin or bioink (see chapters 3,4). Soluble probe can be injected either systemically[219] or intratumorally[220] in animal studies. Soluble probe can also be included in bioinks, but the probe should be added to culture media after printing (see chapter 4). Some EPR probes are considered less toxic than others.[221] For example, deuterated Finland Trityl (dFT) is known to bind to albumin, potentially resulting in some toxicity,[222] while similar EPR probes OX063 and Ox071 do not bind to albumin.[223] Finally, the sensitivity, or change in linewidth proportional to change in parameter being measured, can vary between EPR probes. One example of this is LiNc-BuO, where anoxic LiNc-BuO has linewidth of 210mG and fully oxygenated ($100°$ $O_2$) LiNc-BuO has linewidth of 6.7G.[224] This is compared to LiPc which has anoxic linewidth of 14mG and fully oxygenated of 4.7G.[225] Linewidths of both LiNc-BuO and LiPc above were measured using peak-to-peak first derivative measurement. The result is that LiNc-BuO is more sensitive to oxygen than LiPc, which is good for spectroscopy, but for imaging, good resolution is strongly dependent on small linewidth. A similar case is true for Ox063 and Ox071, with Ox071 having an anoxic linewidth of 80mG and Ox063 an having anoxic linewidth of 160mG.[223] Overall, crystalline probes have greater sensitivity to changes in oxygen compared to soluble probes. In our imaging system (see chapter 4), both crystalline probes and soluble probes are used. For oxymetry, crystalline probes are preferred for repeated measurements at low oxygen concentration and in light-based printing resins (chapter 3), while soluble probes are preferred when full range from anoxia to air concentration oxygen will be imaged (see chapter 4). The wide linewidth of crystalline probes (LiPc and LiNc-BuO) when fully oxygenated make imaging challenging. Soluble probes are the best for observing tissue oxygenation throughout the tumor, while crystalline probes are used for repeated measurements at the point of probe injection. However, crystalline probes mixed with cancer cells before injection can also be used for to give tissue oxygenation throughout the tumor, although the crystalline probe spreads in a heterogeneous way as the tumor grows.[226] Crystalline probe can be dispersed throughout a resin (chapter 3) or bioink (chapter 4) for repeated measurements throughout a printed object. In our lab, it is noted that different batches of EPR probes have slightly different linewidths; each batch of EPR probe should be calibrated

before use to determine fully oxygenated and anoxic linewidths.

Using RS EPRI for animal and bioprinting applications comes with challenges that must be addressed. RS EPRI is a sensitive method: any changes in coupling away from critical coupling, or tuning away from the resonance frequency result in the loss of signal from the sample. This issue is especially egregious during imaging, where slight movements from the animal or changes in the bioprint caused by perfusion or the system heating and drifting over time, can result in loss of data in the image, potentially ruining the image or the entire experiment, if uncorrected over several images. In bioprinting, it will be necessary not only to map the microenvironment in the short term, but to measure how the microenvironment changes over periods days[227] or even weeks.[228] Lastly, most EPR imagers use analog components in the resonator (capacitors are fixed, mechanically tunable or voltage controlled). Mechanical capacitors require human adjustment, while in RS EPR, voltage controlled capacitors are sensitive to noise and induction of small voltages from the rapid scan coils. For all of these reasons, it is be necessary to digitize and enable automation of the RS EPRI system (see chapter 2).

## 1.6   Automation: digital and microcontroller technologies

As discussed in the previous section, analog components have several limitations: inherent background noise, RS coil induction of voltage noise, and the challenge of automation. Digital components have the advantage in each of these limitations. A digital signal takes the form of a zero or a one, called a bit. It could also be thought of as a switch, either on or off, for a short period of time. Digital bits become an advantage when noise in the transmission line from controller to digital component is present, whether induced by RS or coming from the external environment. Transistor to transistor logic (TTL) is a form of digital logic design used by many digital components.[229] Commonly used TTL levels are 3.3V and 5V, referring to the voltage supply. For a 5V TTL gate, a typical "1" is registered when the component receives an input between 2-5V, and a typical "0" is registered when the component receives an input between 0-0.8V. These levels will vary slightly between manufacturers, but the wide range of voltage for each state, on or off, means that any noise, unless exceedingly large, will not be registered by the digital component. As a result, noise in the transmission line between the controller and the component is eliminated and is not transferred to the EPR system. The last advantage is automation. Automation can be done with voltage-controlled components through the use of a digital-analog converter (DAC).

As in the name, a DAC converts digital signal into analog voltage or current. However, automation through digital components only can be faster and more reliable.

There are several automation-enabling technologies that can replace analog versions of RS EPRI equipment. These include digital frequency sources, digitally tunable capacitors (DTCs), digitizers, and the automated controller. For implementation of each of these technologies, see chapter 2. First, digital frequency sources can often be tuned to 1Hz resolution and have low phase and amplitude noise after filtering. For example, a digital frequency source for 6-9GHz has been made to jump frequencies in 7ns, with phase noise about -117dBc/Hz.[230] Another digital frequency source for 1.4-3GHz with time latency of 245ns, power stabilization of 2.7dBm and phase noise -75dBc, and was used to measure single electron spin in a nitrogen-vacancy center in diamond.[231] Next, a DTC is at heart an array of fixed capacitors that are digitally selected with switches to result in a desired net capacitance. An example of a DTC is the Peregrine Semiconductor PE64904, which has 32 possible capacitances between 0.6-4.6pF, operation range 100MHz-3GHz, power range up to 38dBm, and switching time of 12μs.[232] Next is the digitizer. A digitizer is more than an analog-digital converter, but also can include many functions of an oscilloscope like real-time averaging, and multiple inputs and triggers.[196,233] Digitizers are already used in RS EPRI.[202]

The automation-enabling technology at the heart of automation itself is the controller. A controller can be a microprocessor, a microcontroller, an application specific integrated circuit (ASIC), or a field-programmable gate array (FPGA).[234] The microprocessor is, in simple terms, the raw central processing unit (CPU) attached to a printed circuit board (PCB). This microprocessor can be programmed to compute complex tasks like running artificial intelligence (AI) algorithms.[235] Adding memory, clock generator, a display, or other components turns the microprocessor into a computer. Computers have been used to run EPR machines for decades and are relatively easy to program, but can be slow or unpredictable because the CPU needs to process many tasks in serial (one at a time), like running the desktop, background processes, EPR software, automatic updates and antivirus, etc. Second, a microcontroller is an inexpensive and compact computer designed to perform a specific task or tasks using serial processing. Microcontrollers often have electrical inputs and outputs, which makes it easy to connect other digital components and make prototype automation systems. The classic example of a microcontroller is the Arduino.[236] Third, the FPGA uses an array of interconnected logic gates, where the connections between the gates can be changed through programming.[237] A logic gate is a

basic AND, NAND, OR, NOR. For example, an AND gate only returns a "one" digital state if both inputs are also ones; otherwise, it returns a zero. The FPGA is programmed with a special hardware description language, and can conduct automated tasks using parallel processing. Because of both of these traits - the simpler language and parallel processing - FPGAs are faster than microprocessors and microcontrollers. Finally, an ASIC, as the name suggest, is a circuit designed for the specific task, and cannot be reprogrammed like an FPGA, microcontroller, or microprocessor. Similar to an FPGA, an ASIC can compute tasks with parallel processing, and uses a logic gate array that is optimized and hard-wired specifically for the application.[238] For this reason, ASICs are the fastest, most reliable controllers (if designed properly). Once an ASIC has been designed, it can also be mass-produced at lower cost. In general, automated designs progress from proof-of-principle on a computer, to early prototype with a microcontroller, to fast and reliable prototype on FPGA, and finally commercial product with ASIC, although this process is not always the case.

To communicate between different devices, such as computer to microcontroller or microcontroller to components, different communication protocols can be used, including Universal Serial Bus (USB), serial peripheral interface (SPI), inter-integrated circuit (I2C), WiFi, Ethernet, Bluetooth, and many more.[239] Depending on the protocol, components can be set up in a slave/master system, or a multimaster system. In the slave/master system, a slave component waits to receive data, and only sends communication back if requested. In a multimaster system, any of the components can send information. In general, the information sent from one component to another is contained in a finite set of bits (a digital zero or one) called a packet. This packet is structured, typically specifying which component is communicating, which component the packet is being sent to, the number of bits in the packet, and some special bits at the beginning and end of the packet to make sure the packet is received without error. USB connectors contain varying amounts of wires, but at the most basic contain a ground (0V), +V (5V usually), and two data wires (data+ and data- for a differential signal transmission). USB ground and voltage lines can be used to power components (phone charger as an example). SPI uses four lines as well - SCL (serial clock line), MISO (master-in, slave-out), MOSI (master-out, slave-in), and SS (slave select). SCL sets the speed of incoming bits, SS selects the component to be communicated to (each component gets its own dedicated SS line), and MOSI and MISO send data from master to slave or slave to master, respectively. SPI does not require as much structure in the packet, because there is only one master, and the component is selected through an SS line. Next, I2C only uses two lines, SCL and

SDA (serial data line). The component to be communicated with is selected by including an address in the beginning of the communication. Generally in I2C the component sends an acknowledgement bit back to the master to confirm it received the message. Ethernet requires more structured packets, with longer addresses in the beginning of the packet, but allow very fast communication speeds. WiFi and Bluetooth communication require a special chip to send and receive the signal. The advantages and limitations of each is summarized in table 1.3.

In addition to types of communication, this communication can be passed through copper wires (most common), through fiber-optic cables, or through the air (WiFi, Bluetooth). One advantage to fiber-optic cables and air transmission is decoupling of the electric circuits between the two components (potentially reducing noise transmission between devices).

## 1.7   3D Printing

As a gateway to bioprinting, oxygen mapping of 3D printed resin was performed (see chapter 3). Oxygen is important in light-based 3D printing (as well as light-based bioprinting) as an inhibitor to photopolymerization.[240] Here, a brief overview of the types of plastic 3D printers and inks is discussed. 3D printing of other materials (cement,[241] metal[242]) is also possible, but not discussed here.

Similar to bioprinting, mechanical and light-based 3D printing methods are popular.[243] Photopolymerization is used in light-based 3D printing, much the same as in bioprinting.[244] However, in mechanical 3D printing heat is generally used to bind material together as opposed to the ionic, enzymatic, or pH crosslinking seen in bioprinting. Most popular are the filament 3D printers.[243] These print by melting a solid filament, which solidifies as it cools into the final product. Filament can have added wood, metal, or ceramic components.[245,246] Common polymers include polylactic acid (PLA), acrylonitrile butadiene styrene (ABS), polyethylene terephthalate glycol (PETG), and nylon. The next most popular 3D printing methods are light-based 3D printing, SLA and DLP. These methods either use laser-based or projection-based light to polymerize plastic resins to form the object in SLA and DLP printing, respectively.[244] These printing methods are much the same in bioprinting, see section 1.2. As with other methods, a variety of resins are available. The next method is material jetting, where much like inkjet bioprinting, liquid is sprayed from multiple heads to the printing bed in a thin layer.[247] However, the material printed is photopolymerized once deposited. The fourth method, powder bed fusion, works by

selectively melting a powder with a laser to form the object.[248] Mostly for use with metal, polymers can also be used. Finally, sheet lamination works by selectively melting the area of a thin sheet to bind it to a layer below, forming the object.[249] One of the earliest 3D printing techniques, this method suffers from low resolution and generation of large amounts of waste. These technologies are summarized in table 1.4.

## 1.8  Summary

As of the time of writing, 104,000 people are on the waiting list for an organ transplant, yet only 42,800 transplants were performed in the prior year.[1] Additionally, the events leading to donor availability (brain death), removal, storage, transport, and transplantation result in significant hypoxia, potential damage, and changes in the organ to be transplanted.[250] Advances in the field of tissue engineering promise to solve this problem in the future by providing organs on demand. Two technologies in particular hold promise for on-demand organs include recellularized dECM and bioprinting. The field of bioprinting is relatively new and rapidly developing, with advances in bioprinter and bioink strategies and complexity. To enable the bioprinting of transplantable, functional organs, improvements in oxygen and nutrient delivery, waste removal, and tissue maturation still have to be realized. To achieve this, oxygen-releasing materials, full-scale vascularization from arteries down to capillaries, and bioreactors need to be developed and optimized. Testing of the microenvironment is required to confirm the success of these strategies, and, later, the successful integration of the transplanted bioprint in the body. Biosensors, microscopy, MRI, PET, ultrasound, SPECT, and EPRI may all play roles in measuring parameters of the microenvironment. In particular, EPRI is highlighted in this dissertation because of its non-invasive, quantitative, and multifunctional capabilities. RS EPRI has been chosen for this task for its faster image collection and higher signal-to-noise than CW EPRI. However, to enable long-term quantitative mapping of the microenvironment of bioprints, automation and digital-enabling of existing tuning, coupling, and data acquisition will have to be completed. To achieve this goal, several digital and microcontroller technologies have recently become available, such as stable and fast-scanning digital frequency sources, DTCs, digitizers, and automated controllers.

Starting from the development of an automated digital control (ADiC) for optimizing RS EPRI tuning, coupling, and data acquisition, the potential of EPRI to measure the 5D microenvironment (x,

y, z, time, and EPR spectrum) of bioprints will be demonstrated here for the first time. First, a fast microcontroller (Teensy) will be used alongside a digital frequency source, digitally-enabled resonator and digitizer to tune, couple, and collect EPRI data automatically, with minimal input from the user. Next, an application of ADiC towards 3D printing will be demonstrated as a first step towards bioprinting. Finally, RS EPRI mapping of the bioprint microenvironment will be shown.

# References

[1] (2023). United Network for Organ Sharing Frequently asked questions: Waiting and preparing for a transplant. URL unos.org/transplant/frequently-asked-questions.

[2] Hillebrandt KH, Everwien H, Haep N, Keshi E, Pratschke J, and Sauer IM (2019) Strategies based on organ decellularization and recellularization. Transplant International tri.13462. ISSN 0934-0874, 1432-2277. doi:10.1111/tri.13462. URL https://onlinelibrary.wiley.com/doi/10.1111/tri.13462.

[3] Yang J, Dang H, and Xu Y (2022) Recent advancement of decellularization extracellular matrix for tissue engineering and biomedical application. Artificial Organs 46(4):549–567. ISSN 0160-564X, 1525-1594. doi:10.1111/aor.14126. URL https://onlinelibrary.wiley.com/doi/10.1111/aor.14126.

[4] Li J and Pei M (2017) A Protocol to Prepare Decellularized Stem Cell Matrix for Rejuvenation of Cell Expansion and Cartilage Regeneration. In Turksen K, editor, *Decellularized Scaffolds and Organogenesis*, volume 1577, 147–154. Springer New York, New York, NY. ISBN 978-1-4939-7655-3 978-1-4939-7656-0. doi:10.1007/7651_2017_27. URL http://link.springer.com/10.1007/7651_2017_27. Series Title: Methods in Molecular Biology.

[5] Brouki Milan P, Pazouki A, Joghataei MT, Mozafari M, Amini N, Kargozar S, Amoupour M, Latifi N, and Samadikuchaksaraei A (2020) Decellularization and preservation of human skin: A platform for tissue engineering and reconstructive surgery. Methods 171:62–67. ISSN 10462023. doi:10.1016/j.ymeth.2019.07.005. URL https://linkinghub.elsevier.com/retrieve/pii/S1046202318304420.

[6] Feng H, Xu Y, Luo S, Dang H, Liu K, and Sun WQ (2020) Evaluation and preservation of vascular architectures in decellularized whole rat kidneys. Cryobiology 95:72–79. ISSN 00112240. doi:10.1016/j.cryobiol.2020.06.003. URL https://linkinghub.elsevier.com/retrieve/pii/S0011224020300389.

[7] Matuska AM and McFetridge PS (2015) The effect of terminal sterilization on structural and biophysical properties of a decellularized collagen-based scaffold; implications for stem cell ad-

hesion. Journal of Biomedical Materials Research Part B: Applied Biomaterials 103(2):397–406. ISSN 15524973. doi:10.1002/jbm.b.33213. URL https://onlinelibrary.wiley.com/doi/10.1002/jbm.b.33213.

[8] Kitano K, Schwartz DM, Zhou H, Gilpin SE, Wojtkiewicz GR, Ren X, Sommer CA, Capilla AV, Mathisen DJ, Goldstein AM, Mostoslavsky G, and Ott HC (2017) Bioengineering of functional human induced pluripotent stem cell-derived intestinal grafts. Nature Communications 8(1):765. ISSN 2041-1723. doi:10.1038/s41467-017-00779-y. URL https://www.nature.com/articles/s41467-017-00779-y.

[9] Scarrit ME (2015) A review of cellularization strategies for tissue engineering of whole organs. Frontiers in Bioengineering and Biotechnology 3. ISSN 22964185. doi:10.3389/fbioe.2015.00043. URL http://journal.frontiersin.org/Article/10.3389/fbioe.2015.00043/abstract.

[10] Uday Chandrika K, Tripathi R, Kameshwari Y, Rangaraj N, Mahesh Kumar J, and Singh S (2021) Refunctionalization of Decellularized Organ Scaffold of Pancreas by Recellularization: Whole Organ Regeneration into Functional Pancreas. Tissue Engineering and Regenerative Medicine 18(1):99–112. ISSN 1738-2696, 2212-5469. doi:10.1007/s13770-020-00296-y. URL https://link.springer.com/10.1007/s13770-020-00296-y.

[11] Lu TY, Lin B, Kim J, Sullivan M, Tobita K, Salama G, and Yang L (2013) Repopulation of decellularized mouse heart with human induced pluripotent stem cell-derived cardiovascular progenitor cells. Nature Communications 4(1):2307. ISSN 2041-1723. doi:10.1038/ncomms3307. URL https://www.nature.com/articles/ncomms3307.

[12] Yao Q, Zheng YW, Lan QH, Kou L, Xu HL, and Zhao YZ (2019) Recent development and biomedical applications of decellularized extracellular matrix biomaterials. Materials Science and Engineering: C 104:109942. ISSN 09284931. doi:10.1016/j.msec.2019.109942. URL https://linkinghub.elsevier.com/retrieve/pii/S0928493119310513.

[13] Cole W, Samsell B, and Moore MA (2018) Achilles Tendon Augmented Repair Using Human Acellular Dermal Matrix: A Case Series. The Journal of Foot and Ankle Surgery 57(6):1225–1229. ISSN 10672516. doi:10.1053/j.jfas.2018.03.006. URL https://linkinghub.elsevier.com/retrieve/pii/S1067251618300784.

[14] Awad MA, Sparavalo S, Ma J, King JP, and Wong I (2022) Interposition Graft Bridging Reconstruction of Irreparable Rotator Cuff Tears Using Acellular Dermal Matrix: Medium-Term Results. Arthroscopy: The Journal of Arthroscopic & Related Surgery 38(3):692–698. ISSN 07498063. doi:10.1016/j.arthro.2021.08.001. URL https://linkinghub.elsevier.com/retrieve/pii/S0749806321007295.

[15] Guo ZQ, Qiu L, Gao Y, Li JH, Zhang XH, Yang XL, Peszel A, and Chen XL (2016) Use of porcine acellular dermal matrix following early dermabrasion reduces length of stay in extensive deep dermal burns. Burns 42(3):598–604. ISSN 03054179. doi:10.1016/j.burns.2015.10.018. URL https://linkinghub.elsevier.com/retrieve/pii/S0305417915003289.

[16] Petrie K, Cox CT, Becker BC, and MacKay BJ (2022) Clinical applications of acellular dermal matrices: A review. Scars, Burns & Healing 8:205951312110383. ISSN 2059-5131, 2059-5131. doi:10.1177/20595131211038313. URL http://journals.sagepub.com/doi/10.1177/20595131211038313.

[17] Waqanivavalagi SWFR, Bhat S, Ground MB, Milsom PF, and Cornish J (2020) Clinical performance of decellularized heart valves versus standard tissue conduits: a systematic review and meta-analysis. Journal of Cardiothoracic Surgery 15(1):260. ISSN 1749-8090. doi:10.1186/s13019-020-01292-y. URL https://cardiothoracicsurgery.biomedcentral.com/articles/10.1186/s13019-020-01292-y.

[18] Kajbafzadeh AM, Khorramirouz R, Nabavizadeh B, Ladi Seyedian SS, Akbarzadeh A, Heidari R, Masoumi A, Azizi B, and Seyed Hossein Beigi R (2019) Whole organ sheep kidney tissue engineering and in vivo transplantation: Effects of perfusion-based decellularization on vascular integrity. Materials Science and Engineering: C 98:392–400. ISSN 09284931. doi:10.1016/j.msec.2019.01.018. URL https://linkinghub.elsevier.com/retrieve/pii/S0928493118312864.

[19] Guyette JP, Gilpin SE, Charest JM, Tapias LF, Ren X, and Ott HC (2014) Perfusion decellularization of whole organs. Nature Protocols 9(6):1451–1468. ISSN 1754-2189, 1750-2799. doi:10.1038/nprot.2014.097. URL https://www.nature.com/articles/nprot.2014.097.

[20] Wüthrich T, Lese I, Haberthür D, Zubler C, Hlushchuk R, Hewer E, Maistriaux L, Gianello P, Lengelé B, Rieben R, Vögelin E, Olariu R, Duisit J, and Taddeo A (2020) Development of

vascularized nerve scaffold using perfusion-decellularization and recellularization. Materials Science and Engineering: C 117:111311. ISSN 09284931. doi:10.1016/j.msec.2020.111311. URL https://linkinghub.elsevier.com/retrieve/pii/S092849312033229X.

[21] Godehardt AW and Tönjes RR (2020) Xenotransplantation of decellularized pig heart valves—Regulatory aspects in Europe. Xenotransplantation 27(3). ISSN 0908-665X, 1399-3089. doi:10.1111/xen.12609. URL https://onlinelibrary.wiley.com/doi/10.1111/xen.12609.

[22] Kasravi M, Ahmadi A, Babajani A, Mazloomnejad R, Hatamnejad MR, Shariatzadeh S, Bahrami S, and Niknejad H (2023) Immunogenicity of decellularized extracellular matrix scaffolds: a bottleneck in tissue engineering and regenerative medicine. Biomaterials Research 27(1):10. ISSN 2055-7124. doi:10.1186/s40824-023-00348-z. URL https://biomaterialsres.biomedcentral.com/articles/10.1186/s40824-023-00348-z.

[23] Chakraborty J, Roy S, and Ghosh S (2020) Regulation of decellularized matrix mediated immune response. Biomaterials Science 8(5):1194–1215. ISSN 2047-4830, 2047-4849. doi:10.1039/C9BM01780A. URL http://xlink.rsc.org/?DOI=C9BM01780A.

[24] Hussein KH, Park KM, Yu L, Song SH, Woo HM, and Kwak HH (2020) Vascular reconstruction: A major challenge in developing a functional whole solid organ graft from decellularized organs. Acta Biomaterialia 103:68–80. ISSN 17427061. doi:10.1016/j.actbio.2019.12.029. URL https://linkinghub.elsevier.com/retrieve/pii/S174270611930858X.

[25] Momtahan N, Sukavaneshvar S, Roeder BL, and Cook AD (2015) Strategies and Processes to Decellularize and Recellularize Hearts to Generate Functional Organs and Reduce the Risk of Thrombosis. Tissue Engineering Part B: Reviews 21(1):115–132. ISSN 1937-3368, 1937-3376. doi:10.1089/ten.teb.2014.0192. URL https://www.liebertpub.com/doi/10.1089/ten.teb.2014.0192.

[26] Derakhshanfar S, Mbeleck R, Xu K, Zhang X, Zhong W, and Xing M (2018) 3D bioprinting for biomedical devices and tissue engineering: A review of recent trends and advances. Bioactive Materials 3(2):144–156. ISSN 2452199X. doi:10.1016/j.bioactmat.2017.11.008. URL https://linkinghub.elsevier.com/retrieve/pii/S2452199X17301068.

[27] Datta P, Dey M, Ataie Z, Unutmaz D, and Ozbolat IT (2020) 3D bioprinting for reconstituting the cancer microenvironment. npj Precision Oncology 4(1):18. ISSN 2397-768X. doi:10.1038/s41698-020-0121-2. URL https://www.nature.com/articles/s41698-020-0121-2.

[28] Chimene D, Kaunas R, and Gaharwar AK (2020) Hydrogel Bioink Reinforcement for Additive Manufacturing: A Focused Review of Emerging Strategies. Advanced Materials 32(1):1902026. ISSN 0935-9648, 1521-4095. doi:10.1002/adma.201902026. URL https://onlinelibrary.wiley.com/doi/10.1002/adma.201902026.

[29] Moncal KK, Tigli Aydın RS, Godzik KP, Acri TM, Heo DN, Rizk E, Wee H, Lewis GS, Salem AK, and Ozbolat IT (2022) Controlled Co-delivery of pPDGF-B and pBMP-2 from intraoperatively bioprinted bone constructs improves the repair of calvarial defects in rats. Biomaterials 281:121333. ISSN 01429612. doi:10.1016/j.biomaterials.2021.121333. URL https://linkinghub.elsevier.com/retrieve/pii/S014296122100689X.

[30] Li M, Sun L, Liu Z, Shen Z, Cao Y, Han L, Sang S, and Wang J (2023) 3D bioprinting of heterogeneous tissue-engineered skin containing human dermal fibroblasts and keratinocytes. Biomaterials Science 11(7):2461–2477. ISSN 2047-4830, 2047-4849. doi:10.1039/D2BM02092K. URL http://xlink.rsc.org/?DOI=D2BM02092K.

[31] Bejleri D, Robeson MJ, Brown ME, Hunter J, Maxwell JT, Streeter BW, Brazhkina O, Park HJ, Christman KL, and Davis ME (2022) *In vivo* evaluation of bioprinted cardiac patches composed of cardiac-specific extracellular matrix and progenitor cells in a model of pediatric heart failure. Biomaterials Science 10(2):444–456. ISSN 2047-4830, 2047-4849. doi:10.1039/D1BM01539G. URL http://xlink.rsc.org/?DOI=D1BM01539G.

[32] Baeshen NA, Baeshen MN, Sheikh A, Bora RS, Ahmed MMM, Ramadan HAI, Saini KS, and Redwan EM (2014) Cell factories for insulin production. Microbial Cell Factories 13(1):141. ISSN 1475-2859. doi:10.1186/s12934-014-0141-0. URL https://microbialcellfactories.biomedcentral.com/articles/10.1186/s12934-014-0141-0.

[33] Etter EL, Heavey MK, Errington M, and Nguyen J (2023) Microbe-loaded bioink designed to support therapeutic yeast growth. Biomaterials Science 10.1039.D3BM00514C. ISSN 2047-4830, 2047-4849. doi:10.1039/D3BM00514C. URL http://xlink.rsc.org/?DOI=D3BM00514C.

[34] Kang HW, Lee SJ, Ko IK, Kengla C, Yoo JJ, and Atala A (2016) A 3D bioprinting system to produce human-scale tissue constructs with structural integrity. Nature Biotechnology 34(3):312–319. ISSN 1087-0156, 1546-1696. doi:10.1038/nbt.3413. URL https://www.nature.com/articles/nbt.3413.

[35] Gonzalez-Fernandez T, Tenorio AJ, Campbell KT, Silva EA, and Leach JK (2021) Alginate-Based Bioinks for 3D Bioprinting and Fabrication of Anatomically Accurate Bone Grafts. Tissue Engineering Part A 27(17-18):1168–1181. ISSN 1937-3341, 1937-335X. doi:10.1089/ten.tea.2020.0305. URL https://www.liebertpub.com/doi/10.1089/ten.tea.2020.0305.

[36] Hernández-González AC, Téllez-Jurado L, and Rodríguez-Lorenzo LM (2020) Alginate hydrogels for bone tissue engineering, from injectables to bioprinting: A review. Carbohydrate Polymers 229:115514. ISSN 01448617. doi:10.1016/j.carbpol.2019.115514. URL https://linkinghub.elsevier.com/retrieve/pii/S0144861719311828.

[37] Khoeini R, Nosrati H, Akbarzadeh A, Eftekhari A, Kavetskyy T, Khalilov R, Ahmadian E, Nasibova A, Datta P, Roshangar L, Deluca DC, Davaran S, Cucchiarini M, and Ozbolat IT (2021) Natural and Synthetic Bioinks for 3D Bioprinting. Advanced NanoBiomed Research 1(8):2000097. ISSN 2699-9307, 2699-9307. doi:10.1002/anbr.202000097. URL https://onlinelibrary.wiley.com/doi/10.1002/anbr.202000097.

[38] Wang T, Lew J, Premkumar J, Poh CL, and Win Naing M (2017) Production of recombinant collagen: state of the art and challenges. Engineering Biology 1(1):18–23. ISSN 2398-6182, 2398-6182. doi:10.1049/enb.2017.0003. URL https://onlinelibrary.wiley.com/doi/10.1049/enb.2017.0003.

[39] Olegovich Osidak E, Igorevich Kozhukhov V, Sergeevna Osidak M, and Petrovich Domogatskiy S (2020) Collagen as Bioink for Bioprinting: A Comprehensive Review. International Journal of Bioprinting 6(3). ISSN 2424-8002, 2424-7723. doi:10.18063/ijb.v6i3.270. URL http://ijb.whioce.com/index.php/int-j-bioprinting/article/view/270.

[40] Murphy SV and Atala A (2014) 3D bioprinting of tissues and organs. Nature Biotechnology 32(8):773–785. ISSN 1087-0156, 1546-1696. doi:10.1038/nbt.2958. URL https://www.nature.com/articles/nbt.2958.

[41] Zhang Y, Kumar P, Lv S, Xiong D, Zhao H, Cai Z, and Zhao X (2021) Recent advances in 3D bioprinting of vascularized tissues. Materials & Design 199:109398. ISSN 02641275. doi: 10.1016/j.matdes.2020.109398. URL https://linkinghub.elsevier.com/retrieve/pii/S0264127520309345.

[42] Grimm D (2019) EPA plan to end animal testing splits scientists. Science 365(6459):1231–1231. ISSN 0036-8075, 1095-9203. doi:10.1126/science.365.6459.1231. URL https://www.science.org/doi/10.1126/science.365.6459.1231.

[43] Engel M, Belfiore L, Aghaei B, and Sutija M (2022) Enabling high throughput drug discovery in 3D cell cultures through a novel bioprinting workflow. SLAS Technology 27(1):32–38. ISSN 24726303. doi:10.1016/j.slast.2021.10.002. URL https://linkinghub.elsevier.com/retrieve/pii/S2472630321000029.

[44] Ferreira LP, Gaspar VM, Mendes L, Duarte IF, and Mano JF (2021) Organotypic 3D decellularized matrix tumor spheroids for high-throughput drug screening. Biomaterials 275:120983. ISSN 01429612. doi:10.1016/j.biomaterials.2021.120983. URL https://linkinghub.elsevier.com/retrieve/pii/S0142961221003392.

[45] Centeno EGZ, Cimarosti H, and Bithell A (2018) 2D versus 3D human induced pluripotent stem cell-derived cultures for neurodegenerative disease modelling. Molecular Neurodegeneration 13(1):27. ISSN 1750-1326. doi:10.1186/s13024-018-0258-4. URL https://molecularneurodegeneration.biomedcentral.com/articles/10.1186/s13024-018-0258-4.

[46] Langer EM, Allen-Petersen BL, King SM, Kendsersky ND, Turnidge MA, Kuziel GM, Riggers R, Samatham R, Amery TS, Jacques SL, Sheppard BC, Korkola JE, Muschler JL, Thibault G, Chang YH, Gray JW, Presnell SC, Nguyen DG, and Sears RC (2019) Modeling Tumor Phenotypes In Vitro with Three-Dimensional Bioprinting. Cell Reports 26(3):608–623.e6. ISSN 22111247. doi:10.1016/j.celrep.2018.12.090. URL https://linkinghub.elsevier.com/retrieve/pii/S2211124718320503.

[47] Hribar KC, Finlay D, Ma X, Qu X, Ondeck MG, Chung PH, Zanella F, Engler AJ, Sheikh F, Vuori K, and Chen SC (2015) Nonlinear 3D projection printing of concave hydrogel microstructures for

long-term multicellular spheroid and embryoid body culture. Lab on a Chip 15(11):2412–2418. ISSN 1473-0197, 1473-0189. doi:10.1039/C5LC00159E. URL http://xlink.rsc.org/?DOI=C5LC00159E.

[48] Meng F, Meyer CM, Joung D, Vallera DA, McAlpine MC, and Panoskaltsis-Mortari A (2019) 3D Bioprinted In Vitro Metastatic Models via Reconstruction of Tumor Microenvironments. Advanced Materials 31(10):1806899. ISSN 09359648. doi:10.1002/adma.201806899. URL https://onlinelibrary.wiley.com/doi/10.1002/adma.201806899.

[49] Heinrich MA, Bansal R, Lammers T, Zhang YS, Michel Schiffelers R, and Prakash J (2019) 3D-Bioprinted Mini-Brain: A Glioblastoma Model to Study Cellular Interactions and Therapeutics. Advanced Materials 31(14):1806590. ISSN 0935-9648, 1521-4095. doi:10.1002/adma.201806590. URL https://onlinelibrary.wiley.com/doi/10.1002/adma.201806590.

[50] Ravanbakhsh H, Karamzadeh V, Bao G, Mongeau L, Juncker D, and Zhang YS (2021) Emerging Technologies in Multi-Material Bioprinting. Advanced Materials 33(49):2104730. ISSN 0935-9648, 1521-4095. doi:10.1002/adma.202104730. URL https://onlinelibrary.wiley.com/doi/10.1002/adma.202104730.

[51] Mironov V, Boland T, Trusk T, Forgacs G, and Markwald RR (2003) Organ printing: computer-aided jet-based 3D tissue engineering. Trends in Biotechnology 21(4):157–161. ISSN 01677799. doi:10.1016/S0167-7799(03)00033-7. URL https://linkinghub.elsevier.com/retrieve/pii/S0167779903000337.

[52] Li X, Liu B, Pei B, Chen J, Zhou D, Peng J, Zhang X, Jia W, and Xu T (2020) Inkjet Bioprinting of Biomaterials. Chemical Reviews 120(19):10793–10833. ISSN 0009-2665, 1520-6890. doi:10.1021/acs.chemrev.0c00008. URL https://pubs.acs.org/doi/10.1021/acs.chemrev.0c00008.

[53] Ramesh S, Harrysson OL, Rao PK, Tamayol A, Cormier DR, Zhang Y, and Rivero IV (2021) Extrusion bioprinting: Recent progress, challenges, and future opportunities. Bioprinting 21:e00116. ISSN 24058866. doi:10.1016/j.bprint.2020.e00116. URL https://linkinghub.elsevier.com/retrieve/pii/S2405886620300439.

[54] Betancourt N and Chen X (2022) Review of extrusion-based multi-material bioprinting pro-

cesses. Bioprinting 25:e00189. ISSN 24058866. doi:10.1016/j.bprint.2021.e00189. URL https://linkinghub.elsevier.com/retrieve/pii/S2405886621000622.

[55] Kjar A, McFarland B, Mecham K, Harward N, and Huang Y (2021) Engineering of tissue constructs using coaxial bioprinting. Bioactive Materials 6(2):460–471. ISSN 2452199X. doi:10.1016/j.bioactmat.2020.08.020. URL https://linkinghub.elsevier.com/retrieve/pii/S2452199X20301936.

[56] Bova L, Billi F, and Cimetta E (2020) Mini-review: advances in 3D bioprinting of vascularized constructs. Biology Direct 15(1):22. ISSN 1745-6150. doi:10.1186/s13062-020-00273-4. URL https://biologydirect.biomedcentral.com/articles/10.1186/s13062-020-00273-4.

[57] Rocca M, Fragasso A, Liu W, Heinrich MA, and Zhang YS (2018) Embedded Multimaterial Extrusion Bioprinting. SLAS Technology 23(2):154–163. ISSN 24726303. doi:10.1177/2472630317742071. URL https://linkinghub.elsevier.com/retrieve/pii/S2472630322012286.

[58] Hospodiuk M, Dey M, Sosnoski D, and Ozbolat IT (2017) The bioink: A comprehensive review on bioprintable materials. Biotechnology Advances 35(2):217–239. ISSN 07349750. doi:10.1016/j.biotechadv.2016.12.006. URL https://linkinghub.elsevier.com/retrieve/pii/S0734975016301719.

[59] Zheng Z, Eglin D, Alini M, Richards GR, Qin L, and Lai Y (2021) Visible Light-Induced 3D Bioprinting Technologies and Corresponding Bioink Materials for Tissue Engineering: A Review. Engineering 7(7):966–978. ISSN 20958099. doi:10.1016/j.eng.2020.05.021. URL https://linkinghub.elsevier.com/retrieve/pii/S2095809920302496.

[60] Li W, Wang M, Ma H, Chapa-Villarreal FA, Lobo AO, and Zhang YS (2023) Stereolithography apparatus and digital light processing-based 3D bioprinting for tissue fabrication. iScience 26(2):106039. ISSN 25890042. doi:10.1016/j.isci.2023.106039. URL https://linkinghub.elsevier.com/retrieve/pii/S2589004223001165.

[61] Zhang F, Zhu L, Li Z, Wang S, Shi J, Tang W, Li N, and Yang J (2021) The recent development of vat photopolymerization: A review. Additive Manufacturing 48:102423. ISSN

22148604. doi:10.1016/j.addma.2021.102423. URL https://linkinghub.elsevier.com/retrieve/pii/S2214860421005753.

[62] Zandrini T, Florczak S, Levato R, and Ovsianikov A (2023) Breaking the resolution limits of 3D bioprinting: future opportunities and present challenges. Trends in Biotechnology 41(5):604–614. ISSN 01677799. doi:10.1016/j.tibtech.2022.10.009. URL https://linkinghub.elsevier.com/retrieve/pii/S0167779922002803.

[63] Miri AK, Nieto D, Iglesias L, Goodarzi Hosseinabadi H, Maharjan S, Ruiz-Esparza GU, Khoshakhlagh P, Manbachi A, Dokmeci MR, Chen S, Shin SR, Zhang YS, and Khademhosseini A (2018) Microfluidics-Enabled Multimaterial Maskless Stereolithographic Bioprinting. Advanced Materials 30(27):1800242. ISSN 09359648. doi:10.1002/adma.201800242. URL https://onlinelibrary.wiley.com/doi/10.1002/adma.201800242.

[64] Wu Y, Su H, Li M, and Xing H (2023) Digital light processing-based multi-material bioprinting: Processes, applications, and perspectives. Journal of Biomedical Materials Research Part A 111(4):527–542. ISSN 1549-3296, 1552-4965. doi:10.1002/jbm.a.37473. URL https://onlinelibrary.wiley.com/doi/10.1002/jbm.a.37473.

[65] Kowsari K, Akbari S, Wang D, Fang NX, and Ge Q (2018) High-Efficiency High-Resolution Multimaterial Fabrication for Digital Light Processing-Based Three-Dimensional Printing. 3D Printing and Additive Manufacturing 5(3):185–193. ISSN 2329-7662, 2329-7670. doi:10.1089/3dp.2018.0004. URL https://www.liebertpub.com/doi/10.1089/3dp.2018.0004.

[66] Schwartz JJ and Boydston AJ (2019) Multimaterial actinic spatial control 3D and 4D printing. Nature Communications 10(1):791. ISSN 2041-1723. doi:10.1038/s41467-019-08639-7. URL https://www.nature.com/articles/s41467-019-08639-7.

[67] Antoshin A, Churbanov S, Minaev N, Zhang D, Zhang Y, Shpichka A, and Timashev P (2019) LIFT-bioprinting, is it worth it? Bioprinting 15:e00052. ISSN 24058866. doi:10.1016/j.bprint.2019.e00052. URL https://linkinghub.elsevier.com/retrieve/pii/S240588661930017X.

[68] Landerneau S, Lemarié L, Marquette C, and Petiot E (2022) Green 3D bioprinting of plant cells: A

new scope for 3D bioprinting. Bioprinting 27:e00216. ISSN 24058866. doi:10.1016/j.bprint.2022.e00216. URL https://linkinghub.elsevier.com/retrieve/pii/S2405886622000264.

[69] Bejleri D, Streeter BW, Nachlas ALY, Brown ME, Gaetani R, Christman KL, and Davis ME (2018) A Bioprinted Cardiac Patch Composed of Cardiac-Specific Extracellular Matrix and Progenitor Cells for Heart Repair. Advanced Healthcare Materials 7(23):1800672. ISSN 2192-2640, 2192-2659. doi:10.1002/adhm.201800672. URL https://onlinelibrary.wiley.com/doi/10.1002/adhm.201800672.

[70] Tang M, Tiwari SK, Agrawal K, Tan M, Dang J, Tam T, Tian J, Wan X, Schimelman J, You S, Xia Q, Rana TM, and Chen S (2021) Rapid 3D Bioprinting of Glioblastoma Model Mimicking Native Biophysical Heterogeneity. Small 17(15):2006050. ISSN 1613-6810, 1613-6829. doi:10.1002/smll.202006050. URL https://onlinelibrary.wiley.com/doi/10.1002/smll.202006050.

[71] Khoshnood N and Zamanian A (2020) A comprehensive review on scaffold-free bioinks for bioprinting. Bioprinting 19:e00088. ISSN 24058866. doi:10.1016/j.bprint.2020.e00088. URL https://linkinghub.elsevier.com/retrieve/pii/S2405886620300154.

[72] Lee M, Rizzo R, Surman F, and Zenobi-Wong M (2020) Guiding Lights: Tissue Bioprinting Using Photoactivated Materials. Chemical Reviews 120(19):10950–11027. ISSN 0009-2665, 1520-6890. doi:10.1021/acs.chemrev.0c00077. URL https://pubs.acs.org/doi/10.1021/acs.chemrev.0c00077.

[73] Lim KS, Galarraga JH, Cui X, Lindberg GCJ, Burdick JA, and Woodfield TBF (2020) Fundamentals and Applications of Photo-Cross-Linking in Bioprinting. Chemical Reviews 120(19):10662–10694. ISSN 0009-2665, 1520-6890. doi:10.1021/acs.chemrev.9b00812. URL https://pubs.acs.org/doi/10.1021/acs.chemrev.9b00812.

[74] Pagac M, Hajnys J, Ma QP, Jancar L, Jansa J, Stefek P, and Mesicek J (2021) A Review of Vat Photopolymerization Technology: Materials, Applications, Challenges, and Future Trends of 3D Printing. Polymers 13(4):598. ISSN 2073-4360. doi:10.3390/polym13040598. URL https://www.mdpi.com/2073-4360/13/4/598.

[75] Tseytlin O, O'Connell R, Sivashankar V, Bobko AA, and Tseytlin M (2021) Rapid Scan EPR

Oxygen Imaging in Photoactivated Resin Used for Stereolithographic 3D Printing. 3D Printing and Additive Manufacturing 8(6):358–365. ISSN 2329-7662, 2329-7670. doi:10.1089/3dp.2020.0170. URL https://www.liebertpub.com/doi/10.1089/3dp.2020.0170.

[76] De Melo BA, Jodat YA, Cruz EM, Benincasa JC, Shin SR, and Porcionatto MA (2020) Strategies to use fibrinogen as bioink for 3D bioprinting fibrin-based soft and hard tissues. Acta Biomaterialia 117:60–76. ISSN 17427061. doi:10.1016/j.actbio.2020.09.024. URL https://linkinghub.elsevier.com/retrieve/pii/S1742706120305389.

[77] Maiz-Fernández S, Pérez-Álvarez L, Silván U, Vilas-Vilela JL, and Lanceros-Méndez S (2022) pH-Induced 3D Printable Chitosan Hydrogels for Soft Actuation. Polymers 14(3):650. ISSN 2073-4360. doi:10.3390/polym14030650. URL https://www.mdpi.com/2073-4360/14/3/650.

[78] Lee JM, Suen SKQ, Ng WL, Ma WC, and Yeong WY (2021) Bioprinting of Collagen: Considerations, Potentials, and Applications. Macromolecular Bioscience 21(1):2000280. ISSN 1616-5187, 1616-5195. doi:10.1002/mabi.202000280. URL https://onlinelibrary.wiley.com/doi/10.1002/mabi.202000280.

[79] Ke D, Kengla C, Lee SJ, Yoo JJ, Zhu X, and Murphy SV (2022) Release Kinetics and In Vitro Characterization of Sodium Percarbonate and Calcium Peroxide to Oxygenate Bioprinted Tissue Models. International Journal of Molecular Sciences 23(12):6842. ISSN 1422-0067. doi:10.3390/ijms23126842. URL https://www.mdpi.com/1422-0067/23/12/6842.

[80] Gholipourmalekabadi M, Zhao S, Harrison BS, Mozafari M, and Seifalian AM (2016) Oxygen-Generating Biomaterials: A New, Viable Paradigm for Tissue Engineering? Trends in Biotechnology 34(12):1010–1021. ISSN 01677799. doi:10.1016/j.tibtech.2016.05.012. URL https://linkinghub.elsevier.com/retrieve/pii/S0167779916300580.

[81] Alfonso-Prieto M, Biarnés X, Vidossich P, and Rovira C (2009) The Molecular Mechanism of the Catalase Reaction. Journal of the American Chemical Society 131(33):11751–11761. ISSN 0002-7863, 1520-5126. doi:10.1021/ja9018572. URL https://pubs.acs.org/doi/10.1021/ja9018572.

[82] Mortimer CH (1956) The oxygen content of air-saturated fresh waters, and aids in calculating

percentage saturation: With 2 tables and 5 figures in the text. SIL Communications, 1953-1996 6(1):1–20. ISSN 0538-4680. doi:10.1080/05384680.1956.11904088. URL https://www.tandfonline.com/doi/full/10.1080/05384680.1956.11904088.

[83] Thomson L and Paton J (2014) Oxygen Toxicity. Paediatric Respiratory Reviews 15(2):120–123. ISSN 15260542. doi:10.1016/j.prrv.2014.03.003. URL https://linkinghub.elsevier.com/retrieve/pii/S152605421400030X.

[84] Ortega JS, Corrales-Orovio R, Ralph P, Egaña JT, and Gentile C (2022) Photosynthetic microorganisms for the oxygenation of advanced 3D bioprinted tissues. Acta Biomaterialia S1742706122002781. ISSN 17427061. doi:10.1016/j.actbio.2022.05.009. URL https://linkinghub.elsevier.com/retrieve/pii/S1742706122002781.

[85] Maharjan S, Alva J, Cámara C, Rubio AG, Hernández D, Delavaux C, Correa E, Romo MD, Bonilla D, Santiago ML, Li W, Cheng F, Ying G, and Zhang YS (2021) Symbiotic Photosynthetic Oxygenation within 3D-Bioprinted Vascularized Tissues. Matter 4(1):217–240. ISSN 25902385. doi:10.1016/j.matt.2020.10.022. URL https://linkinghub.elsevier.com/retrieve/pii/S2590238520305762.

[86] Zhu J, Wang Y, Zhong L, Pan F, and Wang J (2021) Advances in tissue engineering of vasculature through three-dimensional bioprinting. Developmental Dynamics 250(12):1717–1738. ISSN 1058-8388, 1097-0177. doi:10.1002/dvdy.385. URL https://onlinelibrary.wiley.com/doi/10.1002/dvdy.385.

[87] Biela SA, Su Y, Spatz JP, and Kemkemer R (2009) Different sensitivity of human endothelial cells, smooth muscle cells and fibroblasts to topography in the nano–micro range. Acta Biomaterialia 5(7):2460–2466. ISSN 17427061. doi:10.1016/j.actbio.2009.04.003. URL https://linkinghub.elsevier.com/retrieve/pii/S1742706109001512.

[88] Xu L, Varkey M, Jorgensen A, Ju J, Jin Q, Park JH, Fu Y, Zhang G, Ke D, Zhao W, Hou R, and Atala A (2020) Bioprinting small diameter blood vessel constructs with an endothelial and smooth muscle cell bilayer in a single step. Biofabrication 12(4):045012. ISSN 1758-5090. doi:10.1088/1758-5090/aba2b6. URL https://iopscience.iop.org/article/10.1088/1758-5090/aba2b6.

[89] Zhou X, Nowicki M, Sun H, Hann SY, Cui H, Esworthy T, Lee JD, Plesniak M, and Zhang LG (2020) 3D Bioprinting-Tunable Small-Diameter Blood Vessels with Biomimetic Biphasic Cell Layers. ACS Applied Materials & Interfaces 12(41):45904–45915. ISSN 1944-8244, 1944-8252. doi:10.1021/acsami.0c14871. URL https://pubs.acs.org/doi/10.1021/acsami.0c14871.

[90] Taniguchi D, Matsumoto K, Machino R, Takeoka Y, Elgalad A, Taura Y, Oyama S, Tetsuo T, Moriyama M, Takagi K, Kunizaki M, Tsuchiya T, Miyazaki T, Hatachi G, Matsuo N, Nakayama K, and Nagayasu T (2020) Human lung microvascular endothelial cells as potential alternatives to human umbilical vein endothelial cells in bio-3D-printed trachea-like structures. Tissue and Cell 63:101321. ISSN 00408166. doi:10.1016/j.tice.2019.101321. URL https://linkinghub.elsevier.com/retrieve/pii/S0040816619303660.

[91] Aird WC (2012) Endothelial Cell Heterogeneity. Cold Spring Harbor Perspectives in Medicine 2(1):a006429–a006429. ISSN 2157-1422. doi:10.1101/cshperspect.a006429. URL http://perspectivesinmedicine.cshlp.org/lookup/doi/10.1101/cshperspect.a006429.

[92] Tijore A, Behr JM, Irvine SA, Baisane V, and Venkatraman S (2018) Bioprinted gelatin hydrogel platform promotes smooth muscle cell contractile phenotype maintenance. Biomedical Microdevices 20(2):32. ISSN 1387-2176, 1572-8781. doi:10.1007/s10544-018-0274-8. URL http://link.springer.com/10.1007/s10544-018-0274-8.

[93] Norotte C, Marga FS, Niklason LE, and Forgacs G (2009) Scaffold-free vascular tissue engineering using bioprinting. Biomaterials 30(30):5910–5917. ISSN 01429612. doi:10.1016/j.biomaterials.2009.06.034. URL https://linkinghub.elsevier.com/retrieve/pii/S0142961209006401.

[94] Kinstlinger IS and Miller JS (2016) 3D-printed fluidic networks as vasculature for engineered tissue. Lab on a Chip 16(11):2025–2043. ISSN 1473-0197, 1473-0189. doi:10.1039/C6LC00193A. URL http://xlink.rsc.org/?DOI=C6LC00193A.

[95] Poole DC, Copp SW, Ferguson SK, and Musch TI (2013) Skeletal muscle capillary function: contemporary observations and novel hypotheses: Skeletal muscle capillary function. Experimental Physiology 98(12):1645–1658. ISSN 09580670. doi:10.1113/expphysiol.2013.073874. URL http://doi.wiley.com/10.1113/expphysiol.2013.073874.

[96] Kolesky DB, Homan KA, Skylar-Scott MA, and Lewis JA (2016) Three-dimensional bioprinting of thick vascularized tissues. Proceedings of the National Academy of Sciences 113(12):3179–3184. ISSN 0027-8424, 1091-6490. doi:10.1073/pnas.1521342113. URL https://pnas.org/doi/full/10.1073/pnas.1521342113.

[97] Arakawa C, Gunnarsson C, Howard C, Bernabeu M, Phong K, Yang E, DeForest CA, Smith JD, and Zheng Y (2020) Biophysical and biomolecular interactions of malaria-infected erythrocytes in engineered human capillaries. Science Advances 6(3):eaay7243. ISSN 2375-2548. doi:10.1126/sciadv.aay7243. URL https://www.science.org/doi/10.1126/sciadv.aay7243.

[98] Zhu W, Qu X, Zhu J, Ma X, Patel S, Liu J, Wang P, Lai CSE, Gou M, Xu Y, Zhang K, and Chen S (2017) Direct 3D bioprinting of prevascularized tissue constructs with complex microarchitecture. Biomaterials 124:106–115. ISSN 01429612. doi:10.1016/j.biomaterials.2017.01.042. URL https://linkinghub.elsevier.com/retrieve/pii/S0142961217300650.

[99] Meyer W, Engelhardt S, Novosel E, Elling B, Wegener M, and Krüger H (2012) Soft Polymers for Building up Small and Smallest Blood Supplying Systems by Stereolithography. Journal of Functional Biomaterials 3(2):257–268. ISSN 2079-4983. doi:10.3390/jfb3020257. URL http://www.mdpi.com/2079-4983/3/2/257.

[100] Kolesky DB, Truby RL, Gladman AS, Busbee TA, Homan KA, and Lewis JA (2014) 3D Bioprinting of Vascularized, Heterogeneous Cell-Laden Tissue Constructs. Advanced Materials 26(19):3124–3130. ISSN 09359648. doi:10.1002/adma.201305506. URL https://onlinelibrary.wiley.com/doi/10.1002/adma.201305506.

[101] Khalighi S and Saadatmand M (2021) Bioprinting a thick and cell-laden partially oxidized alginate-gelatin scaffold with embedded micro-channels as future soft tissue platform. International Journal of Biological Macromolecules 193:2153–2164. ISSN 01418130. doi:10.1016/j.ijbiomac.2021.11.046. URL https://linkinghub.elsevier.com/retrieve/pii/S0141813021024442.

[102] Lee W, Lee V, Polio S, Keegan P, Lee JH, Fischer K, Park JK, and Yoo SS (2010) On-demand three-dimensional freeform fabrication of multi-layered hydrogel scaffold with fluidic channels. Biotechnology and Bioengineering n/a–n/a. ISSN 00063592, 10970290. doi:10.1002/bit.22613. URL https://onlinelibrary.wiley.com/doi/10.1002/bit.22613.

[103] Pi Q, Maharjan S, Yan X, Liu X, Singh B, Van Genderen AM, Robledo-Padilla F, Parra-Saldivar R, Hu N, Jia W, Xu C, Kang J, Hassan S, Cheng H, Hou X, Khademhosseini A, and Zhang YS (2018) Digitally Tunable Microfluidic Bioprinting of Multilayered Cannular Tissues. Advanced Materials 30(43):1706913. ISSN 09359648. doi:10.1002/adma.201706913. URL https://onlinelibrary.wiley.com/doi/10.1002/adma.201706913.

[104] Gharacheh H and Guvendiren M (2023) Three-dimensional bioprinting vascularized bone tissue. MRS Bulletin ISSN 0883-7694, 1938-1425. doi:10.1557/s43577-023-00547-y. URL https://link.springer.com/10.1557/s43577-023-00547-y.

[105] Mir A, Lee E, Shih W, Koljaka S, Wang A, Jorgensen C, Hurr R, Dave A, Sudheendra K, and Hibino N (2023) 3D Bioprinting for Vascularization. Bioengineering 10(5):606. ISSN 2306-5354. doi:10.3390/bioengineering10050606. URL https://www.mdpi.com/2306-5354/10/5/606.

[106] Anthon SG and Valente KP (2022) Vascularization Strategies in 3D Cell Culture Models: From Scaffold-Free Models to 3D Bioprinting. International Journal of Molecular Sciences 23(23):14582. ISSN 1422-0067. doi:10.3390/ijms232314582. URL https://www.mdpi.com/1422-0067/23/23/14582.

[107] Grigoryan B, Paulsen SJ, Corbett DC, Sazer DW, Fortin CL, Zaita AJ, Greenfield PT, Calafat NJ, Gounley JP, Ta AH, Johansson F, Randles A, Rosenkrantz JE, Louis-Rosenberg JD, Galie PA, Stevens KR, and Miller JS (2019) Multivascular networks and functional intravascular topologies within biocompatible hydrogels. Science 364(6439):458–464. ISSN 0036-8075, 1095-9203. doi:10.1126/science.aav9750. URL https://www.science.org/doi/10.1126/science.aav9750.

[108] Sekine W and Haraguchi Y (2011) Thickness limitation and cell viability of multi-layered cell sheets and overcoming the diffusion limit by a porous-membrane culture insert. Journal of Biochips & Tissue Chips 01(01). ISSN 21530777. doi:10.4172/2153-0777.S1-007. URL https://www.omicsonline.org/thickness-limitation-and-cell-viability-of-multilayered-cell-sheets-and-overcoming-the-d:php?aid=9796.

[109] Fritschen A, Acedo Mestre M, Scholpp S, and Blaeser A (2023) Influence of the physico-chemical bioink composition on the printability and cell biological properties in 3D-bioprinting of a liver

tumor cell line. Frontiers in Bioengineering and Biotechnology 11:1093101. ISSN 2296-4185. doi:10.3389/fbioe.2023.1093101. URL https://www.frontiersin.org/articles/10.3389/fbioe.2023.1093101/full.

[110] Wang H, Liu X, Gu Q, and Zheng X (2023) Vascularized organ bioprinting: From strategy to paradigm. Cell Proliferation 56(5):e13453. ISSN 0960-7722, 1365-2184. doi:10.1111/cpr.13453. URL https://onlinelibrary.wiley.com/doi/10.1111/cpr.13453.

[111] Kumar H, Dixit K, Sharma R, MacDonald ME, Sinha N, and Kim K (2023) Closed-loop vasculature network design for bioprinting large, solid tissue scaffolds. Biofabrication 15(2):024104. ISSN 1758-5082, 1758-5090. doi:10.1088/1758-5090/acb73c. URL https://iopscience.iop.org/article/10.1088/1758-5090/acb73c.

[112] Gao B, Yang Q, Zhao X, Jin G, Ma Y, and Xu F (2016) 4D Bioprinting for Biomedical Applications. Trends in Biotechnology 34(9):746–756. ISSN 01677799. doi:10.1016/j.tibtech.2016.03.004. URL https://linkinghub.elsevier.com/retrieve/pii/S0167779916000664.

[113] Zhang J, Wehrle E, Rubert M, and Müller R (2021) 3D Bioprinting of Human Tissues: Biofabrication, Bioinks, and Bioreactors. International Journal of Molecular Sciences 22(8):3971. ISSN 1422-0067. doi:10.3390/ijms22083971. URL https://www.mdpi.com/1422-0067/22/8/3971.

[114] Stiehler M, Bünger C, Baatrup A, Lind M, Kassem M, and Mygind T (2008) Effect of dynamic 3-D culture on proliferation, distribution, and osteogenic differentiation of human mesenchymal stem cells. Journal of Biomedical Materials Research Part A ISSN 15493296, 15524965. doi:10.1002/jbm.a.31967. URL https://onlinelibrary.wiley.com/doi/10.1002/jbm.a.31967.

[115] Bernal PN, Bouwmeester M, Madrid-Wolff J, Falandt M, Florczak S, Rodriguez NG, Li Y, Größbacher G, Samsom R, Van Wolferen M, Van Der Laan LJW, Delrot P, Loterie D, Malda J, Moser C, Spee B, and Levato R (2022) Volumetric Bioprinting of Organoids and Optically Tuned Hydrogels to Build Liver-Like Metabolic Biofactories. Advanced Materials 34(15):2110054. ISSN 0935-9648, 1521-4095. doi:10.1002/adma.202110054. URL https://onlinelibrary.wiley.com/doi/10.1002/adma.202110054.

[116] Doolin EJ, Geldziler B, Strande L, Kain M, and Hewitt C (1999) Effects of Microgravity on

Growing Cultured Skin Constructs. Tissue Engineering 5(6):573–581. ISSN 1076-3279, 1557-8690. doi:10.1089/ten.1999.5.573. URL https://www.liebertpub.com/doi/10.1089/ten.1999.5.573.

[117] Bursac N, Loo Y, Leong K, and Tung L (2007) Novel anisotropic engineered cardiac tissues: Studies of electrical propagation. Biochemical and Biophysical Research Communications 361(4):847–853. ISSN 0006291X. doi:10.1016/j.bbrc.2007.07.138. URL https://linkinghub.elsevier.com/retrieve/pii/S0006291X07014921.

[118] Teoh JH, Thamizhchelvan AM, Davoodi P, Ramasamy S, Vijayavenkataraman S, Yang Q, Dicolandrea T, Zhao H, Fuh JY, Liou YC, and Wang CH (2019) Investigation of the application of a Taylor-Couette bioreactor in the post-processing of bioprinted human dermal tissue. Biochemical Engineering Journal 151:107317. ISSN 1369703X. doi:10.1016/j.bej.2019.107317. URL https://linkinghub.elsevier.com/retrieve/pii/S1369703X19302529.

[119] Zhang J, Griesbach J, Ganeyev M, Zehnder AK, Zeng P, Schädli GN, Leeuw AD, Lai Y, Rubert M, and Müller R (2022) Long-term mechanical loading is required for the formation of 3D bioprinted functional osteocyte bone organoids. Biofabrication 14(3):035018. ISSN 1758-5082, 1758-5090. doi:10.1088/1758-5090/ac73b9. URL https://iopscience.iop.org/article/10.1088/1758-5090/ac73b9.

[120] Zhang YS, Arneri A, Bersini S, Shin SR, Zhu K, Goli-Malekabadi Z, Aleman J, Colosi C, Busignani F, Dell'Erba V, Bishop C, Shupe T, Demarchi D, Moretti M, Rasponi M, Dokmeci MR, Atala A, and Khademhosseini A (2016) Bioprinting 3D microfibrous scaffolds for engineering endothelialized myocardium and heart-on-a-chip. Biomaterials 110:45–59. ISSN 01429612. doi:10.1016/j.biomaterials.2016.09.003. URL https://linkinghub.elsevier.com/retrieve/pii/S014296121630477X.

[121] Sego TJ, Prideaux M, Sterner J, McCarthy BP, Li P, Bonewald LF, Ekser B, Tovar A, and Jeshua Smith L (2020) Computational fluid dynamic analysis of bioprinted self-supporting perfused tissue models. Biotechnology and Bioengineering 117(3):798–815. ISSN 0006-3592, 1097-0290. doi:10.1002/bit.27238. URL https://onlinelibrary.wiley.com/doi/10.1002/bit.27238.

[122] Kuo CY, Shevchuk M, Opfermann J, Guo T, Santoro M, Fisher JP, and Kim PC (2019) Trophoblast-endothelium signaling involves angiogenesis and apoptosis in a dynamic bioprinted placenta model. Biotechnology and Bioengineering 116(1):181–192. ISSN 00063592. doi: 10.1002/bit.26850. URL https://onlinelibrary.wiley.com/doi/10.1002/bit.26850.

[123] Chen AM, Lashmet M, Isidan A, Sterner JL, Walsh J, Koehler C, Li P, Ekser B, and Smith L (2020) Oxygenation Profiles of Human Blood, Cell Culture Medium, and Water for Perfusion of 3D-Bioprinted Tissues using the FABRICA Bioreactor Platform. Scientific Reports 10(1):7237. ISSN 2045-2322. doi:10.1038/s41598-020-64256-1. URL https://www.nature.com/articles/s41598-020-64256-1.

[124] O'Mara P, Farrell A, Bones J, and Twomey K (2018) Staying alive! Sensors used for monitoring cell health in bioreactors. Talanta 176:130–139. ISSN 00399140. doi:10.1016/j.talanta.2017.07.088. URL https://linkinghub.elsevier.com/retrieve/pii/S0039914017308093.

[125] Kaur B, Kumar S, and Kaushik BK (2022) Recent advancements in optical biosensors for cancer detection. Biosensors and Bioelectronics 197:113805. ISSN 09565663. doi:10.1016/j.bios.2021.113805. URL https://linkinghub.elsevier.com/retrieve/pii/S0956566321008423.

[126] Gandhi M, Chu S, Senthilnathan K, Babu P, Nakkeeran K, and Li Q (2019) Recent Advances in Plasmonic Sensor-Based Fiber Optic Probes for Biological Applications. Applied Sciences 9(5):949. ISSN 2076-3417. doi:10.3390/app9050949. URL https://www.mdpi.com/2076-3417/9/5/949.

[127] Yoo EH and Lee SY (2010) Glucose Biosensors: An Overview of Use in Clinical Practice. Sensors 10(5):4558–4576. ISSN 1424-8220. doi:10.3390/s100504558. URL http://www.mdpi.com/1424-8220/10/5/4558.

[128] Iannazzo D, Espro C, Celesti C, Ferlazzo A, and Neri G (2021) Smart Biosensors for Cancer Diagnosis Based on Graphene Quantum Dots. Cancers 13(13):3194. ISSN 2072-6694. doi:10.3390/cancers13133194. URL https://www.mdpi.com/2072-6694/13/13/3194.

[129] Mowbray S and Amiri A (2019) A Brief Overview of Medical Fiber Optic Biosensors and Techniques

in the Modification for Enhanced Sensing Ability. Diagnostics 9(1):23. ISSN 2075-4418. doi: 10.3390/diagnostics9010023. URL http://www.mdpi.com/2075-4418/9/1/23.

[130] Alemdar N, Leijten J, Camci-Unal G, Hjortnaes J, Ribas J, Paul A, Mostafalu P, Gaharwar AK, Qiu Y, Sonkusale S, Liao R, and Khademhosseini A (2017) Oxygen-Generating Photo-Cross-Linkable Hydrogels Support Cardiac Progenitor Cell Survival by Reducing Hypoxia-Induced Necrosis. ACS Biomaterials Science & Engineering 3(9):1964–1971. ISSN 2373-9878, 2373-9878. doi:10.1021/acsbiomaterials.6b00109. URL https://pubs.acs.org/doi/10.1021/acsbiomaterials.6b00109.

[131] Figueiredo L, Le Visage C, Weiss P, and Yang J (2020) Quantifying Oxygen Levels in 3D Bioprinted Cell-Laden Thick Constructs with Perfusable Microchannel Networks. Polymers 12(6):1260. ISSN 2073-4360. doi:10.3390/polym12061260. URL https://www.mdpi.com/2073-4360/12/6/1260.

[132] Campioni G, Pasquale V, Busti S, Ducci G, Sacco E, and Vanoni M (2022) An Optimized Workflow for the Analysis of Metabolic Fluxes in Cancer Spheroids Using Seahorse Technology. Cells 11(5):866. ISSN 2073-4409. doi:10.3390/cells11050866. URL https://www.mdpi.com/2073-4409/11/5/866.

[133] García-Lizarribar A, Villasante A, Lopez-Martin JA, Flandez M, Soler-Vázquez MC, Serra D, Herrero L, Sagrera A, Efeyan A, and Samitier J (2023) 3D bioprinted functional skeletal muscle models have potential applications for studies of muscle wasting in cancer cachexia. Biomaterials Advances 150:213426. ISSN 27729508. doi:10.1016/j.bioadv.2023.213426. URL https://linkinghub.elsevier.com/retrieve/pii/S2772950823001498.

[134] Acosta MA, Ymele-Leki P, Kostov YV, and Leach JB (2009) Fluorescent microparticles for sensing cell microenvironment oxygen levels within 3D scaffolds. Biomaterials 30(17):3068–3074. ISSN 01429612. doi:10.1016/j.biomaterials.2009.02.021. URL https://linkinghub.elsevier.com/retrieve/pii/S0142961209001811.

[135] Wu L, Sedgwick AC, Sun X, Bull SD, He XP, and James TD (2019) Reaction-Based Fluorescent Probes for the Detection and Imaging of Reactive Oxygen, Nitrogen, and Sulfur Species. Accounts

of Chemical Research 52(9):2582–2597. ISSN 0001-4842, 1520-4898. doi:10.1021/acs.accounts.9b00302. URL https://pubs.acs.org/doi/10.1021/acs.accounts.9b00302.

[136] Wang S, Ren WX, Hou JT, Won M, An J, Chen X, Shu J, and Kim JS (2021) Fluorescence imaging of pathophysiological microenvironments. Chemical Society Reviews 50(16):8887–8902. ISSN 0306-0012, 1460-4744. doi:10.1039/D1CS00083G. URL http://xlink.rsc.org/?DOI=D1CS00083G.

[137] Jonkman J, Brown CM, Wright GD, Anderson KI, and North AJ (2020) Tutorial: guidance for quantitative confocal microscopy. Nature Protocols 15(5):1585–1611. ISSN 1754-2189, 1750-2799. doi:10.1038/s41596-020-0313-9. URL https://www.nature.com/articles/s41596-020-0313-9.

[138] Liu P, Mu X, Zhang XD, and Ming D (2020) The Near-Infrared-II Fluorophores and Advanced Microscopy Technologies Development and Application in Bioimaging. Bioconjugate Chemistry 31(2):260–275. ISSN 1043-1802, 1520-4812. doi:10.1021/acs.bioconjchem.9b00610. URL https://pubs.acs.org/doi/10.1021/acs.bioconjchem.9b00610.

[139] Erdem A, Darabi MA, Nasiri R, Sangabathuni S, Ertas YN, Alem H, Hosseini V, Shamloo A, Nasr AS, Ahadian S, Dokmeci MR, Khademhosseini A, and Ashammakhi N (2020) 3D Bioprinting of Oxygenated Cell-Laden Gelatin Methacryloyl Constructs. Advanced Healthcare Materials 9(15):1901794. ISSN 2192-2640, 2192-2659. doi:10.1002/adhm.201901794. URL https://onlinelibrary.wiley.com/doi/10.1002/adhm.201901794.

[140] Okkelman IA, Vercruysse C, Kondrashina AV, Borisov SM, and Dmitriev RI (2022) Affordable Oxygen Microscopy-Assisted Biofabrication of Multicellular Spheroids. Journal of Visualized Experiments (182):63403. ISSN 1940-087X. doi:10.3791/63403. URL https://www.jove.com/t/63403/affordable-oxygen-microscopy-assisted-biofabrication-multicellular.

[141] Trampe E, Koren K, Akkineni AR, Senwitz C, Krujatz F, Lode A, Gelinsky M, and Kühl M (2018) Functionalized Bioink with Optical Sensor Nanoparticles for $O_2$ Imaging in 3D-Bioprinted Constructs. Advanced Functional Materials 28(45):1804411. ISSN 1616301X. doi:10.1002/adfm.201804411. URL https://onlinelibrary.wiley.com/doi/10.1002/adfm.201804411.

[142] Schermelleh L, Ferrand A, Huser T, Eggeling C, Sauer M, Biehlmaier O, and Drummen GPC (2019) Super-resolution microscopy demystified. Nature Cell Biology 21(1):72–84. ISSN 1465-7392, 1476-4679. doi:10.1038/s41556-018-0251-8. URL https://www.nature.com/articles/s41556-018-0251-8.

[143] Ntziachristos V (2010) Going deeper than microscopy: the optical imaging frontier in biology. Nature Methods 7(8):603–614. ISSN 1548-7091, 1548-7105. doi:10.1038/nmeth.1483. URL https://www.nature.com/articles/nmeth.1483.

[144] O'Connor JPB, Robinson SP, and Waterton JC (2019) Imaging tumour hypoxia with oxygen-enhanced MRI and BOLD MRI. The British Journal of Radiology 92(1096):20180642. ISSN 0007-1285, 1748-880X. doi:10.1259/bjr.20180642. URL https://www.birpublications.org/doi/10.1259/bjr.20180642.

[145] Lu H, Chen A, Zhang X, Wei Z, Cao R, Zhu Y, Lu J, Wang Z, and Tian L (2022) A pH-responsive T1-T2 dual-modal MRI contrast agent for cancer imaging. Nature Communications 13(1):7948. ISSN 2041-1723. doi:10.1038/s41467-022-35655-x. URL https://www.nature.com/articles/s41467-022-35655-x.

[146] Li X, Liu Y, Qi X, Xiao S, Xu Z, Yuan Z, Liu Q, Li H, Ma S, Liu T, Huang Y, Zhang X, Zhang X, Mao Z, Luo G, and Deng J (2022) Sensitive Activatable Nanoprobes for Real-Time Ratiometric Magnetic Resonance Imaging of Reactive Oxygen Species and Ameliorating Inflammation In Vivo. Advanced Materials 34(19):2109004. ISSN 0935-9648, 1521-1095. doi:10.1002/adma.202109004. URL https://onlinelibrary.wiley.com/doi/10.1002/adma.202109004.

[147] Schmieg B, Gretzinger S, Schuhmann S, Guthausen G, and Hubbuch J (2022) Magnetic resonance imaging as a tool for quality control in extrusion-based bioprinting. Biotechnology Journal 17(5):2100336. ISSN 1860-6768, 1860-7314. doi:10.1002/biot.202100336. URL https://onlinelibrary.wiley.com/doi/10.1002/biot.202100336.

[148] Kérourédan O, Ribot EJ, Fricain JC, Devillard R, and Miraux S (2018) Magnetic Resonance Imaging for tracking cellular patterns obtained by Laser-Assisted Bioprinting. Scientific Reports 8(1):15777. ISSN 2045-2322. doi:10.1038/s41598-018-34226-9. URL https://www.nature.com/articles/s41598-018-34226-9.

[149] Loai S, Szulc DA, and Cheng HM (2022) Three-Dimensional Bioprinted MR-Trackable Regenerative Scaffold for Postimplantation Monitoring on T1-Weighted MRI. Journal of Magnetic Resonance Imaging 56(2):570–578. ISSN 1053-1807, 1522-2586. doi:10.1002/jmri.28057. URL https://onlinelibrary.wiley.com/doi/10.1002/jmri.28057.

[150] Amoroso M, Apelgren P, Säljö K, Montelius M, Orrhult LS, Engström M, Gatenholm P, and Kölby L (2021) Functional and morphological studies of in vivo vascularization of 3D-bioprinted human fat grafts. Bioprinting 23:e00162. ISSN 24058866. doi:10.1016/j.bprint.2021.e00162. URL https://linkinghub.elsevier.com/retrieve/pii/S240588662100035X.

[151] Apelgren P, Amoroso M, Säljö K, Montelius M, Lindahl A, Stridh Orrhult L, Gatenholm P, and Kölby L (2021) Vascularization of tissue engineered cartilage - Sequential in vivo MRI display functional blood circulation. Biomaterials 276:121002. ISSN 01429612. doi: 10.1016/j.biomaterials.2021.121002. URL https://linkinghub.elsevier.com/retrieve/pii/S0142961221003586.

[152] Zhou H, Chiguru S, Hallac RR, Yang D, Hao G, Peschke P, and Mason RP (2019) Examining correlations of oxygen sensitive MRI (BOLD/TOLD) with [18F]FMISO PET in rat prostate tumors. American Journal of Nuclear Medicine and Molecular Imaging 9(2):156–167. ISSN 2160-8407.

[153] White DA, Zhang Z, Li L, Gerberich J, Stojadinovic S, Peschke P, and Mason RP (2016) Developing oxygen-enhanced magnetic resonance imaging as a prognostic biomarker of radiation response. Cancer Letters 380(1):69–77. ISSN 03043835. doi:10.1016/j.canlet.2016.06.003. URL https://linkinghub.elsevier.com/retrieve/pii/S0304383516303603.

[154] Salem A, Little RA, Latif A, Featherstone AK, Babur M, Peset I, Cheung S, Watson Y, Tessyman V, Mistry H, Ashton G, Behan C, Matthews JC, Asselin MC, Bristow RG, Jackson A, Parker GJ, Faivre-Finn C, Williams KJ, and O'Connor JP (2019) Oxygen-enhanced MRI Is Feasible, Repeatable, and Detects Radiotherapy-induced Change in Hypoxia in Xenograft Models and in Patients with Non–small Cell Lung Cancer. Clinical Cancer Research 25(13):3818–3829. ISSN 1078-0432, 1557-3265. doi:10.1158/1078-0432. CCR-18-3932. URL https://aacrjournals.org/clincancerres/article/25/13/3818/81902/Oxygen-enhanced-MRI-Is-Feasible-Repeatable-and.

[155] Matsumoto Ki, Bernardo M, Subramanian S, Choyke P, Mitchell JB, Krishna MC, and Lizak MJ (2006) MR assessment of changes of tumor in response to hyperbaric oxygen treatment. Magnetic Resonance in Medicine 56(2):240–246. ISSN 0740-3194, 1522-2594. doi:10.1002/mrm.20961. URL https://onlinelibrary.wiley.com/doi/10.1002/mrm.20961.

[156] Basu S, Kwee TC, Surti S, Akin EA, Yoo D, and Alavi A (2011) Fundamentals of PET and PET/CT imaging: Fundamentals of PET and PET/CT. Annals of the New York Academy of Sciences 1228(1):1–18. ISSN 00778923. doi:10.1111/j.1749-6632.2011.06077.x. URL https://onlinelibrary.wiley.com/doi/10.1111/j.1749-6632.2011.06077.x.

[157] Lee ST and Scott AM (2007) Hypoxia Positron Emission Tomography Imaging With 18F-Fluoromisonidazole. Seminars in Nuclear Medicine 37(6):451–461. ISSN 00012998. doi:10.1053/j.semnuclmed.2007.07.001. URL https://linkinghub.elsevier.com/retrieve/pii/S0001299807000840.

[158] Mintun MA, Raichle ME, Martin WR, and Herscovitch P (1984) Brain oxygen utilization measured with O-15 radiotracers and positron emission tomography. Journal of Nuclear Medicine: Official Publication, Society of Nuclear Medicine 25(2):177–187. ISSN 0161-5505.

[159] Shibuya K, Saito H, Nishikido F, Takahashi M, and Yamaya T (2020) Oxygen sensing ability of positronium atom for tumor hypoxia imaging. Communications Physics 3(1):173. ISSN 2399-3650. doi:10.1038/s42005-020-00440-z. URL https://www.nature.com/articles/s42005-020-00440-z.

[160] Holly TA, Abbott BG, Al-Mallah M, Calnon DA, Cohen MC, DiFilippo FP, Ficaro EP, Freeman MR, Hendel RC, Jain D, Leonard SM, Nichols KJ, Polk DM, and Soman P (2010) Single photon-emission computed tomography. Journal of Nuclear Cardiology 17(5):941–973. ISSN 1071-3581, 1532-6551. doi:10.1007/s12350-010-9246-y. URL http://link.springer.com/10.1007/s12350-010-9246-y.

[161] Dhawan RT, Gopalan D, Howard L, Vicente A, Park M, Manalan K, Wallner I, Marsden P, Dave S, Branley H, Russell G, Dharmarajah N, and Kon OM (2021) Beyond the clot: perfusion imaging of the pulmonary vasculature after COVID-19. The Lancet Respiratory Medicine 9(1):107–116.

ISSN 22132600. doi:10.1016/S2213-2600(20)30407-0. URL https://linkinghub.elsevier.com/retrieve/pii/S2213260020304070.

[162] Yoo SW, Jung H, Park J, Kim JH, Lee KS, and Kim JS (2023) Unilateral Myoclonus as an initial manifestation in Moyamoya disease with asymmetric vasculature. Clinical Neurology and Neurosurgery 225:107587. ISSN 03038467. doi:10.1016/j.clineuro.2023.107587. URL https://linkinghub.elsevier.com/retrieve/pii/S0303846723000033.

[163] Doyen M, Hossu G, Heyer S, Zaragori T, Imbert L, and Verger A (2023) Identification of resting-state networks using dynamic brain perfusion SPECT imaging: A fSPECT case report. Frontiers in Human Neuroscience 17:1125765. ISSN 1662-5161. doi:10.3389/fnhum.2023.1125765. URL https://www.frontiersin.org/articles/10.3389/fnhum.2023.1125765/full.

[164] Lienemann PS, Metzger S, Kiveliö AS, Blanc A, Papageorgiou P, Astolfo A, Pinzer BR, Cinelli P, Weber FE, Schibli R, Béhé M, and Ehrbar M (2015) Longitudinal in vivo evaluation of bone regeneration by combined measurement of multi-pinhole SPECT and micro-CT for tissue engineering. Scientific Reports 5(1):10238. ISSN 2045-2322. doi:10.1038/srep10238. URL https://www.nature.com/articles/srep10238.

[165] Ozsahin I, Chen L, Könik A, King MA, Beekman FJ, and Mok GSP (2020) The clinical utilities of multi-pinhole single photon emission computed tomography. Quantitative Imaging in Medicine and Surgery 10(10):2006–2029. ISSN 22234292, 22234306. doi:10.21037/qims-19-1036. URL http://qims.amegroups.com/article/view/50182/html.

[166] Kim H, Furenlid LR, Crawford MJ, Wilson DW, Barber HB, Peterson TE, Hunter WCJ, Liu Z, Woolfenden JM, and Barrett HH (2006) SemiSPECT: A small-animal single-photon emission computed tomography (SPECT) imager based on eight cadmium zinc telluride (CZT) detector arrays: A CZT SPECT system. Medical Physics 33(2):465–474. ISSN 00942405. doi:10.1118/1.2164070. URL http://doi.wiley.com/10.1118/1.2164070.

[167] Leighton TG (2007) What is ultrasound? Progress in Biophysics and Molecular Biology 93(1-3):3–83. ISSN 00796107. doi:10.1016/j.pbiomolbio.2006.07.026. URL https://linkinghub.elsevier.com/retrieve/pii/S0079610706000812.

[168] Wang F, Jin P, Feng Y, Fu J, Wang P, Liu X, Zhang Y, Ma Y, Yang Y, Yang A, and Feng X (2021) Flexible Doppler ultrasound device for the monitoring of blood flow velocity. Science Advances 7(44):eabi9283. ISSN 2375-2548. doi:10.1126/sciadv.abi9283. URL https://www.science.org/doi/10.1126/sciadv.abi9283.

[169] Kreitz S, Dohmen G, Hasken S, Schmitz-Rode T, Mela P, and Jockenhoevel S (2011) Nondestructive Method to Evaluate the Collagen Content of Fibrin-Based Tissue Engineered Structures Via Ultrasound. Tissue Engineering Part C: Methods 17(10):1021–1026. ISSN 1937-3384, 1937-3392. doi:10.1089/ten.tec.2010.0669. URL https://www.liebertpub.com/doi/10.1089/ten.tec.2010.0669.

[170] Rice M, Waters K, and Anseth K (2009) Ultrasound monitoring of cartilaginous matrix evolution in degradable PEG hydrogels. Acta Biomaterialia 5(1):152–161. ISSN 17427061. doi:10.1016/j.actbio.2008.07.036. URL https://linkinghub.elsevier.com/retrieve/pii/S1742706108002341.

[171] Mercado KP, Helguera M, Hocking DC, and Dalecki D (2014) Estimating Cell Concentration in Three-Dimensional Engineered Tissues Using High Frequency Quantitative Ultrasound. Annals of Biomedical Engineering 42(6):1292–1304. ISSN 0090-6964, 1573-9686. doi:10.1007/s10439-014-0994-8. URL http://link.springer.com/10.1007/s10439-014-0994-8.

[172] Deng CX, Hong X, and Stegemann JP (2016) Ultrasound Imaging Techniques for Spatiotemporal Characterization of Composition, Microstructure, and Mechanical Properties in Tissue Engineering. Tissue Engineering Part B: Reviews 22(4):311–321. ISSN 1937-3368, 1937-3376. doi:10.1089/ten.teb.2015.0453. URL https://www.liebertpub.com/doi/10.1089/ten.teb.2015.0453.

[173] Hysi E, Moore MJ, Strohm EM, and Kolios MC (2021) A tutorial in photoacoustic microscopy and tomography signal processing methods. Journal of Applied Physics 129(14):141102. ISSN 0021-8979, 1089-7550. doi:10.1063/5.0040783. URL https://pubs.aip.org/aip/jap/article/157380.

[174] Ma C, Li W, Li D, Chen M, Wang M, Jiang L, Mille LS, Garciamendez CE, Zhao Z, Zhou Q, Zhang YS, and Yao J (2022) Photoacoustic imaging of 3D-printed vascular networks. Biofabrication

14(2):025001. ISSN 1758-5082, 1758-5090. doi:10.1088/1758-5090/ac49d5. URL https://iopscience.iop.org/article/10.1088/1758-5090/ac49d5.

[175] Khramtsov VV (2018) *In Vivo* Molecular Electron Paramagnetic Resonance-Based Spectroscopy and Imaging of Tumor Microenvironment and Redox Using Functional Paramagnetic Probes. Antioxidants & Redox Signaling 28(15):1365–1377. ISSN 1523-0864, 1557-7716. doi:10.1089/ars.2017.7329. URL http://www.liebertpub.com/doi/10.1089/ars.2017.7329.

[176] Bobko AA, Eubank TD, Driesschaert B, Dhimitruka I, Evans J, Mohammad R, Tchekneva EE, Dikov MM, and Khramtsov VV (2017) Interstitial Inorganic Phosphate as a Tumor Microenvironment Marker for Tumor Progression. Scientific Reports 7(1):41233. ISSN 2045-2322. doi:10.1038/srep41233. URL https://www.nature.com/articles/srep41233.

[177] Roshchupkina GI, Bobko AA, Bratasz A, Reznikov VA, Kuppusamy P, and Khramtsov VV (2008) In vivo EPR measurement of glutathione in tumor-bearing mice using improved disulfide biradical probe. Free Radical Biology and Medicine 45(3):312–320. ISSN 08915849. doi:10.1016/j.freeradbiomed.2008.04.019. URL https://linkinghub.elsevier.com/retrieve/pii/S0891584908002177.

[178] Dhimitruka I, Bobko AA, Eubank TD, Komarov DA, and Khramtsov VV (2013) Phosphonated Trityl Probes for Concurrent in Vivo Tissue Oxygen and pH Monitoring Using Electron Paramagnetic Resonance-Based Techniques. Journal of the American Chemical Society 135(15):5904–5910. ISSN 0002-7863, 1520-5126. doi:10.1021/ja401572r. URL https://pubs.acs.org/doi/10.1021/ja401572r.

[179] Bobko AA, Eubank TD, Voorhees JL, Efimova OV, Kirilyuk IA, Petryakov S, Trofimiov DG, Marsh CB, Zweier JL, Grigor'ev IA, Samouilov A, and Khramtsov VV (2012) In vivo monitoring of pH, redox status, and glutathione using L-band EPR for assessment of therapeutic effectiveness in solid tumors: In Vivo Assessment of Therapeutic Effectiveness in Tumors. Magnetic Resonance in Medicine 67(6):1827–1836. ISSN 07403194. doi:10.1002/mrm.23196. URL https://onlinelibrary.wiley.com/doi/10.1002/mrm.23196.

[180] Poncelet M and Driesschaert B (2020) A 13 C-Labeled Triarylmethyl Radical as an EPR Spin Probe Highly Sensitive to Molecular Tumbling. Angewandte Chemie ange.202006591. ISSN

0044-8249, 1521-3757. doi:10.1002/ange.202006591. URL https://onlinelibrary.wiley.com/doi/10.1002/ange.202006591.

[181] Milutinovic M, Nakarada D, Bozunovic J, Todorovic M, Gasic U, Zivkovic S, Skoric M, Ivkovic D, Savic J, Devrnja N, Anicic N, Banjanac T, Mojovic M, and Misic D (2023) Solanum dulcamara L. Berries: A Convenient Model System to Study Redox Processes in Relation to Fruit Ripening. Antioxidants 12(2):346. ISSN 2076-3921. doi:10.3390/antiox12020346. URL https://www.mdpi.com/2076-3921/12/2/346.

[182] Liu W, Tao O, Chen L, Ling Y, Zeng M, Jin H, and Jiang D (2022) Synthesis and characterization of a Cu 2 coordination-containing TAM radical as a nitroxyl probe. RSC Advances 12(25):15980–15985. ISSN 2046-2069. doi:10.1039/D1RA07511J. URL http://xlink.rsc.org/?DOI=D1RA07511J.

[183] Nikolantonaki M, Coelho C, Noret L, Zerbib M, Vileno B, Champion D, and Gougeon RD (2019) Measurement of white wines resistance against oxidation by Electron Paramagnetic Resonance spectroscopy. Food Chemistry 270:156–161. ISSN 03088146. doi:10.1016/j.foodchem.2018.07.052. URL https://linkinghub.elsevier.com/retrieve/pii/S0308814618311919.

[184] Gulaboski R, Mirčeski V, Kappl R, Hoth M, and Bozem M (2019) Review—Quantification of Hydrogen Peroxide by Electrochemical Methods and Electron Spin Resonance Spectroscopy. Journal of The Electrochemical Society 166(8):G82–G101. ISSN 0013-4651, 1945-7111. doi:10.1149/2.1061908jes. URL https://iopscience.iop.org/article/10.1149/2.1061908jes.

[185] Zhu D, Peng X, Wagner P, and Xiao P (2022) Symmetrically-substituted Thiazolo[5,4-d]thiazole derivatives as both photoinitiators and dyes for 3D printing under violet LED. Dyes and Pigments 206:110638. ISSN 01437208. doi:10.1016/j.dyepig.2022.110638. URL https://linkinghub.elsevier.com/retrieve/pii/S0143720822005605.

[186] Ma Y, Wei W, Gong L, Li C, Hong Y, Wang X, Liang R, Shao Q, Liang Q, Huang W, Shipston MJ, and Ouyang H (2022) Biomacromolecule-based agent for high-precision light-based 3D hydrogel bioprinting. Cell Reports Physical Science 3(8):100985. ISSN 26663864. doi:10.1016/j.xcrp.2022.100985. URL https://linkinghub.elsevier.com/retrieve/pii/S2666386422002715.

[187] Dutta SD, Ganguly K, Hexiu J, Randhawa A, Moniruzzaman M, and Lim K (2023) A 3D Bioprinted Nanoengineered Hydrogel with Photoactivated Drug Delivery for Tumor Apoptosis and Simultaneous Bone Regeneration via Macrophage Immunomodulation. Macromolecular Bioscience 2300096. ISSN 1616-5187, 1616-5195. doi:10.1002/mabi.202300096. URL https://onlinelibrary.wiley.com/doi/10.1002/mabi.202300096.

[188] Kempe S, Metz H, Bastrop M, Hvilsom A, Contri RV, and Mäder K (2008) Characterization of thermosensitive chitosan-based hydrogels by rheology and electron paramagnetic resonance spectroscopy. European Journal of Pharmaceutics and Biopharmaceutics 68(1):26–33. ISSN 09396411. doi:10.1016/j.ejpb.2007.05.020. URL https://linkinghub.elsevier.com/retrieve/pii/S0939641107002494.

[189] Gluth TD, Poncelet M, Gencheva M, Hoblitzell EH, Khramtsov VV, Eubank TD, and Driesschaert B (2022) Biocompatible Monophosphonated Trityl Spin Probe, HOPE71, for In Vivo Measurement of pO$_2$, pH, and [P$_i$] by Electron Paramagnetic Resonance Spectroscopy. Analytical Chemistry acs.analchem.2c03476. ISSN 0003-2700, 1520-6882. doi:10.1021/acs.analchem.2c03476. URL https://pubs.acs.org/doi/abs/10.1021/acs.analchem.2c03476.

[190] Weil JA and Bolton JR (2007) *Electron paramagnetic resonance: elementary theory and practical applications*. Wiley-Interscience, Hoboken, N.J, 2nd ed edition. ISBN 978-0-471-75496-1. OCLC: ocm69020990.

[191] Poole CP (1996) *Electron spin resonance: a comprehensive treatise on experimental techniques*. Dover Publications, Mineola, N.Y. ISBN 978-0-486-69444-3.

[192] Eaton GR, Eaton SS, Barr DP, and Weber RT (2010) Basics of Continuous Wave EPR. In *Quantitative EPR*, 1–14. Springer Vienna, Vienna. ISBN 978-3-211-92947-6 978-3-211-92948-3. doi:10.1007/978-3-211-92948-3_1. URL http://link.springer.com/10.1007/978-3-211-92948-3_1.

[193] Yu Z, Tseytlin M, Eaton SS, and Eaton GR (2015) Multiharmonic electron paramagnetic resonance for extended samples with both narrow and broad lines. Journal of Magnetic Resonance 254:86–92. ISSN 10907807. doi:10.1016/j.jmr.2015.03.006. URL https://linkinghub.elsevier.com/retrieve/pii/S1090780715000683.

[194] Gonet M, Baranowski M, Czechowski T, Kucinska M, Plewinski A, Szczepanik P, Jurga S, and Murias M (2020) Multiharmonic electron paramagnetic resonance imaging as an innovative approach for in vivo studies. Free Radical Biology and Medicine 152:271–279. ISSN 08915849. doi: 10.1016/j.freeradbiomed.2020.03.018. URL https://linkinghub.elsevier.com/retrieve/pii/S0891584920301155.

[195] Eaton SS, Shi Y, Woodcock L, Buchanan LA, McPeak J, Quine RW, Rinard GA, Epel B, Halpern HJ, and Eaton GR (2017) Rapid-scan EPR imaging. Journal of Magnetic Resonance 280:140–148. ISSN 10907807. doi:10.1016/j.jmr.2017.02.013. URL https://linkinghub.elsevier.com/retrieve/pii/S1090780717300538.

[196] Tseytlin M (2020) Continuous-wave rapid scan EPR. In Chechik V, Murphy DM, and Bode BE, editors, *Electron Paramagnetic Resonance*, volume 27, 188–213. Royal Society of Chemistry, Cambridge. ISBN 978-1-83916-171-1. doi:10.1039/9781839162534-00188. URL http://ebook.rsc.org/?DOI=10.1039/9781839162534-00188.

[197] Joshi JP, Ballard JR, Rinard GA, Quine RW, Eaton SS, and Eaton GR (2005) Rapid-scan EPR with triangular scans and fourier deconvolution to recover the slow-scan spectrum. Journal of Magnetic Resonance 175(1):44–51. ISSN 10907807. doi:10.1016/j.jmr.2005.03.013. URL https://linkinghub.elsevier.com/retrieve/pii/S1090780705001102.

[198] Hirata H and Swartz HM (2022) RF/Microwave Resonators for Preclinical and Clinical EPR Applications: Current Status and Challenges. Applied Magnetic Resonance 53(1):167–191. ISSN 0937-9347, 1613-7507. doi:10.1007/s00723-021-01413-3. URL https://link.springer.com/10.1007/s00723-021-01413-3.

[199] Walczak T, Leśniewski P, Salikhov I, Sucheta A, Szybiński K, and Swartz HM (2005) L-band electron paramagnetic resonance spectrometer for use *in vivo* and in studies of aqueous biological samples. Review of Scientific Instruments 76(1):013107. ISSN 0034-6748, 1089-7623. doi: 10.1063/1.1832271. URL https://pubs.aip.org/aip/rsi/article/895594.

[200] Li H, Hou H, Sucheta A, Williams BB, Lariviere JP, Khan MN, Lesniewski PN, Gallez B, and Swartz HM (2010) Implantable Resonators - A Technique for Repeated Measurement of Oxygen

at Multiple Deep Sites with In Vivo EPR. In Takahashi E and Bruley DF, editors, *Oxygen Transport to Tissue XXXI*, volume 662, 265–272. Springer US, Boston, MA. ISBN 978-1-4419-1239-8 978-1-4419-1241-1. doi:10.1007/978-1-4419-1241-1_38. URL http://link.springer.com/10.1007/978-1-4419-1241-1_38. Series Title: Advances in Experimental Medicine and Biology.

[201] Elas M, Ahn KH, Parasca A, Barth ED, Lee D, Haney C, and Halpern HJ (2006) Electron Paramagnetic Resonance Oxygen Images Correlate Spatially and Quantitatively with Oxylite Oxygen Measurements. Clinical Cancer Research 12(14):4209–4217. ISSN 1078-0432, 1557-3265. doi:10.1158/1078-0432.CCR-05-0446. URL https://aacrjournals.org/clincancerres/article/12/14/4209/284777/Electron-Paramagnetic-Resonance-Oxygen-Images.

[202] Tseytlin O, Guggilapu P, Bobko AA, AlAhmad H, Xu X, Epel B, O'Connell R, Hoblitzell EH, Eubank TD, Khramtsov VV, Driesschaert B, Kazkaz E, and Tseytlin M (2019) Modular imaging system: Rapid scan EPR at 800 MHz. Journal of Magnetic Resonance 305:94–103. ISSN 10907807. doi:10.1016/j.jmr.2019.06.003. URL https://linkinghub.elsevier.com/retrieve/pii/S1090780719301107.

[203] Caryotakis G (1998) The klystron: A microwave source of surprising range and endurance. Physics of Plasmas 5(5):1590–1598. ISSN 1070-664X, 1089-7674. doi:10.1063/1.872826. URL https://pubs.aip.org/aip/pop/article/5/5/1590-1598/1069126.

[204] Oles T, Strangeway RA, Luglio J, Froncisz W, and Hyde JS (1992) *X* -band low phase noise Gunn diode oscillator for EPR spectroscopy. Review of Scientific Instruments 63(8):4010–4011. ISSN 0034-6748, 1089-7623. doi:10.1063/1.1143257. URL https://pubs.aip.org/aip/rsi/article/63/8/4010-4011/328112.

[205] Eaton G and Eaton S (1988) EPR Imaging: Progress and Prospects. Bulletin of Magnetic Resonance 10:22–31.

[206] Herman GT (2009) *Fundamentals of Computerized Tomography*. Advances in Pattern Recognition. Springer London, London. ISBN 978-1-85233-617-2 978-1-84628-723-7. doi:10.1007/978-1-84628-723-7. URL http://link.springer.com/10.1007/978-1-84628-723-7.

[207] Breit G (1928) The Magnetic Moment of the Electron. Nature 122(3078):649–649. ISSN 0028-0836, 1476-4687. doi:10.1038/122649a0. URL https://www.nature.com/articles/122649a0.

[208] Schweiger A and Jeschke G (2001) *Principles of pulse electron paramagnetic resonance*. Oxford University Press, Oxford, UK ; New York. ISBN 978-0-19-850634-8.

[209] Dikanov SA and Tsvetkov YD (1992) *Electron spin echo envelope modulation (ESEEM) spectroscopy*. CRC Press, Boca Raton. ISBN 978-0-8493-4224-0.

[210] Jeschke G and Schweiger A (1995) Time-domain chirp electron nuclear double resonance spectroscopy in one and two dimensions. The Journal of Chemical Physics 103(19):8329–8337. ISSN 0021-9606, 1089-7690. doi:10.1063/1.470145. URL https://pubs.aip.org/jcp/article/103/19/8329/179911/Time-domain-chirp-electron-nuclear-double.

[211] Epel B, Kotecha M, and Halpern HJ (2017) In vivo preclinical cancer and tissue engineering applications of absolute oxygen imaging using pulse EPR. Journal of Magnetic Resonance 280:149–157. ISSN 10907807. doi:10.1016/j.jmr.2017.04.017. URL https://linkinghub.elsevier.com/retrieve/pii/S1090780717301179.

[212] Biskup T (2019) Structure–Function Relationship of Organic Semiconductors: Detailed Insights From Time-Resolved EPR Spectroscopy. Frontiers in Chemistry 7:10. ISSN 2296-2646. doi:10.3389/fchem.2019.00010. URL https://www.frontiersin.org/article/10.3389/fchem.2019.00010/full.

[213] Van Der Est A and Poddutoori PK (2017) Time-Resolved EPR in Artificial Photosynthesis. In Hou HJ, Najafpour MM, Moore GF, and Allakhverdiev SI, editors, *Photosynthesis: Structures, Mechanisms, and Applications*, 359–387. Springer International Publishing, Cham. ISBN 978-3-319-48871-4 978-3-319-48873-8. doi:10.1007/978-3-319-48873-8_18. URL http://link.springer.com/10.1007/978-3-319-48873-8_18.

[214] Spitzbarth M, Wessig M, Lemke T, Schachtschneider A, Polarz S, and Drescher M (2015) Simultaneous Monitoring of Macroscopic and Microscopic Diffusion of Guest Molecules in Silica and

Organosilica Aerogels by Spatially and Time-Resolved Electron Paramagnetic Resonance Spectroscopy. The Journal of Physical Chemistry C 119(30):17474–17479. ISSN 1932-7447, 1932-7455. doi:10.1021/acs.jpcc.5b05743. URL https://pubs.acs.org/doi/10.1021/acs.jpcc.5b05743.

[215] Mignion L, Desmet CM, Harkemanne E, Tromme I, Joudiou N, Wehbi M, Baurain JF, and Gallez B (2022) Noninvasive detection of the endogenous free radical melanin in human skin melanomas using electron paramagnetic resonance (EPR). Free Radical Biology and Medicine 190:226–233. ISSN 08915849. doi:10.1016/j.freeradbiomed.2022.08.020. URL https://linkinghub.elsevier.com/retrieve/pii/S0891584922005482.

[216] Fattibene P and Callens F (2010) EPR dosimetry with tooth enamel: A review. Applied Radiation and Isotopes 68(11):2033–2116. ISSN 09698043. doi:10.1016/j.apradiso.2010.05.016. URL https://linkinghub.elsevier.com/retrieve/pii/S0969804310002356.

[217] Momo F, Colacicchi S, and Sotgiu A (1993) Limits of deconvolution in enhancing the resolution in EPR imaging experiments. Measurement Science and Technology 4(1):60–64. ISSN 0957-0233, 1361-6501. doi:10.1088/0957-0233/4/1/010. URL https://iopscience.iop.org/article/10.1088/0957-0233/4/1/010.

[218] Gallez B (2021) Oxygenation Status in Normal Tissues, Pathological Tissues and Malignant Tumors: A pO2 Database Based on Electron Paramagnetic Resonance (EPR) Oximetry Measurements. Applied Magnetic Resonance 52(10):1395–1450. ISSN 0937-9347, 1613-7507. doi:10.1007/s00723-021-01358-7. URL https://link.springer.com/10.1007/s00723-021-01358-7.

[219] Poncelet M, Ngendahimana T, Gluth TD, Hoblitzell EH, Eubank TD, Eaton GR, Eaton SS, and Driesschaert B (2022) Synthesis and characterization of a biocompatible $^{13}$C$_1$ isotopologue of trityl radical OX071 for *in vivo* EPR viscometry. The Analyst 147(24):5643–5648. ISSN 0003-2654, 1364-5528. doi:10.1039/D2AN01527G. URL http://xlink.rsc.org/?DOI=D2AN01527G.

[220] Sanzhaeva U, Poncelet M, Tseytlin O, Tseytlin M, Gencheva M, Eubank TD, Khramtsov VV, and Driesschaert B (2020) Synthesis, Characterization, and Application of a Highly Hydrophilic

Triarylmethyl Radical for Biomedical EPR. The Journal of Organic Chemistry 85(16):10388–10398. ISSN 0022-3263, 1520-6904. doi:10.1021/acs.joc.0c00557. URL https://pubs.acs.org/doi/10.1021/acs.joc.0c00557.

[221] Velayutham M, Poncelet M, Eubank TD, Driesschaert B, and Khramtsov VV (2021) Biological Applications of Electron Paramagnetic Resonance Viscometry Using a 13C-Labeled Trityl Spin Probe. Molecules 26(9):2781. ISSN 1420-3049. doi:10.3390/molecules26092781. URL https://www.mdpi.com/1420-3049/26/9/2781.

[222] Song Y, Liu Y, Liu W, Villamena FA, and Zweier JL (2014) Characterization of the binding of the Finland trityl radical with bovine serum albumin. RSC Adv. 4(88):47649–47656. ISSN 2046-2069. doi:10.1039/C4RA04616A. URL http://xlink.rsc.org/?DOI=C4RA04616A.

[223] Poncelet M, Huffman JL, Khramtsov VV, Dhimitruka I, and Driesschaert B (2019) Synthesis of hydroxyethyl tetrathiatriarylmethyl radicals OX063 and OX071. RSC Advances 9(60):35073–35076. ISSN 2046-2069. doi:10.1039/C9RA08633A. URL http://xlink.rsc.org/?DOI=C9RA08633A.

[224] Pandian RP, Parinandi NL, Ilangovan G, Zweier JL, and Kuppusamy P (2003) Novel particulate spin probe for targeted determination of oxygen in cells and tissues. Free Radical Biology and Medicine 35(9):1138–1148. ISSN 08915849. doi:10.1016/S0891-5849(03)00496-9. URL https://linkinghub.elsevier.com/retrieve/pii/S0891584903004969.

[225] Liu KJ, Gast P, Moussavi M, Norby SW, Vahidi N, Walczak T, Wu M, and Swartz HM (1993) Lithium phthalocyanine: a probe for electron paramagnetic resonance oximetry in viable biological systems. Proceedings of the National Academy of Sciences 90(12):5438–5442. ISSN 0027-8424, 1091-6490. doi:10.1073/pnas.90.12.5438. URL https://pnas.org/doi/full/10.1073/pnas.90.12.5438.

[226] Bienia A, Kowolik E, Drzał A, Jasiński K, Bobko A, and Elas M (2023) Non-invasive oximetric imaging of orthotopic glioblastoma tumors in mice. Photodiagnosis and Photodynamic Therapy 41:103469. ISSN 15721000. doi:10.1016/j.pdpdt.2023.103469. URL https://linkinghub.elsevier.com/retrieve/pii/S1572100023001965.

[227] Blakely AM, Manning KL, Tripathi A, and Morgan JR (2015) Bio-Pick, Place, and Perfuse: A New Instrument for Three-Dimensional Tissue Engineering. Tissue Engineering Part C: Methods 21(7):737–746. ISSN 1937-3384, 1937-3392. doi:10.1089/ten.tec.2014.0439. URL https://www.liebertpub.com/doi/10.1089/ten.tec.2014.0439.

[228] Aguilar IN, Smith LJ, Olivos DJ, Chu TMG, Kacena MA, and Wagner DR (2019) Scaffold-free bioprinting of mesenchymal stem cells with the regenova printer: Optimization of printing parameters. Bioprinting 15:e00048. ISSN 24058866. doi:10.1016/j.bprint.2019.e00048. URL https://linkinghub.elsevier.com/retrieve/pii/S240588661830040X.

[229] Kuphaldt T (2015) *Lessons in Electric Circuits*. URL freetechbooks.com/lessons-in-electric-circuits-t258.html.

[230] Zhang Y, Zhang A, He C, Hou C, and Xu J (2021) Design of a Fast-Sweeping C-X band Frequency Source Based on DDS and Frequency Multiplying Chain. In *2021 International Conference on Microwave and Millimeter Wave Technology (ICMMT)*, 1–3. IEEE, Nanjing, China. ISBN 978-1-66543-437-9. doi:10.1109/ICMMT52847.2021.9618243. URL https://ieeexplore.ieee.org/document/9618243/.

[231] Liu Y, Ye R, Hu Q, Chen B, Zhang W, Zhou F, and Xu N (2023) A DDS-based integrated microwave source for fast frequency sweeping in quantum magnetic-resonance systems. AIP Advances 13(1):015009. ISSN 2158-3226. doi:10.1063/5.0131395. URL https://pubs.aip.org/aip/adv/article/2871074.

[232] (2018). Peregrine Semiconductor PE64904 Datasheet. URL https://www.psemi.com/pdf/datasheets/pe64904ds.pdf.

[233] Habib S (2013) MTCA Fast Digitizer for Direct RF Measurements. In *Proceedings of the 20th International Conference Mixed Design of Intergrated Circuits and Systems - MIXDES 2013*, 118–122. IEEE.

[234] Dargie W and Poellabauer C (2010) *Fundamentals of wireless sensor networks: theory and practice*. Wiley series on wireless communications and mobile computing. Wiley, Chichester, West Sussex, U.K. ; Hoboken, NJ. ISBN 978-0-470-99765-9.

[235] Khan FH, Pasha MA, and Masud S (2021) Advancements in Microprocessor Architecture for Ubiquitous AI—An Overview on History, Evolution, and Upcoming Challenges in AI Implementation. Micromachines 12(6):665. ISSN 2072-666X. doi:10.3390/mi12060665. URL https://www.mdpi.com/2072-666X/12/6/665.

[236] McRoberts M (2010) *Beginning Arduino*. Technology in action. Apress ; Distributed to the book trade worldwide by Springer Science + Business Media, LLC, New York : New York. ISBN 978-1-4302-3240-7. OCLC: ocn630509635.

[237] Serrano J (2008) Introduction to FPGA design doi:10.5170/CERN-2008-003.231. URL http://cds.cern.ch/record/1100537. Publisher: CERN.

[238] Einspruch NG and Hilbert JL, editors (1991) *Application specific integrated circuit (ASIC) technology: edited by Norman G. Einspruch, Jeffrey L. Hilbert*. Number v. 23 in VLSI electronics. Academic Press, San Diego. ISBN 978-0-12-234123-6.

[239] Dawoud D and Dawoud P (2022) *Serial communication protocols and standards*. CRC Press. ISBN 978-87-7022-154-2.

[240] Tumbleston JR, Shirvanyants D, Ermoshkin N, Janusziewicz R, Johnson AR, Kelly D, Chen K, Pinschmidt R, Rolland JP, Ermoshkin A, Samulski ET, and DeSimone JM (2015) Continuous liquid interface production of 3D objects. Science 347(6228):1349–1352. ISSN 0036-8075, 1095-9203. doi:10.1126/science.aaa2397. URL https://www.science.org/doi/10.1126/science.aaa2397.

[241] Cao X, Yu S, Cui H, and Li Z (2022) 3D Printing Devices and Reinforcing Techniques for Extruded Cement-Based Materials: A Review. Buildings 12(4):453. ISSN 2075-5309. doi:10.3390/buildings12040453. URL https://www.mdpi.com/2075-5309/12/4/453.

[242] Lange J, Feucht T, and Erven M (2020) 3D printing with steel: Additive Manufacturing for connections and structures. Steel Construction 13(3):144–153. ISSN 1867-0520, 1867-0539. doi:10.1002/stco.202000031. URL https://onlinelibrary.wiley.com/doi/10.1002/stco.202000031.

[243] Horvath J and Cameron R (2020) *Mastering 3D Printing: A Guide to Modeling, Printing, and*

*Prototyping*. Apress, Berkeley, CA. ISBN 978-1-4842-5841-5 978-1-4842-5842-2. doi:10.1007/978-1-4842-5842-2. URL http://link.springer.com/10.1007/978-1-4842-5842-2.

[244] Zhang J and Xiao P (2018) 3D printing of photopolymers. Polymer Chemistry 9(13):1530–1540. ISSN 1759-9954, 1759-9962. doi:10.1039/C8PY00157J. URL http://xlink.rsc.org/?DOI=C8PY00157J.

[245] (2022). Best 3D printer filament: the main types in 2023. URL https://all3dp.com/1/3d-printer-filament-types-3d-printing-3d-filament/.

[246] Shahrubudin N, Lee T, and Ramlan R (2019) An Overview on 3D Printing Technology: Technological, Materials, and Applications. Procedia Manufacturing 35:1286–1296. ISSN 23519789. doi:10.1016/j.promfg.2019.06.089. URL https://linkinghub.elsevier.com/retrieve/pii/S2351978919308169.

[247] Gülcan O, Günaydın K, and Tamer A (2021) The State of the Art of Material Jetting—A Critical Review. Polymers 13(16):2829. ISSN 2073-4360. doi:10.3390/polym13162829. URL https://www.mdpi.com/2073-4360/13/16/2829.

[248] Sun S, Brandt M, and Easton M (2017) Powder bed fusion processes. In *Laser Additive Manufacturing*, 55–77. Elsevier. ISBN 978-0-08-100433-3. doi:10.1016/B978-0-08-100433-3.00002-6. URL https://linkinghub.elsevier.com/retrieve/pii/B9780081004333000026.

[249] Gibson I, Rosen D, Stucker B, and Khorasani M (2021) *Additive manufacturing technologies*. Springer, Cham, Switzerland, third edition edition. ISBN 978-3-030-56127-7 978-3-030-56129-1 978-3-030-56126-0.

[250] Rosenberger C and Eckardt KU (2019) Oxygenation of the Transplanted Kidney. Seminars in Nephrology 39(6):554–566. ISSN 02709295. doi:10.1016/j.semnephrol.2019.10.005. URL https://linkinghub.elsevier.com/retrieve/pii/S0270929519300907.

[251] Corbett DC, Olszewski E, and Stevens K (2019) A FRESH Take on Resolution in 3D Bioprinting. Trends in Biotechnology 37(11):1153–1155. ISSN 01677799. doi:10.1016/j.tibtech.2019.09.003. URL https://linkinghub.elsevier.com/retrieve/pii/S016777991930229X.

[252] Ugolini GS, Visone R, Redaelli A, Moretti M, and Rasponi M (2017) Generating Multicompart-
mental 3D Biological Constructs Interfaced through Sequential Injections in Microfluidic Devices.
Advanced Healthcare Materials 6(10):1601170. ISSN 21922640. doi:10.1002/adhm.201601170.
URL https://onlinelibrary.wiley.com/doi/10.1002/adhm.201601170.

[253] Miri AK, Mirzaee I, Hassan S, Mesbah Oskui S, Nieto D, Khademhosseini A, and Zhang YS
(2019) Effective bioprinting resolution in tissue model fabrication. Lab on a Chip 19(11):2019–
2037. ISSN 1473-0197, 1473-0189. doi:10.1039/C8LC01037D. URL http://xlink.rsc.org/
?DOI=C8LC01037D.

[254] Al-Hallaq HA, Fan X, Zamora M, River JN, Moulder JE, and Karczmar GS (2002) Spectrally
inhomogeneous BOLD contrast changes detected in rodent tumors with high spectral and spatial
resolution MRI. NMR in Biomedicine 15(1):28–36. ISSN 0952-3480, 1099-1492. doi:10.1002/
nbm.728. URL https://onlinelibrary.wiley.com/doi/10.1002/nbm.728.

[255] Han S, Paulsen JL, Zhu G, Song Y, Chun S, Cho G, Ackerstaff E, Koutcher JA, and Cho H (2012)
Temporal/spatial resolution improvement of in vivo DCE-MRI with compressed sensing-optimized
FLASH. Magnetic Resonance Imaging 30(6):741–752. ISSN 0730725X. doi:10.1016/j.mri.2012.
02.001. URL https://linkinghub.elsevier.com/retrieve/pii/S0730725X12000355.

[256] Moses WW (2011) Fundamental limits of spatial resolution in PET. Nuclear Instruments and
Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated
Equipment 648:S236–S240. ISSN 01689002. doi:10.1016/j.nima.2010.11.092. URL https:
//linkinghub.elsevier.com/retrieve/pii/S0168900210026276.

[257] Beekman FJ and Vastenhouw B (2004) Design and simulation of a high-resolution stationary
SPECT system for small animals. Physics in Medicine and Biology 49(19):4579–4592. ISSN
0031-9155, 1361-6560. doi:10.1088/0031-9155/49/19/009. URL https://iopscience.iop.
org/article/10.1088/0031-9155/49/19/009.

[258] Christensen-Jeffries K, Couture O, Dayton PA, Eldar YC, Hynynen K, Kiessling F, O'Reilly M,
Pinton GF, Schmitz G, Tang MX, Tanter M, and Van Sloun RJ (2020) Super-resolution Ul-
trasound Imaging. Ultrasound in Medicine & Biology 46(4):865–891. ISSN 03015629. doi:

10.1016/j.ultrasmedbio.2019.11.013. URL https://linkinghub.elsevier.com/retrieve/pii/S0301562919315959.

[259] USB: Port Types and Speeds Compared. URL https://tripplite.eaton.com/products/usb-connectivity-types-standards.

[260] (2014). 400Gb/s Ethernet study group. URL www.ieee802.org/3/400GSG/.

[261] Rhodes S (2023). 3D Printer Resolution: How detailed can a 3D print be? URL https://all3dp.com/2/3d-printer-resolution/.

[262] Tee YL, Tran P, Leary M, Pille P, and Brandt M (2020) 3D Printing of polymer composites with material jetting: Mechanical and fractographic analysis. Additive Manufacturing 36:101558. ISSN 22148604. doi:10.1016/j.addma.2020.101558. URL https://linkinghub.elsevier.com/retrieve/pii/S2214860420309301.

[263] Chua CK, Chou SM, and Wong TS (1998) A study of the state-of-the-art rapid prototyping technologies. The International Journal of Advanced Manufacturing Technology 14(2):146–152. ISSN 0268-3768, 1433-3015. doi:10.1007/BF01322222. URL http://link.springer.com/10.1007/BF01322222.

[264] Mao M, He J, Li X, Zhang B, Lei Q, Liu Y, and Li D (2017) The Emerging Frontiers and Applications of High-Resolution 3D Printing. Micromachines 8(4):113. ISSN 2072-666X. doi:10.3390/mi8040113. URL http://www.mdpi.com/2072-666X/8/4/113.
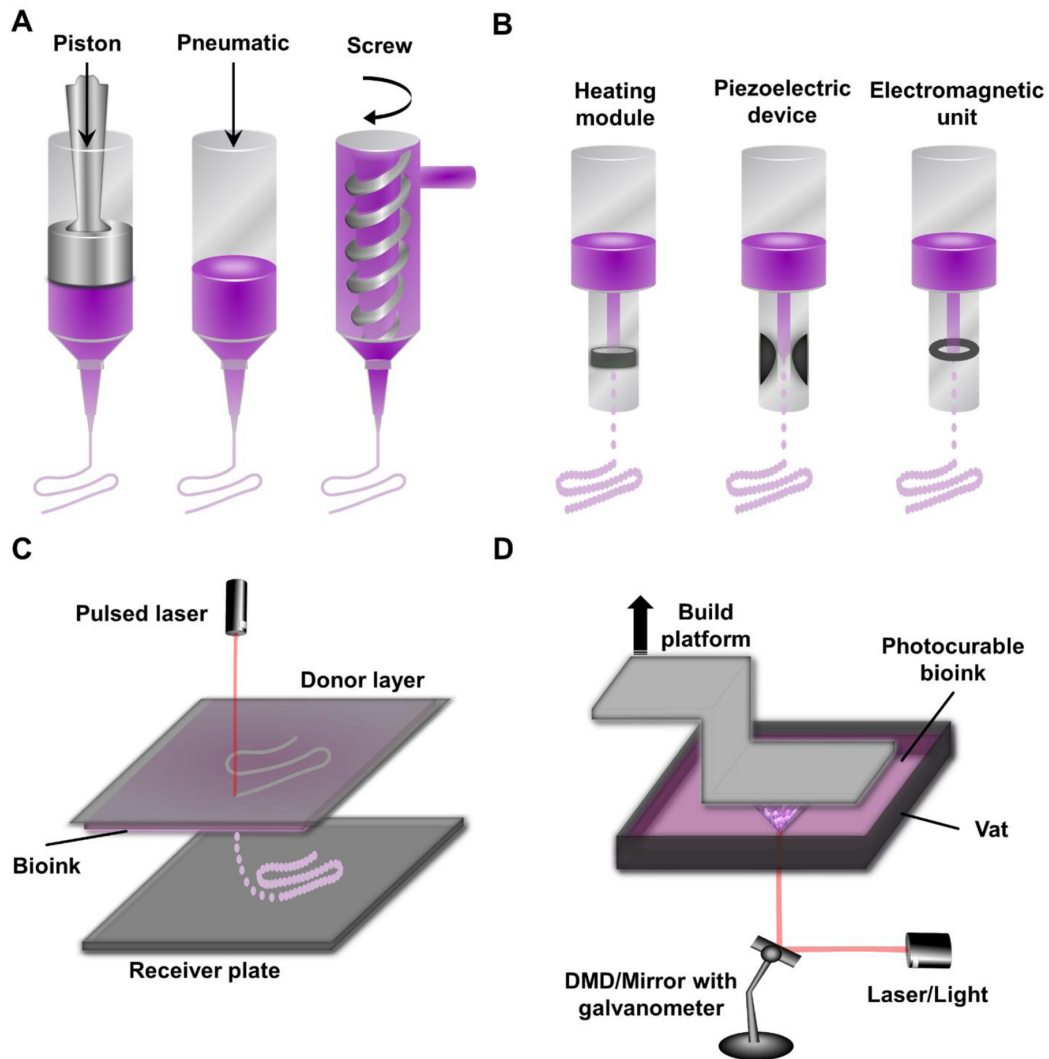
Figure 1.1: Main categories of bioprinters. A) Extrusion bioprinting. B) Inkjet bioprinting. C) LIFT bioprinting. D) Photopolymerization bioprinting. Taken from Figure 1, Ravanbakhsh H, Karamzadeh V, Bao G, Mongeau L, Juncker D, Zhang YS (2021) Emerging Technologies in Multi-Material Bioprinting. Adv Mater 33(49): e2104730.

| Method | Crosslinking strategy | Approximate Resolution |
|---|---|---|
| Inkjet[52] | light, ionic, or enzymatic | 10µm |
| Extrusion[53] | typically ionic or enzymatic | 200µm |
| Multi-nozzle extrusion | same as extrusion | same as extrusion |
| Co-axial extrusion[55] | ionic or enzymatic | core nozzle 210µm |
| Embedded extrusion[251] | ionic or enzymatic | 20 µm |
| SLA[60] | light | 1-10µm |
| DLP[60] | light | 10µm |
| Sequential injection[252] | light | 200µm feature size |
| Liquid deposition[65] | light | 15 µm |
| Multi-wavelength[66] | light | same as DLP |
| LIFT[253] | light | 100 µm |

Table 1.1: Summary of selected bioprinting technologies. Bioprinting resolution generally different in the z axis compared to x and y axes. Published resolution is generally the x,y resolution. Minimum possible feature size is usually several times larger than resolution. Resolution depends on bioink and printer components, and usually has an inverse relationship with printing speed.

| Method | What is measured? | Depth | Resolution | Notes |
|---|---|---|---|---|
| Biosensors | molecules, reaction rate | single point | single point | inserting probe usually invasive |
| Microscopy | molecules, reaction rate, live/dead cells | 3 mm | 200 nm* | surface measurements |
| BOLD MRI | hemoglobin saturation | 1m | 1mm[254] | |
| DCE MRI | perfusion and permeability | 1m | 100 µm[255] | |
| PET | hypoxia, $[O_2]$, glucose metabolism, pH, etc | >1 m | >0.5 mm[256] | Probes are radioactive |
| SPECT | perfusion, blood flow | >1 m | 200 µm[257] | typical resolution for commercial system is 1 cm |
| Ultrasound | blood flow, ECM remodelling, live/dead cells | 30 cm | 10-100 µm[258] | penetration depth for 1MHz system |
| EPRI | $pO_2$, $[O_2]$, pH, small molecules, molecular structure, reaction rate | 10 cm | 100µm | minimally invasive, quantitative |

Table 1.2: Summary of techniques to measure the microenvironment. DCE MRI stands for dynamic contrast-enhanced MRI. * Microscopy resolution does not include superresolution techniques.

| Protocol | Max Speed | Distance | Wires? | Structured packet? | C or P | slave-master? |
|---|---|---|---|---|---|---|
| USB 3.2 [259] | 10 Gbps | 3m | 10-24 + ground | yes | both | slave-master |
| USB 2.0 [259] | 480 Mbps | 5m | 4 | yes | both | slave-master |
| SPI | ~50 Mbps | 10m | 3 + SS | not required | peripheral | slave-master |
| I2C | ~5Mbps | 3-4m | 2 | address required | peripheral | multimaster |
| WiFi | 9.6 Gbps (802.11ax) | typically ¿50m | air transmission | yes | both | one master per channel |
| Bluetooth | 2Mbps (BT 5.0) | ¡100m | air transmission | yes | both | multimaster |
| Ethernet | currently 400Gbps [260] | long-distance | 8 | yes | both | multimaster |

Table 1.3: Communications protocols summarized. C represents computer to computer communication, P represents computer to peripheral component communication, Gbps = $10^9$ bits per second, Mbps = $10^6$ bits per second

| Method | Category | Resolution* |
|---|---|---|
| Filament | Mechanical | 0.5 mm[261] |
| SLA | Light | 25 μm[261] |
| DLP | Light | 25 μm |
| Material jetting | Light | 0.5 mm[262] |
| Powder bed fusion | doesn't fit | 100μm[263] |
| Sheet lamination | Either | 50 μm[263] |

Table 1.4: Summary of selected 3D printing technologies. Resolution figures are for typical commercial models - machines purpose built for high resolution will have different figures.[264] Resolution is in the x,y plane. Z resolution is usually slightly different. All values are approximate and vary between manufacturers/models.

# Chapter 2

# Rapid scan EPR: Automated digital resonator control for low-latency data acquisition

**Authors:** Ryan C O'Connell[a,c], Oxana Tseytlin[a,c], Andrey A Bobko[a,c], Timothy D Eubank[c,d], Mark Tseytlin[a,b,c],*

**Affiliations:**

[a]Department of Biochemistry and Molecular Medicine, West Virginia University, Morgantown, WV 26506, USA

[b]West Virginia University Cancer Institute, Morgantown, WV 26506, USA

[c]In Vivo Multifunctional Magnetic Resonance Center at Robert C. Byrd Health Sciences Center, West Virginia University, Morgantown, WV 26506, USA

[d]Department of Microbiology, Immunology, and Cell Biology, West Virginia University, Morgantown, WV 26506, USA

**Note:** This chapter includes text and figures from the publication above.

## 2.1  Abstract

Automation has become an essential component of modern scientific instruments which often capture large amounts of complex dynamic data. Algorithms are developed to read multiple sensors in parallel with data acquisition and to adjust instrumental parameters on the fly. Decisions are made on a time scale unattainable to the human operator. In addition to speed, automation reduces human error, improves the reproducibility of experiments, and improves the reliability of acquired data. An automatic digital control (ADiC) was developed to reliably sustain critical coupling of a resonator over a wide range of time-varying loading conditions. The ADiC uses the computational power of a microcontroller that directly communicates with all system components independent of a personal computer (PC). The PC initiates resonator tuning and coupling by sending a command to MC via serial port. After receiving the command, ADiC establishes critical coupling conditions within approximately 5ms. A printed circuit board resonator was designed to permit digital control. The performance of the resonator together with the ADiC was evaluated by varying the resonator loading from empty to heavily loaded. For the loading, samples containing aqueous sodium chloride that strongly absorb electromagnetic waves were used. A previously reported rapid scan (RS) electron paramagnetic resonance (EPR) imaging instrument was upgraded by the incorporation of ADiC. RS spectra and an *in vivo* image of oxygen in a mouse tumor model have been acquired using the upgraded system. ADiC robustly sustained critical coupling of the resonator to the transmission line during these measurements. The design implemented in this study can be used in slow-scan and pulsed EPR with modifications.

See graphical abstract here: GA.

## 2.2  Introduction

Rapid scan (RS) electron paramagnetic resonance (EPR) has been gaining interest among researchers and developers in the field of magnetic resonance, both in academia and industry.[1–19] RS EPR promises two important deliverables: snapshot spectroscopy and enhanced sensitivity. In principle, a spectrum can be measured in nanoseconds. In practice, averaging is often needed to increase signal-to-noise (SNR) of the measured signals. EPR sensitivity enhancement, when compared to the standard slow-scan methods, is achieved through de-saturation of the electron spins system. Rapid passage through

resonance reduces the interaction time between the spins and the excitation field, $B_1$, resulting in an increased power threshold for saturation. Maximum enhancement is achieved when the rate of passage is accelerated by increasing the scan frequency (or reducing the scan period) rather than widening the scan amplitude. The latter is selected to cover the spectral width of interest. Increasing scan rates permit the use of higher excitation powers without distorting EPR spectra. In this regime, the spin system is well-approximated as a linear system, and SNR enhancement becomes proportional to the amplitude of the $B_1$ field.[1,20–22]

The use of higher powers to boost SNR may come with elevated challenges for signal detection. For example, in continuous-wave (CW) RS EPR, an incident excitation wave produces reflection that may overwhelm the EPR signal. The power of the reflected wave is proportional to that of the incident wave. Excessive levels of reflection in the RS experiment may cause EPR signal distortion and increased levels of noise and background.[23–25]

An automatic digital control (ADiC) method has been developed to protect EPR detection from elevated reflection. ADiC reliably establishes the critically coupling condition (CCC) for a wide range of coupling and tuning conditions (from empty to fully loaded resonator) within a time interval of tens of milliseconds. Within several milliseconds after establishing CCC, ADiC automatically initiates RS data acquisition. User input is not typically required; EPR measurements can be started from the moment the sample is placed in the resonator.

The robust performance of ADiC is achieved with the use of a global resonance frequency search. In every retuning event, a 20 MHz frequency sweep is generated, and the frequency of the lowest reflection is calculated. As a result, CCC is reliably established provided that the frequency drift does not exceed the sweeping range. In the case of failed tuning, which is very unlikely, ADiC changes the tuning range. In comparison with the standard automatic frequency control (AFC), the ADiC frequency search is non-continuous. After tuning/coupling is completed, ADiC switches to CW operation. As a result, the risk of EPR spectra distortion caused by simultaneous tuning and data acquisition is eliminated. This is especially important for EPR imaging, where probes with very narrow lines are used. For example, EPR oximetry is based on the conversion of linewidth into oxygen concentration or partial pressure.[26–29] Any line broadening due to the shift of the frequency would skew measured oxygen values.

The method of automatic digital tuning and coupling described here was successfully tested on a wide range of samples, including for a series of *in vivo* imaging experiments. The key control measurements

are reported in this manuscript together with a detailed description of the ADiC methods, including enabling resonator design, software, firmware, and hardware. The main components of the presented technology are the microcontroller, the digitally tuned capacitor (see Fig. 2.1), the digital frequency source, and the arbitrary waveform generator (see Figs. 2.2,2.3).

## 2.3 EPR resonator with digital components enables fast automation

CW EPR remains one of the most commonly used methods to excite and detect responses from the electron spin system. In comparison with the pulse modality, CW methods are dead-time-free.[29] However, EPR signals are measured in the presence of continuous excitation, and therefore continuous reflection. The reflected power is at its minimum at CCC when the net resonator impedance is close to 50 Ohms. Reactive elements, such as variable inductors and/or trimmer capacitors[30–33] are introduced into the resonator circuitry to permit impedance adjustability to establish CCC. Depending on the reactive elements used, their inductance and capacitance values can be changed either mechanically[33–40] or electrically.[5,41–47] The latter can be described as a semi-digital method because an impedance controlling voltage is used that is often generated by a digital circuit.

The mechanical approach is reliable but rather slow. A trimmer capacitor, coupling screw, or coupling inductance loop are used to change the resonator impedance. Despite its limitations, the mechanical approach remains a practical necessity for pulse EPR. Only manually adjustable elements are available that can withstand the high-power (kilowatts) radiofrequency pulses.

In the semi-digital approach, the use of a digital-analog converter (DAC) permits remote control of reactive elements, such as voltage-tunable capacitors (VTC) recently developed by STMicroelectronics (Switzerland). The VTCs demonstrated improved performance when used in EPR resonators.[6] The downside of using voltage-controlled capacitors is that any voltage variation directly translate into EPR noise. Low-pass filtering[41,43,48] can be used to partially suppress the noise at the expense of a prolonged (dead) time with which the capacitance can be changed. An additional complication related to the use of analog controls is that magnetic field modulation (or field scan in RS) induces a small yet detectable voltage that contributes to the background signal.[1,5,16]

A fully digital approach to resonator coupling is proposed that is both fast and noise insensitive. The enabling technology for this method is UltraCMOS Digitally Tunable Capacitors (DTC). A DTC is

a millimeter-sized chip containing several fixed capacitors with associated fast digital switches (see Fig. 2.1). This chip communicates with external devices through Serial Peripheral Interface (SPI). A total of five wires are required for communication. Two wires provide power and ground connections. Three digital lines are used for the clock, data, and chip-enable signals. The data sent to the chip defines positions ('on' or 'off') for all switches. The total number of on/off combination is $2^N$, where N is the number of fixed capacitors. For the example in Fig. 2.1, N=5 and the net capacitance between points A and B has 32 discrete values. The advantage of DTC-based tuning/coupling is that digital signals are insensitive to noise. There are only two states, low and high, defined by the corresponding voltage levels (see Table 2.1). The switching speed of capacitances depends on data transfer rates ($\sim$ megabits per second).

A printed circuit board resonator (PCBR) has been developed (see Fig. 2.2) with incorporated DTCs that constitutes a functional EPR resonator. The schematics for this PCB resonator were designed using Eagle software (Autodesk, CA, USA). The boards were manufactured by PCBWay (Shenzhen, China). The resonator was originally designed for *in vivo* EPR imaging at 800MHz. However, a wider range of frequencies (see Table 2.1) can be used. Frequency can be adjusted using a trimmer capacitor (2.5 pF - 10 pF), denoted in the figure as TC. The purpose of the fixed coupling capacitors (FCC in Fig. 2.2) is to optimize the coupling range so that the lowest DTC value permits the tuning of an empty resonator, while the highest DTC value establishes CCC when the resonator is fully loaded (see Table 2.2). For the single-loop resonator shown in Fig. 2.2, the capacitances for $FCC_1$ and $FCC_2$ were 2.4 pF and 0.3 pF, respectively. A chip balun (TCN2-122+, Mini-Circuits US) was used in the circuit that permits design miniaturization. Decoupling inductors (300nH, LQW18AN_8Z, Murata Electronics, Japan) decouple RF from the digital control circuit. DTCs PE64906 developed by Peregrine Semiconductors (San Diego, USA) were used that give a capacitance range from 0.9 to 4.6 pF. Digital control pins (DCP in Fig. 2.2) connect power, ground, clock, data, and chip-enabled wires, according to SPI protocol. Any device that outputs digital signals can be used to control DTCs such as Arduino, Teensy, Raspberry Pi, and a wide range of other available digital cards.

Specifications related to the EPR resonator shown in Fig. 2.2 are summarized in Table 2.1. The DTC chip used in the PCB resonator has a 5-bit resolution, permitting capacitance change from 0.9 pF to 4.6 pF with an increment $\approx$ of 0.15 pF. It was found experimentally that a small one-increment asymmetry between $DTC_1$ and $DTC_2$ can be used to effectively increase the total coupling resolution

to 94 combinations ($\approx$ 6.6 bits). For example (see Table 2.2), CCC of the empty resonator is achieved when $DTC_1$ index $= 1$ and $DTC_2$ index $= 2$ (the index runs from 0 to 31, a total of 32 positions). This small asymmetry does not negatively affect resonator performance.

The transition time between two steady-state reflection conditions was measured using a digital oscilloscope during DTC switching from index 10 to index 1. The reflection level was measured using Minicircuit's ZX47-40 + power detector. For the clock rate of 1 MHz, the transient time is approximately 20 µs. The DTC specifications permit 25-times faster data transfer speed.

Resonator performance related to coupling range was tested experimentally by varying the resonator loading conditions. To mimic a lossy sample, a sodium chloride solution in water was used with different concentrations of salt. A glass bottle (ID $= 9$mm, OD $= 11$ mm, height of 25 mm) was used to hold the solution. Three boundary cases are presented in Table 2.2. For an unloaded resonator, CCC is achieved at a low DTC index. Deionized water shifts the resonance frequency due to its high dielectric constant but does not strongly affect the coupling. A NaCl solution is conductive, and as such, strongly absorbs RF, changing both the resonator frequency and the quality factor. At the salt concentration of 50g/L in the sample, DTCs indices reach their maximum value (highest capacitance). Any further increase in sodium chloride concentration made it impossible to critically couple the resonator. It is worth noticing that the 50g/L concentration exceeds physiologically relevant salt concentration (e.g., in the blood) by a factor of five. As a result, this PCBR can be used for *in vivo* imaging.

Several resonator models analogous to the one shown in Fig. 2.2 have been tested using a previously reported rapid scan (RS) EPR imaging system.[5] For example, additional loops coaxial to L shown in Fig. 2.2 were soldered together to increase the resonator volume. A three-loop resonator was found to be optimal for volume imaging of extended samples, while a single-loop resonator gives better results for surface measurements. These resonators have been used to measure a wide variety of samples as well as for *in vivo* oxygen imaging of breast tumors in mouse models.

## 2.4 The automatic digital control unit

### 2.4.1 Block diagram and working principles

A previously described modular RS EPR imaging system[5] was upgraded to permit automated tuning and coupling. In the previous system, all the external hardware units were controlled by a personal computer

(PC). PC continuously runs a multitude of processes most of which are not related to EPR. Several tasks requiring external control of the imaging system are not time sensitive. External magnetic field control in RS EPR is an example. In comparison, the task of resonator tuning and coupling must be performed within the shortest possible time. The PC, which coordinates signal flow between multiple external devices and internal computer components, may not be the optimal choice for this task. When it comes to automation, a standard industrial approach is to use microcontrollers (MC) that are the 'brains' of many electronic devices surrounding us in everyday life. In comparison with the PC, MC performs only the tasks explicitly programmed by the software developer. Many types of microcontrollers are commercially available, including the well-known Arduino. For ADiC, a Teensy 4.1 board was selected that has a clock rate of 600MHz (it can be overclocked to 1GHz). This MC supports three SPI ports and provides data handling capabilities (both analog and digital) sufficient to perform fast automated resonator tuning and coupling. Two separate ports SPI are used to control DTC and DFS as shown in Fig. 2.3.

The block diagram in Fig. 2.3 shows the essential components of the automatic digital control unit (ADiC) that perform the task of resonator tuning and coupling. Descriptions of these components are presented in Table 2.3.

As shown in Fig. 2.3, ADiC interfaces with the PC and the resonator. The automated tuning procedure is initiated by a MATLAB program that communicates with MC via a serial port. Upon receiving a command from the PC, the MC executes an algorithm (see Fig. 2.4) that establishes CCC in the shortest possible time. Following the tuning initiation command, the PC sets the digitizer (DIG in Fig. 2.3) into data acquisition mode. A U1084 PCIe card is used to capture and average periodic RS EPR signals. DIG does not measure RS data during the tuning/coupling process because MC blocks trigger signals to the digitizer. When CCC is established, MC releases triggers and DIG averages a pre-defined number of RS EPR signals. Data are transferred to the MATLAB software for further post-processing. [3,16,49]

### 2.4.2 Software and firmware

The main user-interacting program that communicates with MC and acquires data from DIG is written using MATLAB graphical user interface (GUI) application APP Designer (*appdesigner*). In addition, this GUI software controls the magnetic field scan, main magnet, and gradient units, as previously

described.[5] Two major experimental modes can be distinguished: (S) spectroscopy and (I) imaging. ADiC is performed immediately before the initiation of either mode. However, if the goal in the S-mode is to acquire a series of RS spectra over a relatively long period, retuning is performed with user-defined time intervals. In the I-mode, the acquisition of 2500-3500 projections (depending on the imaging protocol) is divided into blocks with approximately 100 projections in each. Each data block is acquired within several seconds, depending on the number of averages. Retuning is automatically executed at the beginning of each block of measurements.

MATLAB-generated frequency sweep waveforms are uploaded into the 33600A AWG module (see Table 2.3). This instrument operates in the frequency range from DC (direct current) to 120 MHz. To permit resonator tuning at a central frequency (CF) of approximately 800MHz, an eightfold frequency multiplication was implemented. After multiplication, the sweeps cover a range of CF $\pm$ 10MHz. The software permits the selection of the CF using serial communication.

As discussed above, microcontrollers have the advantage of real-time operation after a code is loaded into its firmware (see Supplemental Materials in Appendix). The Arduino integrated development environment (IDE) is used towards this goal in conjunction with a Teensyduino loader. Both Arduino and Teensy supporting software can be downloaded from their corresponding websites. A C++ library was created to coordinate commands to the DFS (see Supplemental Materials in Appendix). The block diagram describing the firmware algorithm executed by Teensy MC is shown in Fig. 2.4. The goal of this algorithm is to establish CCC within the shortest time interval. This goal is achieved by sequential switching of DTC values and measuring two parameters: (i) the lowest level of reflection and (ii) the frequency corresponding to this level. The search continues until the reflection is below a pre-selected threshold or all DTC combinations are tried. As discussed above, there are a total of 94 coupling indices, corresponding to 94 DTC values. In its search, the algorithm starts with the index $i$, selected during the previous tuning execution. If the threshold is not reached at $i$, the closest lower index ($i$ - 1) is tried, followed by ($i$ + 1), ($i$ - 2), ($i$ + 2), etc. The process proceeds until either the reflection level is below the threshold, or all 94 indices are analyzed.

MC digitizes the reflection signal from the RP synchronously with the AWG-produced frequency sweep. The measured time-domain signal is translated into the frequency domain using a linear calibration function. Short-term Fourier analysis (change of frequency as a function of time) was used for the calibration. The function's slope equals the known rate of change (20 MHz over 1 ms). The function's

offset is a measured delay between the trigger and the acquisition of the first digitized data point. The major contributor to the delay is the ZX57-40 + probe which has a rise time of 400 ns. This time is much smaller compared to the one-millisecond duration of the frequency scan. The bandwidth of the probe is larger enough not to cause significant distortion of the measured reflection signal. Additional fine-tuning of the resonance frequency is achieved by sampling two neighboring frequencies and measuring the corresponding reflection levels (see Fig. 2.4). The frequency corresponding to each measured block of data is transferred to the PC to be used for spectral alignment in post-processing.

### 2.4.3 Timing and trade-offs

The firmware algorithm described in the previous section is aimed to reduce the experimental overhead tuning/coupling time, OT, in comparison with the data acquisition time, AT. The signal-to-noise ratio (SNR) reduction expressed in percentage units can be estimated as:

$$SNRreduction = \left(1 - \sqrt{\frac{AT}{AT + OT}}\right) * 100\% \tag{2.1}$$

PC to MC serial communication constitutes the longest fixed period of approximately 10 ms. Generating and acquiring reflection signals for a given DTC index takes $\cong 1$. Fine frequency tuning (see Fig. 2.4) requires an additional $\approx 2$ ms. After the initial tuning and coupling when the sample is placed in the resonator, ADiC tracks and corrects incremental deviations from CCC. As a result, changes in the resonator DTC index change do not normally exceed one, $|\Delta i| \leq 1$, and OT $< 20$ ms. SNR reduction can be evaluated for a given ADiC repetition period using Eq. 2.1.

## 2.5 Experimental evaluations

### 2.5.1 ADiC response to abrupt change in resonator loading conditions

A test was conducted to evaluate ADiC performance under the conditions of abrupt resonator impedance change. Towards this goal, a vial containing NaCl solution was placed in and removed from the resonator. A capillary containing degassed lithium phthalocyanine (LiPc)[50] crystals was used to measure changes in EPR spectra under loaded and unloaded conditions with and without the use of ADiC. The movement of this sample did not affect the position of the LiPc crystals. Fig. 2.5 describes the results of the

conducted experiment. First, the empty resonator was tuned to CCC and the EPR spectrum (blue trace in Fig. 2.5) was acquired. After the ADiC was turned off, the salt-containing sample was inserted. As expected, neither the frequency nor DTC values changed. The corresponding spectrum is shown as a green trace in Fig. 2.5. Turning on ADiC resulted in the purple trace, as the ADiC changed DTCs values from (1, 1) to (31, 31) and readjusted the resonator frequency. The ADiC was turned off and the sample was removed. The EPR spectrum under the unloaded resonator conditions was measured (red trace).

The most dramatic change in EPR intensity was observed in the transition from an empty tuned to a loaded untuned resonator (blue to green). This is the result of a resonator frequency shift ($\approx$ 18 MHz) exceeding the resonator bandwidth ($\approx$ 10 MHz). Also, the resonator was under-coupled, which is an unfavorable condition.[51] The difference between loaded/tuned and loaded/untuned is not as substantial due to a more favorable over-coupling condition[51] and the fact that the frequency still remains within the resonator bandwidth of $\geq$40 MHz (see Table 2.2).

### 2.5.2   Long-term stability of ADiC using a sample with time-varying electric properties

The experiment described in this section was designed to test the ability of ADiC to reliably maintain CCC over a long period of time during a gradual change of resonator loading from fully loaded to empty (see Fig. 2.6). A tube containing 200 µL of saturated salt (NaCl) solution was placed in the resonator. The solution underwent electrolysis using a pair of platinum wires. A voltage of 5 V was applied to these electrodes. The purpose of using electrolysis, also accompanied by heating and evaporation, was to gradually remove the volume of water from the sample, thereby changing both resonance frequency and DTC values required to achieve CCC. As in the previous test, LiPc crystals loaded into a capillary tube provided the EPR signal. The ADiC sustained critical coupling over the entire measurement period (25 h). As expected, an overall increase in EPR frequency, in synchronization with the decrease of DTC indices, can be observed during the loaded to empty resonator transition. At the beginning of the experiment, there was an increase in DTC index values and a decrease in frequency which was attributed to an increase in the sample salinity. Continuing electrolysis caused further reduction in the sample volume which resulted in less loading. Throughout the experiment, the resonator quality factor increased fourfold, which was also reflected in EPR intensity.

### 2.5.3    In vivo imaging demonstration

Functional imaging of oxygen distribution in a breast tumor mouse model was performed using a home-built RS EPR imaging system (see Fig. 2.7). During data acquisition, ADiC was engaged. As a result, the resonator remained critically coupled throughout the experiment. A set of 3276 projections was measured. RS signals were acquired at a constant scan frequency of 8130 kHz. The four-dimensional spectral-spatial image ($355 \times 80 \times 80 \times 80$ points) was reconstructed from projections using the algorithm developed by Komarov et al.[52] 3D oxygen maps were obtained by fitting EPR spectra within each voxel.

Oxygen-sensitive probe OX071, synthesized in-house using recently published protocols,[53] was used in the experiments. A 14-week-old female FVB/N MMTV-PyMT + breast tumor-bearing mouse was anesthetized using airway isoflurane (1.5 % at 1L/min). 50 µL of 2 mM OX071 was injected into the tumor (number four mammary gland). The tumor size of $6 \times 8$ mm was measured with a caliper. The tumor depth was estimated as 6 mm. The mouse was positioned into the resonator so that the tumor was placed inside the resonator loop (ID = 11 mm). The *in vivo* experiment was performed in strict accordance with protocols (Eubank, PI: #1602000254) approved by the Institutional Animal Care and Use Committee at West Virginia University. A 2D $pO_2$ cross-section across the image and distribution of $pO_2$ within the tumor and surrounding area were shown in Fig. 2.7.

Projections were acquired in blocks of 126. At the beginning of each block, zero-gradient spectra were measured. These spectra were used to correct the magnetic field drift occurring due to the permanent magnet heating by the gradients.

## 2.6    Discussion

Automation has been and remains a major trend and the goal for many industrial applications, from washing machines to self-driving cars. The microcontroller is a key enabling technology; it is compact, fast, and highly reliable. Personal computers may be more computationally powerful compared to MC. However, you would not trust a critical process or an autopilot to a computer that may decide to install an update, reboot, or perform a noncritical task when least expected. An interruption caused by the computer may result in lost experimental time and/or sample.

The ADiC approach to EPR resonator handling is aligned with the industrial trend of digital automation. This trend is in the use of microcontrollers in concert with other digitally controlled devices

that intercommunicate using several established protocols. Serial Peripheral Interface (SPI) is one such protocol. ADiC, which uses SPI to control both resonator tuning and coupling, is compatible with any digitally controlled EPR device. It can be easily integrated into new spectrometer/imager designs and/or be used as an upgrade for existing instruments.

The PCB resonator design described (see Fig. 2.2) in this article can be modified to create a wide variety of resonators tailored to a specific application of interest. For example, the loop can be replaced by a coil with a complex shape, such as the Alderman-Grant[30,54,55] or birdcage designs.[56–59] In addition to coupling, digital resonator tuning can be implemented by the addition of a DTC in parallel with TC (see Fig. 2.2) for the use of a wider frequency tuning range. The coupling range can be increased by changing the values of fixed capacitors $FCC_1$ and $FCC_2$. However, this change will increase the related increment of the net resonator impedance, increasing the change the desired critical coupling conditions are not established. Finer adjustments can be made by the addition of a second DTC pair parallel to $FCC_1$. However, this may increase the overall tuning/coupling time. The latency can be decreased by reducing the frequency sweep time and/or increasing the digital communication speed. However, there would be an upper limit for both parameters beyond which the ADiC performance may be negatively impacted. Also, that change would require the use of a faster MC, or, alternatively, the clock rate can be increased. For example, Teensy 4.1 can be over-clocked from 600MHz to 1 GHz. The latter would require the installation of MC cooling elements. In addition, overclocking may result in clock frequency instability which would negatively impact the time to frequency domain reflection signal conversion used to compute the first guess of the resonance frequency (see Fig. 2.4). Increasingly more powerful microcontrollers are continually becoming available for the users. As a result, ADiC performance will only improve over time. Field Programmable Gate Array (FPGA) technology can be used as an alternative to the MC. FPGAs have several advantages, including high-frequency operation. However, they are more expensive and require high-level expertise.

Due to the use of a localized surface coil placed on the mammary gland of a mouse in the experiment described in the *in vivo* section 2.5.3, animal breathing did not affect tuning and coupling conditions to the extent of changing CCC. In the past, slow animal spontaneous repositioning would cause resonator decoupling and data corruption, making it challenging to perform long-term imaging. If a volume resonator was used the effect would be stronger, as the whole animal body would move inside the resonator. In this case, ADiC would help to maintain CCC.

Since the true distribution of oxygen is not known *in vivo*, a quantitative evaluation of ADiC in an imaging experiment is not feasible unless an independent method is used. However, it is obvious from the presented experimental results (Fig. 2.5) that decoupling and detuning of the resonator lead to SNR reduction which would directly translate into reduced accuracy of oxygen values in the images.

The ADiC method suits RS EPR the most because of its fast data acquisition capability. However, this method can be applied to the first-derivative continuous-wave (CW) EPR with modifications. For example, automatic tuning/coupling can be performed before every magnetic field sweep. However, such sweeps are often relatively slow compared to RS data acquisition times. As a result, ADiC would not help if a substantial deviation from critical coupling occurs during the sweep. A recently proposed fast scan EPR imaging method[33,59] would benefit from this technology. ADiC can also be used in pulsed EPR; however, DTCs and the balun (see Fig. 2.2) must be protected from excessive peak power. This can be done by the introduction of a limiter, either passive or active. CCC is not required for pulsed experiments. The resonator is often over-coupled. The use of ADiC will be beneficial if the frequency changes become comparable to the resonator bandwidth (see Fig. 2.5).

## 2.7 Data availability

Data will be made available on request.

## 2.8 Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## 2.9 Acknowledgments

## 2.10 Author Disclosure Statement

Oxana and Mark Tseytlin are the owners of a startup company BioMap5D. However, no competing financial interests exist related to this article.

# References

[1] Tseytlin M (2020) Continuous-wave rapid scan EPR. In Chechik V, Murphy DM, and Bode BE, editors, *Electron Paramagnetic Resonance*, volume 27, 188–213. Royal Society of Chemistry, Cambridge. ISBN 978-1-83916-171-1. doi:10.1039/9781839162534-00188. URL http://ebook.rsc.org/?DOI=10.1039/9781839162534-00188.

[2] Tseytlin O, O'Connell R, Sivashankar V, Bobko AA, and Tseytlin M (2021) Rapid Scan EPR Oxygen Imaging in Photoactivated Resin Used for Stereolithographic 3D Printing. 3D Printing and Additive Manufacturing 8(6):358–365. ISSN 2329-7662, 2329-7670. doi:10.1089/3dp.2020.0170. URL https://www.liebertpub.com/doi/10.1089/3dp.2020.0170.

[3] Tseytlin M (2020) General solution for rapid scan EPR deconvolution problem. Journal of Magnetic Resonance 318:106801. ISSN 10907807. doi:10.1016/j.jmr.2020.106801. URL https://linkinghub.elsevier.com/retrieve/pii/S1090780720301191.

[4] Tseytlin O, Bobko AA, and Tseytlin M (2020) Rapid Scan EPR Imaging as a Tool for Magnetic Field Mapping. Applied Magnetic Resonance 51(9-10):1117–1124. ISSN 0937-9347, 1613-7507. doi:10.1007/s00723-020-01238-6. URL https://link.springer.com/10.1007/s00723-020-01238-6.

[5] Tseytlin O, Guggilapu P, Bobko AA, AlAhmad H, Xu X, Epel B, O'Connell R, Hoblitzell EH, Eubank TD, Khramtsov VV, Driesschaert B, Kazkaz E, and Tseytlin M (2019) Modular imaging system: Rapid scan EPR at 800 MHz. Journal of Magnetic Resonance 305:94–103. ISSN 10907807. doi:10.1016/j.jmr.2019.06.003. URL https://linkinghub.elsevier.com/retrieve/pii/S1090780719301107.

[6] Sato-Akaba H and Tseytlin M (2019) Development of an L-band rapid scan EPR digital console. Journal of Magnetic Resonance 304:42–52. ISSN 10907807. doi:10.1016/j.jmr.2019.05.003. URL https://linkinghub.elsevier.com/retrieve/pii/S1090780719300874.

[7] Epel B, Sundramoorthy SV, Krzykawska-Serda M, Maggio MC, Tseytlin M, Eaton GR, Eaton SS, Rosen GM, Kao JP, and Halpern HJ (2017) Imaging thiol redox status in murine tumors in vivo with rapid-scan electron paramagnetic resonance. Journal of Magnetic Resonance 276:31–36.

ISSN 10907807. doi:10.1016/j.jmr.2016.12.015. URL https://linkinghub.elsevier.com/retrieve/pii/S1090780716302798.

[8] Eaton SS, Shi Y, Woodcock L, Buchanan LA, McPeak J, Quine RW, Rinard GA, Epel B, Halpern HJ, and Eaton GR (2017) Rapid-scan EPR imaging. Journal of Magnetic Resonance 280:140–148. ISSN 10907807. doi:10.1016/j.jmr.2017.02.013. URL https://linkinghub.elsevier.com/retrieve/pii/S1090780717300538.

[9] Raylman R, Tseytlin M, and Stolin A (2017) A combined PET-EPR system: initial testing. Journal of Nuclear Medicine 58(Supplement 1):396. ISSN 0161-5505. URL https://jnm.snmjournals.org/content/58/supplement_1/396.short.

[10] Buchanan L, Rinard G, Quine R, Eaton S, and Eaton G (2016) A Rapid Scan Method to Measure T1 Relaxation Times. Breckenridge, Colorado.

[11] Biller JR, Mitchell DG, Tseytlin M, Elajaili H, Rinard GA, Quine RW, Eaton SS, and Eaton GR (2016) Rapid Scan Electron Paramagnetic Resonance Opens New Avenues for Imaging Physiologically Important Parameters In Vivo. Journal of Visualized Experiments (115):54068. ISSN 1940-087X. doi:10.3791/54068. URL https://www.jove.com/t/54068/rapid-scan-electron-paramagnetic-resonance-opens-new-avenues-for.

[12] Möser J, Lips K, Tseytlin M, Eaton G, Eaton S, and Schnegg A (2017) Using rapid-scan EPR to improve the detection limit of quantitative EPR by more than one order of magnitude. Journal of Magnetic Resonance 281:17–25. ISSN 10907807. doi:10.1016/j.jmr.2017.04.003. URL https://linkinghub.elsevier.com/retrieve/pii/S1090780717300939.

[13] McPeak JE, Höfer P, Kacprzak S, Carl P, Weber R, Eaton SS, and Eaton GR (2020) Rapid-Scan Electron Paramagnetic Resonance of Highly Resolved Hyperfine Lines in Organic Radicals. ChemPhysChem 21(22):2564–2570. ISSN 1439-4235, 1439-7641. doi:10.1002/cphc.202000701. URL https://onlinelibrary.wiley.com/doi/10.1002/cphc.202000701.

[14] Hassan MA, Elrifai T, Sakr A, Kern M, Lips K, and Anders J (2021) A 14-channel 7 GHz VCO-based EPR-on-a-chip sensor with rapid scan capabilities. In *2021 IEEE Sensors*, 1–4. IEEE, Sydney,

Australia. ISBN 978-1-72819-501-8. doi:10.1109/SENSORS47087.2021.9639513. URL https://ieeexplore.ieee.org/document/9639513/.

[15] Laguta O, Tuček M, Van Slageren J, and Neugebauer P (2018) Multi-frequency rapid-scan HFEPR. Journal of Magnetic Resonance 296:138–142. ISSN 10907807. doi:10.1016/j.jmr.2018.09.005. URL https://linkinghub.elsevier.com/retrieve/pii/S1090780718302301.

[16] Tseytlin M (2017) Full cycle rapid scan EPR deconvolution algorithm. Journal of Magnetic Resonance 281:272–278. ISSN 10907807. doi:10.1016/j.jmr.2017.06.008. URL https://linkinghub.elsevier.com/retrieve/pii/S1090780717301635.

[17] Eaton GR and Eaton SS (2022) Advances in rapid scan EPR spectroscopy. In *Methods in Enzymology*, volume 666, 1–24. Elsevier. ISBN 978-0-323-99974-8. doi:10.1016/bs.mie.2022.02.013. URL https://linkinghub.elsevier.com/retrieve/pii/S0076687922000441.

[18] Neufeld L, Yeini E, Reisman N, Shtilerman Y, Ben-Shushan D, Pozzi S, Madi A, Tiram G, Eldar-Boock A, Ferber S, Grossman R, Ram Z, and Satchi-Fainaro R (2021) Microengineered perfusable 3D-bioprinted glioblastoma model for in vivo mimicry of tumor microenvironment. Science Advances 7(34):eabi9119. ISSN 2375-2548. doi:10.1126/sciadv.abi9119. URL https://www.science.org/doi/10.1126/sciadv.abi9119.

[19] Künstner S, Chu A, Dinse KP, Schnegg A, McPeak JE, Naydenov B, Anders J, and Lips K (2021) Rapid-scan electron paramagnetic resonance using an EPR-on-a-Chip sensor. Magnetic Resonance 2(2):673–687. ISSN 2699-0016. doi:10.5194/mr-2-673-2021. URL https://mr.copernicus.org/articles/2/673/2021/.

[20] Biller JR, Tseitlin M, Mitchell DG, Yu Z, Buchanan LA, Elajaili H, Rosen GM, Kao JPY, Eaton SS, and Eaton GR (2015) Improved Sensitivity for Imaging Spin Trapped Hydroxyl Radical at 250 MHz. ChemPhysChem 16(3):528–531. ISSN 14394235. doi:10.1002/cphc.201402835. URL https://onlinelibrary.wiley.com/doi/10.1002/cphc.201402835.

[21] Mitchell DG, Tseitlin M, Quine RW, Meyer V, Newton ME, Schnegg A, George B, Eaton SS, and Eaton GR (2013) X-band rapid-scan EPR of samples with long electron spin relaxation times: a comparison of continuous wave, pulse and rapid-scan EPR. Molecular Physics

111(18-19):2664–2673. ISSN 0026-8976, 1362-3028. doi:10.1080/00268976.2013.792959. URL http://www.tandfonline.com/doi/abs/10.1080/00268976.2013.792959.

[22] Mitchell DG, Rosen GM, Tseitlin M, Symmes B, Eaton SS, and Eaton GR (2013) Use of Rapid-Scan EPR to Improve Detection Sensitivity for Spin-Trapped Radicals. Biophysical Journal 105(2):338–342. ISSN 00063495. doi:10.1016/j.bpj.2013.06.005. URL https://linkinghub.elsevier.com/retrieve/pii/S0006349513006796.

[23] Tseitlin M, Mitchell DG, Eaton SS, and Eaton GR (2012) Corrections for sinusoidal background and non-orthogonality of signal channels in sinusoidal rapid magnetic field scans. Journal of Magnetic Resonance 223:80–84. ISSN 10907807. doi:10.1016/j.jmr.2012.07.023. URL https://linkinghub.elsevier.com/retrieve/pii/S1090780712002650.

[24] Tseitlin M, Czechowski T, Quine RW, Eaton SS, and Eaton GR (2009) Background removal procedure for rapid scan EPR. Journal of Magnetic Resonance 196(1):48–53. ISSN 10907807. doi:10.1016/j.jmr.2008.10.012. URL https://linkinghub.elsevier.com/retrieve/pii/S1090780708003315.

[25] Buchanan LA, Woodcock LB, Quine RW, Rinard GA, Eaton SS, and Eaton GR (2018) Background correction in rapid scan EPR spectroscopy. Journal of Magnetic Resonance 293:1–8. ISSN 10907807. doi:10.1016/j.jmr.2018.05.010. URL https://linkinghub.elsevier.com/retrieve/pii/S1090780718301411.

[26] Bobko AA, Dhimitruka I, Eubank TD, Marsh CB, Zweier JL, and Khramtsov VV (2009) Trityl-based EPR probe with enhanced sensitivity to oxygen. Free Radical Biology and Medicine 47(5):654–658. ISSN 08915849. doi:10.1016/j.freeradbiomed.2009.06.007. URL https://linkinghub.elsevier.com/retrieve/pii/S0891584909003402.

[27] Schaner PE, Pettus JR, Flood AB, Williams BB, Jarvis LA, Chen EY, Pastel DA, Zuurbier RA, di-Florio Alexander RM, Swartz HM, and Kuppusamy P (2020) OxyChip Implantation and Subsequent Electron Paramagnetic Resonance Oximetry in Human Tumors Is Safe and Feasible: First Experience in 24 Patients. Frontiers in Oncology 10:572060. ISSN 2234-943X. doi:10.3389/fonc.2020.572060. URL https://www.frontiersin.org/articles/10.3389/fonc.2020.572060/full.

[28] Epel B, Maggio MC, Barth ED, Miller RC, Pelizzari CA, Krzykawska-Serda M, Sundramoorthy SV, Aydogan B, Weichselbaum RR, Tormyshev VM, and Halpern HJ (2019) Oxygen-Guided Radiation Therapy. International Journal of Radiation Oncology*Biology*Physics 103(4):977–984. ISSN 03603016. doi:10.1016/j.ijrobp.2018.10.041. URL https://linkinghub.elsevier.com/retrieve/pii/S0360301618339476.

[29] Epel B, Sundramoorthy SV, Barth ED, Mailer C, and Halpern HJ (2011) Comparison of 250 MHz electron spin echo and continuous wave oxygen EPR imaging methods for *in vivo* applications: Electron spin echo and continuous wave oxygen imaging. Medical Physics 38(4):2045–2052. ISSN 00942405. doi:10.1118/1.3555297. URL http://doi.wiley.com/10.1118/1.3555297.

[30] Rinard GA, Quine RW, Buchanan LA, Eaton SS, Eaton GR, Epel B, Sundramoorthy SV, and Halpern HJ (2017) Resonators for In Vivo Imaging: Practical Experience. Applied Magnetic Resonance 48(11-12):1227–1247. ISSN 0937-9347, 1613-7507. doi:10.1007/s00723-017-0947-0. URL http://link.springer.com/10.1007/s00723-017-0947-0.

[31] Hirata H, Petryakov S, and Schreiber W (2020) Resonators for Clinical Electron Paramagnetic Resonance (EPR). In Berliner LJ and Parinandi NL, editors, *Measuring Oxidants and Oxidative Stress in Biological Systems*, volume 34, 189–219. Springer International Publishing, Cham. ISBN 978-3-030-47317-4 978-3-030-47318-1. doi:10.1007/978-3-030-47318-1_10. URL http://link.springer.com/10.1007/978-3-030-47318-1_10. Series Title: Biological Magnetic Resonance.

[32] Hyde JS, Sidabras JW, and Mett RR (2019) Uniform Field Resonators for EPR Spectroscopy: A Review. Cell Biochemistry and Biophysics 77(1):3–14. ISSN 1085-9195, 1559-0283. doi:10.1007/s12013-018-0845-6. URL http://link.springer.com/10.1007/s12013-018-0845-6.

[33] Samouilov A, Komarov D, Petryakov S, Iosilevich A, and Zweier JL (2021) Development of an L-band resonator optimized for fast scan EPR imaging of the mouse head. Magnetic Resonance in Medicine 86(4):2316–2327. ISSN 0740-3194, 1522-2594. doi:10.1002/mrm.28821. URL https://onlinelibrary.wiley.com/doi/10.1002/mrm.28821.

[34] Rinard GA, Quine RW, McPeak J, Buchanan L, Eaton SS, and Eaton GR (2017) An X-Band Crossed-Loop EPR Resonator. Applied Magnetic Resonance 48(11-12):1219–1226. ISSN 0937-

9347, 1613-7507. doi:10.1007/s00723-017-0945-2. URL http://link.springer.com/10.1007/s00723-017-0945-2.

[35] Brivati J, Stevens A, and Symons M (1991) A radiofrequency ESR spectrometer for in vivo imaging. Journal of Magnetic Resonance (1969) 92(3):480–489. ISSN 00222364. doi:10.1016/0022-2364(91)90344-S. URL https://linkinghub.elsevier.com/retrieve/pii/002223649190344S.

[36] Junwang G, Qingquan Y, Jianbo C, Lei M, Guofu D, Guoshan Y, and Ke W (2014) New Developed Cylindrical TM010 Mode EPR Cavity for X-Band In Vivo Tooth Dosimetry. PLoS ONE 9(9):e106587. ISSN 1932-6203. doi:10.1371/journal.pone.0106587. URL https://dx.plos.org/10.1371/journal.pone.0106587.

[37] Piasecki W, Froncisz W, and Hyde JS (1996) Bimodal loop-gap resonator. Review of Scientific Instruments 67(5):1896–1904. ISSN 0034-6748, 1089-7623. doi:10.1063/1.1147001. URL https://pubs.aip.org/aip/rsi/article/67/5/1896-1904/444842.

[38] Epel B, Sundramoorthy SV, and Halpern HJ (2011) Retractable loop-gap resonators for electron paramagnetic resonance imaging with in situ irradiation capabilities. Concepts in Magnetic Resonance Part B: Magnetic Resonance Engineering 39B(4):167–172. ISSN 15525031. doi:10.1002/cmr.b.20205. URL https://onlinelibrary.wiley.com/doi/10.1002/cmr.b.20205.

[39] Rinard GA, Quine RW, and Eaton GR (2000) An L-Band Crossed-Loop (Bimodal) EPR Resonator. Journal of Magnetic Resonance 144(1):85–88. ISSN 10907807. doi:10.1006/jmre.2000.2014. URL https://linkinghub.elsevier.com/retrieve/pii/S1090780700920147.

[40] Rinard GA, Quine RW, Biller JR, and Eaton GR (2010) A wire-crossed-loop resonator for rapid scan EPR. Concepts in Magnetic Resonance Part B: Magnetic Resonance Engineering 37B(2):86–91. ISSN 15525031, 1552504X. doi:10.1002/cmr.b.20161. URL https://onlinelibrary.wiley.com/doi/10.1002/cmr.b.20161.

[41] Yokoyama H, Sato T, Ogata T, Ohya H, and Kamada H (2001) Automatic Coupling Control of a Loop–Gap Resonator by a Variable Capacitor Attached Coupling Coil for EPR Measurements at 650

MHz. Journal of Magnetic Resonance 149(1):29–35. ISSN 10907807. doi:10.1006/jmre.2000.2268. URL https://linkinghub.elsevier.com/retrieve/pii/S1090780700922687.

[42] McCallum S and Resmer F (1999) Automatic coupling control system for radio frequency *in vivo* electron paramagnetic resonance based on a piezoelectric controlled capacitor. Review of Scientific Instruments 70(12):4706–4710. ISSN 0034-6748, 1089-7623. doi:10.1063/1.1150134. URL https://pubs.aip.org/aip/rsi/article/70/12/4706-4710/442115.

[43] Hirata H, Walczak T, and Swartz HM (2000) Electronically Tunable Surface-Coil-Type Resonator for L-Band EPR Spectroscopy. Journal of Magnetic Resonance 142(1):159–167. ISSN 10907807. doi:10.1006/jmre.1999.1927. URL https://linkinghub.elsevier.com/retrieve/pii/S1090780799919274.

[44] Sugawara H, Hirata H, Petryakov S, Lesniewski P, Williams BB, Flood AB, and Swartz HM (2014) Design and Evaluation of a 1.1-GHz Surface Coil Resonator for Electron Paramagnetic Resonance-Based Tooth Dosimetry. IEEE Transactions on Biomedical Engineering 61(6):1894–1901. ISSN 0018-9294, 1558-2531. doi:10.1109/TBME.2014.2310217. URL http://ieeexplore.ieee.org/document/6758389/.

[45] Petryakov S, Samouilov A, Chzhan-Roytenberg M, Kesselring E, Sun Z, and Zweier JL (2009) Segmented surface coil resonator for in vivo EPR applications at 1.1GHz. Journal of Magnetic Resonance 198(1):8–14. ISSN 10907807. doi:10.1016/j.jmr.2008.12.014. URL https://linkinghub.elsevier.com/retrieve/pii/S1090780708004205.

[46] Hirata H, He G, Deng Y, Salikhov I, Petryakov S, and Zweier JL (2008) A loop resonator for slice-selective in vivo EPR imaging in rats. Journal of Magnetic Resonance 190(1):124–134. ISSN 10907807. doi:10.1016/j.jmr.2007.10.012. URL https://linkinghub.elsevier.com/retrieve/pii/S1090780707003321.

[47] Sato-Akaba H, Emoto MC, Hirata H, and Fujii HG (2017) Design and testing of a 750 MHz CW-EPR digital console for small animal imaging. Journal of Magnetic Resonance 284:48–58. ISSN 10907807. doi:10.1016/j.jmr.2017.09.008. URL https://linkinghub.elsevier.com/retrieve/pii/S109078071730232X.

[48] Yu Z, Tseytlin M, Eaton SS, and Eaton GR (2015) Multiharmonic electron paramagnetic resonance for extended samples with both narrow and broad lines. Journal of Magnetic Resonance 254:86–92. ISSN 10907807. doi:10.1016/j.jmr.2015.03.006. URL https://linkinghub.elsevier.com/retrieve/pii/S1090780715000683.

[49] Tseitlin M, Rinard GA, Quine RW, Eaton SS, and Eaton GR (2011) Deconvolution of sinusoidal rapid EPR scans. Journal of Magnetic Resonance 208(2):279–283. ISSN 10907807. doi:10.1016/j.jmr.2010.11.015. URL https://linkinghub.elsevier.com/retrieve/pii/S1090780710003708.

[50] Afeworki M, Miller NR, Devasahayam N, Cook J, Mitchell JB, Subramanian S, and Krishna MC (1998) Preparation and EPR Studies of Lithium Phthalocyanine Radical as an Oxymetric Probe. Free Radical Biology and Medicine 25(1):72–78. ISSN 08915849. doi:10.1016/S0891-5849(98)00039-2. URL https://linkinghub.elsevier.com/retrieve/pii/S0891584998000392.

[51] Rinard G, Quine R, Eaton S, Eaton G, and Froncisz W (1994) Relative Benefits of Overcoupled Resonators vs Inherently Low-Q Resonators for Pulsed Magnetic Resonance. Journal of Magnetic Resonance, Series A 108(1):71–81. ISSN 10641858. doi:10.1006/jmra.1994.1090. URL https://linkinghub.elsevier.com/retrieve/pii/S1064185884710904.

[52] Komarov DA and Hirata H (2017) Fast backprojection-based reconstruction of spectral-spatial EPR images from projections with the constant sweep of a magnetic field. Journal of Magnetic Resonance 281:44–50. ISSN 10907807. doi:10.1016/j.jmr.2017.05.005. URL https://linkinghub.elsevier.com/retrieve/pii/S1090780717301313.

[53] Poncelet M, Huffman JL, Khramtsov VV, Dhimitruka I, and Driesschaert B (2019) Synthesis of hydroxyethyl tetrathiatriarylmethyl radicals OX063 and OX071. RSC Advances 9(60):35073–35076. ISSN 2046-2069. doi:10.1039/C9RA08633A. URL http://xlink.rsc.org/?DOI=C9RA08633A.

[54] Sundramoorthy SV, Epel B, and Halpern HJ (2014) Orthogonal resonators for pulse in vivo electron paramagnetic imaging at 250MHz. Journal of Magnetic Resonance 240:45–51. ISSN 10907807. doi:10.1016/j.jmr.2013.12.015. URL https://linkinghub.elsevier.com/retrieve/pii/S109078071400007X.

[55] Petryakov S, Samouilov A, Roytenberg M, Li H, and Zweier JL (2006) Modified Alderman-Grant resonator with high-power stability for proton electron double resonance imaging. Magnetic Resonance in Medicine 56(3):654–659. ISSN 0740-3194, 1522-2594. doi:10.1002/mrm.20992. URL https://onlinelibrary.wiley.com/doi/10.1002/mrm.20992.

[56] Tian N, Romberg J, and Sabra K (2014) Multichannel myopic deconvolution using ambient noise sources. The Journal of the Acoustical Society of America 135(4):2360–2360. ISSN 0001-4966. doi:10.1121/1.4877777. URL https://pubs.aip.org/asa/jasa/article/135/4_Supplement/2360/711005.

[57] Qian C, Masad IS, Rosenberg JT, Elumalai M, Brey WW, Grant SC, and Gor'kov PL (2012) A volume birdcage coil with an adjustable sliding tuner ring for neuroimaging in high field vertical magnets: Ex and in vivo applications at 21.1 T. Journal of Magnetic Resonance 221:110–116. ISSN 10907807. doi:10.1016/j.jmr.2012.05.016. URL https://linkinghub.elsevier.com/retrieve/pii/S1090780712002017.

[58] Chin CL, Collins CM, Li S, Dardzinski BJ, and Smith MB (2002) BirdcageBuilder: Design of specified-geometry birdcage coils with desired current pattern and resonant frequency. Concepts in Magnetic Resonance 15(2):156–163. ISSN 1043-7347, 1099-0534. doi:10.1002/cmr.10030. URL https://onlinelibrary.wiley.com/doi/10.1002/cmr.10030.

[59] Samouilov A, Ahmad R, Boslett J, Liu X, Petryakov S, and Zweier JL (2019) Development of a fast-scan EPR imaging system for highly accelerated free radical imaging. Magnetic Resonance in Medicine 82(2):842–853. ISSN 0740-3194, 1522-2594. doi:10.1002/mrm.27759. URL https://onlinelibrary.wiley.com/doi/10.1002/mrm.27759.

## 2.11 Figures
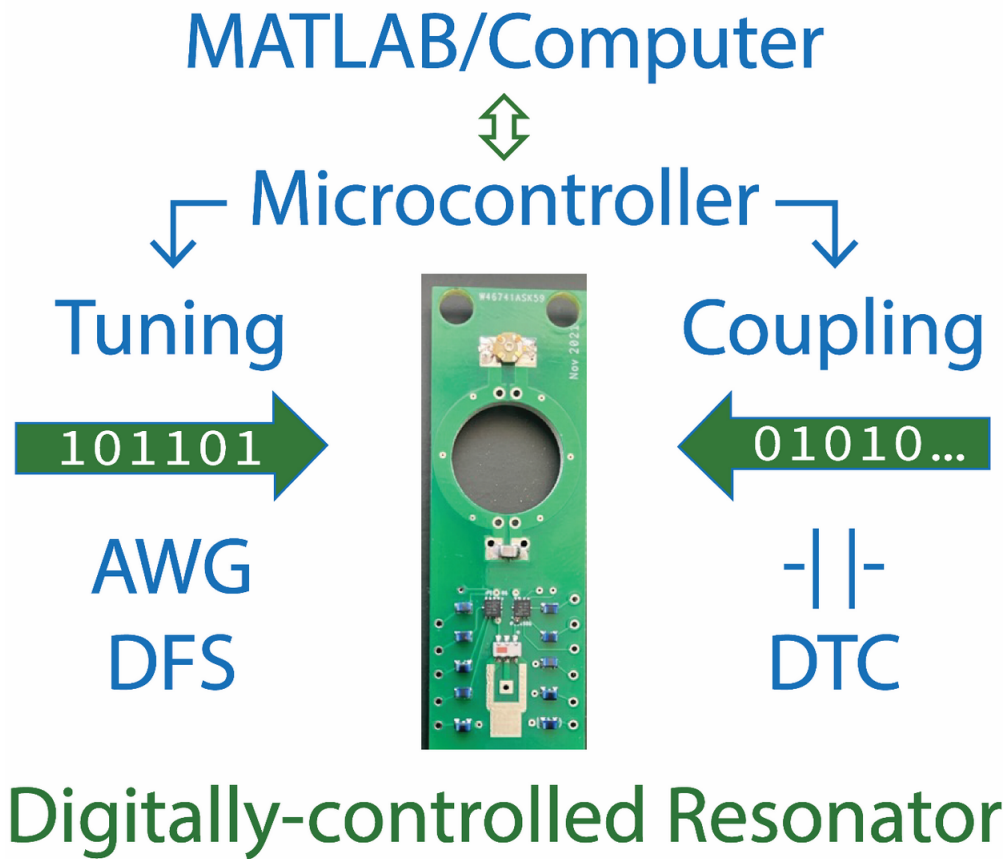


Figure GA: Graphical Abstract: Rapid scan EPR: Automated digital resonator control for low-latency data acquisition
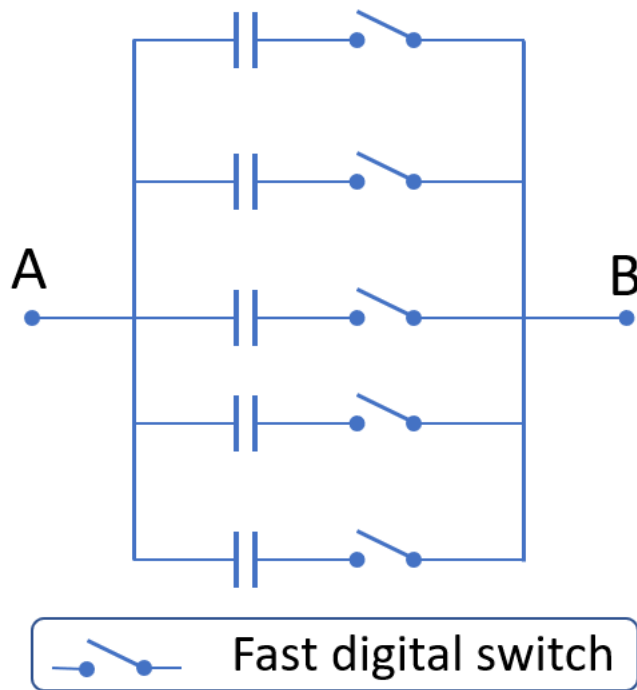
Figure 2.1: Function diagram of a digitally tunable capacitor. The net capacitance between points A and B depends on the state (on/off) of the five switches permitting $2^5 = 32$ net discrete capacitance values.



DI: Decoupling inductance
RC: Resonating capacitor
TC: Trimmer capacitor
FCC: Fixed coupling capacitors
DTC: Digitally tuned capacitors
BLN: Balun
RFC: Radiofrequency connector
L: Resonator loop
DCP: Digital control pins
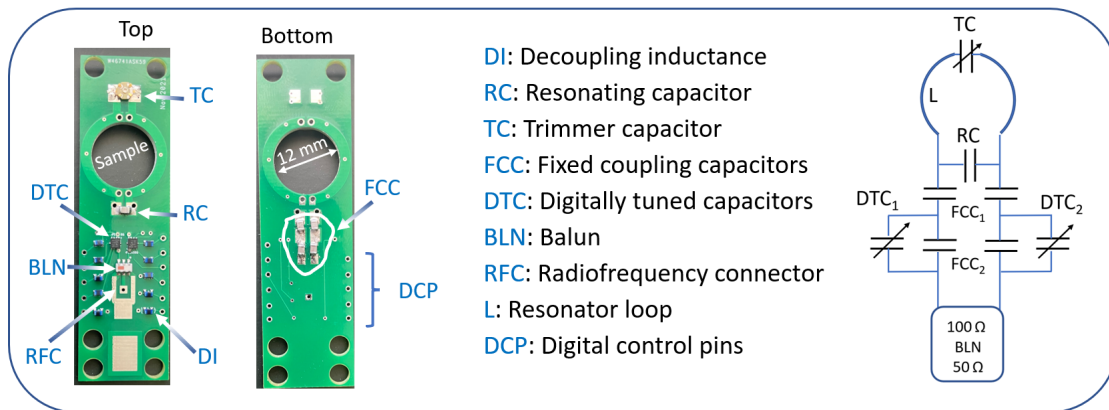
Figure 2.2: Fully functional printed circuit board resonator (PCBR). Photos of the top and bottom of the PCBR (left part of the figure) show essential resonator parts soldered on the PCB. The schematic diagram of the resonator is presented on the right part of the figure. Digital inputs to the PCB include clock, data, ground, and chip enable lines following Serial Peripheral Interface (SPI).

RES: Digitally controlled resonator
MC: Microcontroller
PC: Personal computer
SPI: Serial peripheral interface
RP: Reflection probe
AWG: Arbitrary waveform generator
DFS: Digital frequency source
DTC: Digitally tuned capacitors
DIG: Fast FPGA digitizer
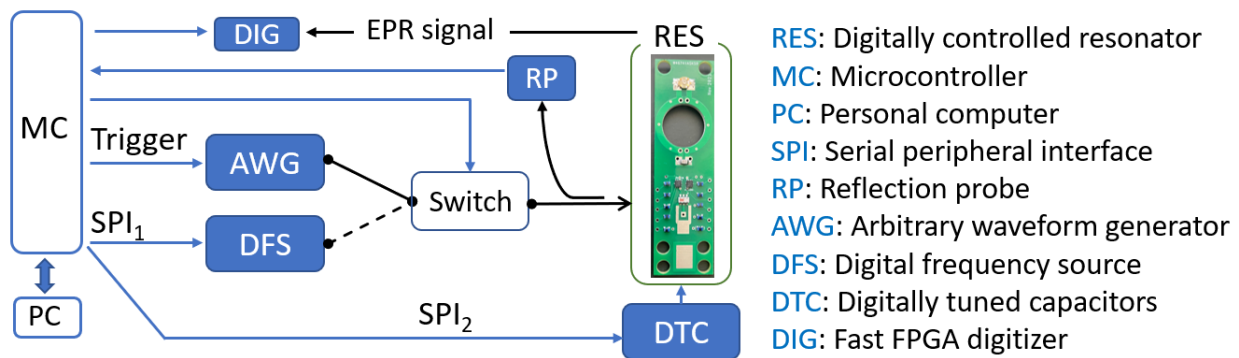
Figure 2.3: Block diagram of major components used for ADiC. The PC communicates with MC via serial protocol by sending short commands. MC communicates in real time (there are no interruptions) with all units involved in ADiC. The microcontroller sends commands to AWG, DFS, the switch, and DTCs, and reads reflection levels from BP. A pair of DTC values are sent to RES via SPI. A frequency sweep waveform (pre-loaded into AWG) is output upon receiving a trigger from MC. Immediately before this event, MC flips the digitally-controlled switch to pass the frequency sweep signal to the resonator. The signal reflected from the resonator is measured by RP and digitized by MC, which also computes the frequency corresponding to the lowest reflection. Depending on the result, this process is repeated for neighboring DTC values. The DTC values for CCC are set at the end of this procedure. The resonance frequency is established by sending a digital command via SPI to DFS. The digital switch is flipped back to enable CW data acquisition. To reduce overhead time and ensure the collection of relevant RS EPR signals, MC blocks digitizer (DIG) from receiving incoming triggers during the tuning process. The entire tuning/coupling procedure is executed in real-time without involving the PC.

93
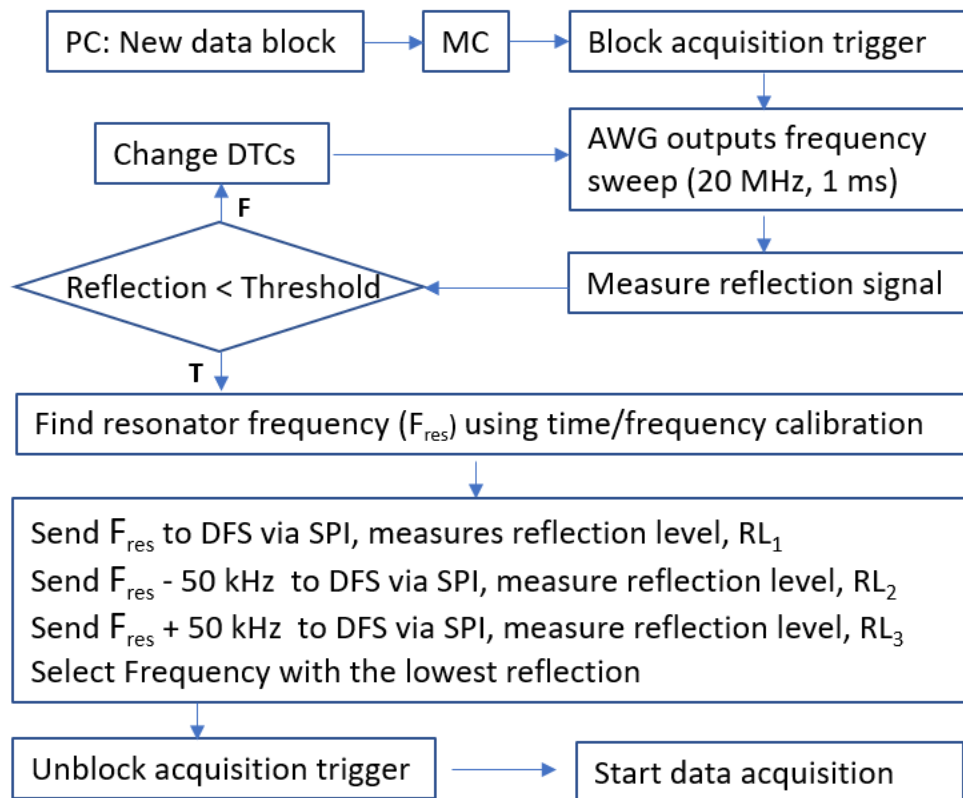
Figure 2.4: Firmware algorithm executed by Teensy MC. The PC initiates the tuning procedure by sending a serial command. After receiving this command, MC performs several steps aimed to find a pair of DTC values and the frequency that minimizes reflection from the resonator. Immediately after initiating tuning, the PC sends a command to the digitizer (DIG in Fig. 2.3) to measure RS EPR data. However, MC blocks the acquisition until the critical coupling conditions are established. As a result, there is no delay between the end of the tuning cycle and data acquisition. The reflection threshold is selected to satisfy the critical coupling ($\approx$ -40 dB reflection compared to the incident power).
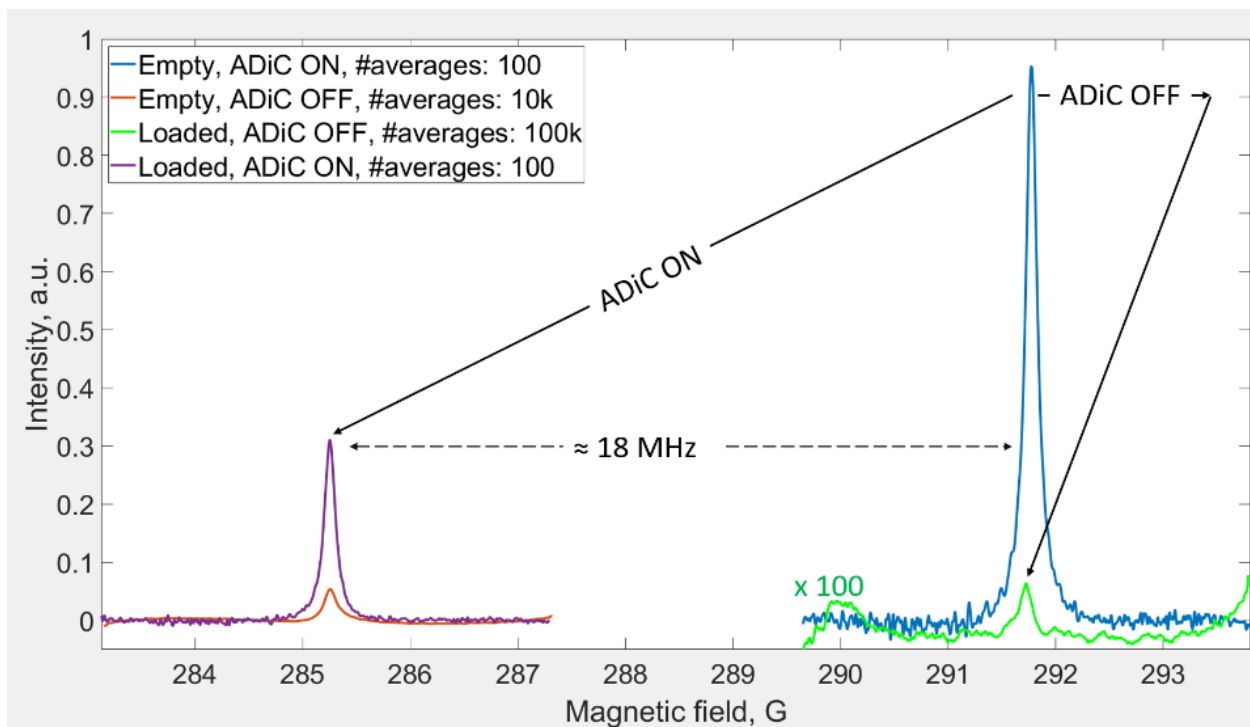
Figure 2.5: ADiC test over widely varied resonator loading conditions. EPR spectra were measured with and without ADiC when a tube filled with NaCl solution was used to abruptly change the resonator impedance.
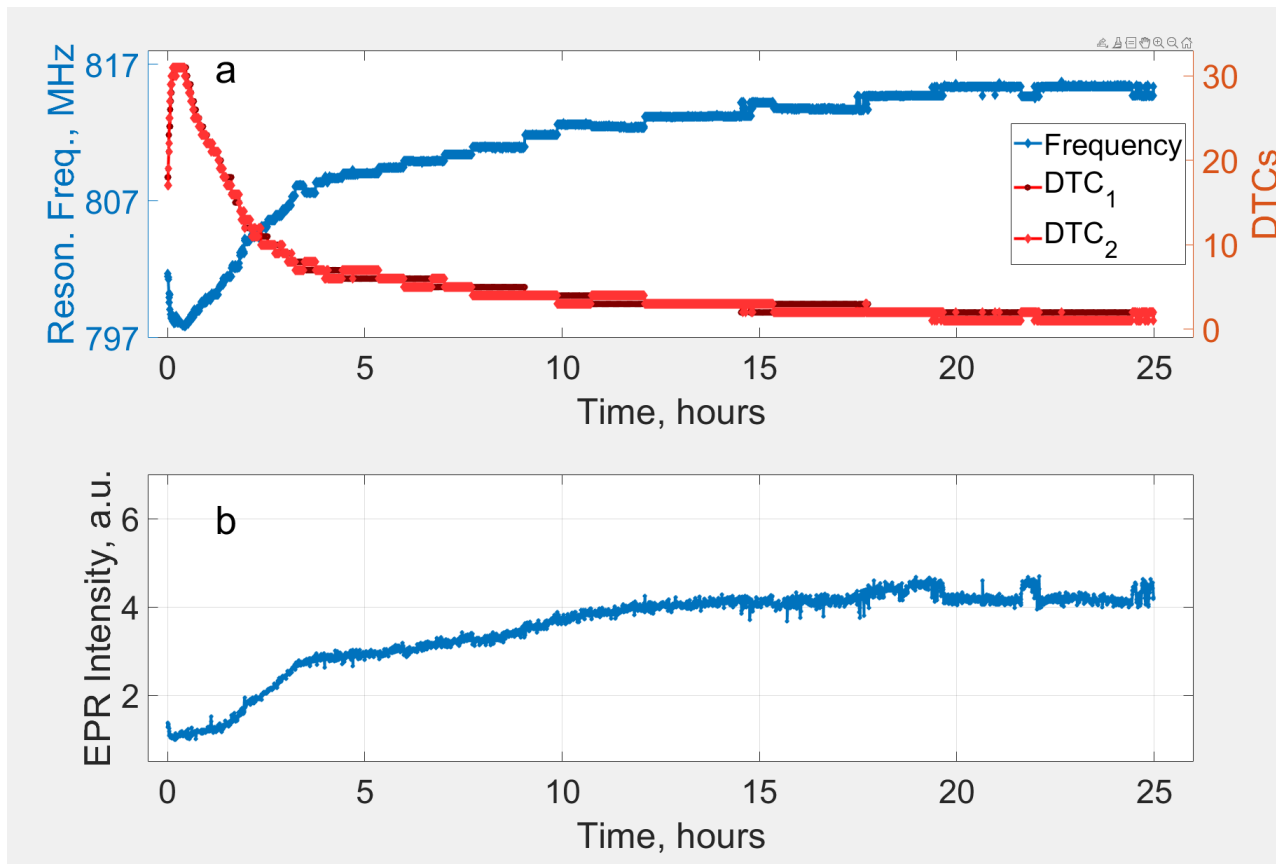
Figure 2.6: Long-term experiment showing critical resonator coupling as resonator coupling changes. a) Resonance frequency (blue) and DTC values (red and black) change over time as the resonator coupling conditions change. b) EPR spectral intensity remains optimal over time. A total of 3000 EPR spectra were successfully collected with a time interval of 30 s without a single failure.
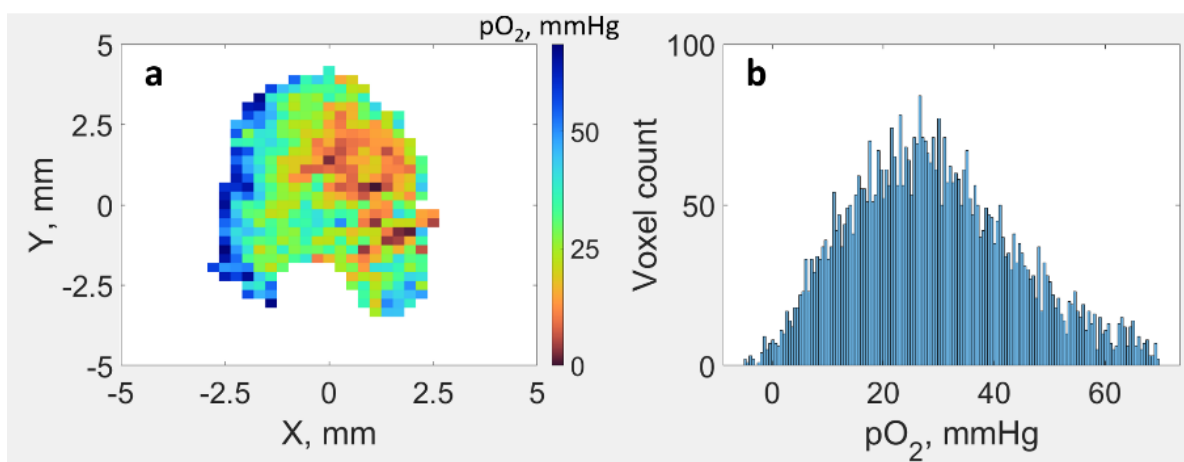


Figure 2.7: *In vivo* EPR image quantifying oxygen partial pressure ($pO_2$) in a murine mammary tumor. (a) 2D cross-section of 3D oxygen map. (b) Histogram of $pO_2$ values obtained from 3D image.

| Parameter | Range | Unit |
|---|---|---|
| Resonance frequency range (TC-defined) | 700-1000 | MHz |
| Max RF power (continuous) | 34 | dBm |
| DTC resolution | 5 | bits |
| Effective DTC1 and 2 resolution | 6.6 | bits |
| DTC range | 0.9 - 4.6 | pF |
| SPI clock | 1-25 | MHz |
| Capacitance switching time | 1-20 (depends on SPI clock frequency) | µs |
| Quality factor (unloaded, CCC) | 83 | |
| Digital input high | 1.2 - 3.1 | V |
| Digital input low | 0 - 0.6 | V |
| Power supply | 2.3 - 4.8 | V |

Table 2.1: PCB resonator specifications

| Sample | $DTC_1/DTC_2$ | Resonance frequency | Quality factor | Bandwidth |
|---|---|---|---|---|
| Empty | 1/2 | 824.5 MHz | 83 | $\approx$ 10 MHz |
| Deionized water | 2/2 | 810.5 MHz | 70 | $\approx$ 12 MHz |
| NaCl solution, 50 g/L | 31/31 | 798 MHz | 20 | $\approx$ 40 MHz |

Table 2.2: Measurements at critical coupling conditions and filling factor of 70%

| | Name | Vendor | Key specifications and use details |
|---|---|---|---|
| MC | Teensy 4.1 | PJRC, USA | Clock: 600 MHz |
| DIG | U1084 | Keysight, USA | Averaging at sampling rate: 62.5 MS/s |
| AWG | 33600A | Keysight, USA | The output frequency is multiplied by eight |
| Switch | ZSW2-63DR+ | Minicircuits, NY USA | Switching time: 1.6 µs |
| DFS | SC5511a | SignalCore, TX USA | Phase noise 1 GHz @ 10 kHz: -137 dBc/Hz |
| RP | ZX47-40+ | Minicircuits, NY USA | Pulse response time: 800 ns |
| DI | LQW18AN_8Z | Murata Electronics, Japan | 300 nH |
| DTC | PE64906 | Peregrine Semiconductors, USA | 0.9 - 4.6 pF, 5 bits |

Table 2.3: Descriptions of the components in the block diagram shown in Fig. 2.3

# Chapter 3

# Rapid scan EPR oxygen imaging in photoactivated resin used for stereolithographic 3D printing

**Authors:** Oxana Tseytlin[1,2,*], Ryan O'Connell[1,2,*], Vignesh Sivashankar[3], Andrey A. Bobko[1,2], and Mark Tseytlin[1,2,4]

* Authors contributed equally to this article.

**Affiliations:**

[1] Biochemistry Department, West Virginia University, Morgantown, West Virginia, USA

[2] In Vivo Multifunctional Magnetic Resonance Center at Robert C. Byrd Health Sciences Center, West Virginia University, Morgantown, West Virginia, USA

[3] Statler College of Engineering and Mineral Resources, West Virginia University, Morgantown, West Virginia, USA

[4] West Virginia University Cancer Institute, Morgantown, West Virginia, USA

**Note:** This chapter includes text and figures from the publication above.

## 3.1 Abstract

Oxygen plays a critical role in the photopolymerization process resulting in the formation of solid structures from liquid resins during three-dimensional (3D) printing: it acts as a polymerization inhibitor. Upon exposure to light, oxygen is depleted. As a result, the polymerization process becomes activated. Electron paramagnetic resonance (EPR) imaging is described as a tool to visualize changes in oxygen distribution caused by light exposure. This nondestructive method uses radio waves and, therefore, is not constrained by optical opacity offering greater penetrating depth. Three proof-of-principle imaging experiments were demonstrated: (1) spatial propagation of the photopolymerization process; (2) oxygen depletion as a result of postcuring; and (3) oxygen visualization in a 3D-printed spiral model. Commercial stereolithography (SLA) resin was used in these experiments. Lithium octa-n-butoxynaphthalocyanine (LiNc-BuO) probe was mixed with the resin to permit oxygen imaging. Li-naphthalocyanine probes are routinely used in various EPR applications because of their long-term stability and high functional sensitivity to oxygen. In this study, we demonstrate that EPR imaging has the potential to become a powerful visualization tool in the development of 3D printing technology, including bioprinting and tissue engineering.

## 3.2 Introduction

Ambient oxygen plays a critically important role in the photopolymerization process acting as an inhibitor. Modern three-dimensional (3D) stereolithography printers use this property of oxygen to spatially separate areas where the polymerization process should take place from the bulk resin.[1,2] Specially designed oxygen-permeable materials are used to achieve this goal.[1] No analytical imaging method is currently available to directly observe oxygen distribution during 3D printing and postcuring. Electron paramagnetic resonance imaging (EPRI) permits such measurements.[3–8] EPRI has the potential to become a useful tool for the developers of new 3D printers and photopolymerized resins. The first oxygen imaging results in the application to 3D printing are presented in this article.

Electron paramagnetic resonance spectroscopy is used to measure compounds containing unpaired electrons, such as metal complexes, free radicals, and oxygen.[9,10] EPRI is in many ways similar to the well-known magnetic resonance imaging (MRI) modality. Both methods are based on the same quantum

mechanical principles, such as interactions between spins and magnetic fields.

In the case of EPR, electron spin is used as the reporter of inter- and intramolecular interactions. EPR imaging is routinely used in biomedical research to observe changes in the microenvironment of tissues and tumors.[11–15] Chemically stable electron spin probes have been developed to measure redox potential, pH, inorganic phosphate concentration, enzymatic activity, and oxygen partial pressure $pO_2$.[6,11,13,16–24] These parameters are important biological markers that are used to study various physiological conditions but can also be applied for use in general 3D printing applications.

The goal for this article is to explore the application of EPR imaging to 3D printing. Toward this goal, proof-of-concept experiments were conducted to map $pO_2$ in photopolymerized resin after light exposure. During polymerization, molecules of monomers and/or oligomers cross-link to form solid structures. Atmospheric oxygen, which is naturally dissolved in the resin, acts as an inhibitor or the reaction.[25] First, it quenches the photoexcited photoinitiator that is used to generate initial free radicals.[26] Second, $O_2$ molecules attack radical species to form peroxides[27] that remove oxygen from the media. Due to slow oxygen diffusion in resins, $O_2$ is not immediately replenished, and once removed, the polymerization chain reaction can begin.

Developers of stereolithography printers implement an ingenious engineering solution that takes advantage of oxygen dissolved in the 3D printing media. The 3D printers are designed to utilize a thin "dead zone" layer that separates the liquid resin and the bed of the vat, through which light is projected, to form two-dimensional (2D) slices of the printed 3D objects.[2] This window is made from an oxygen-permeable material that permits constant oxygenation of the "dead zone." The formation of this zone creates a liquid interface between the freshly printed object and the vat window. This technology permits fast and reliable printing.

An alternative fast 3D printing approach (in which oxygen also is a key player) was recently published in Science.[28] The 3D objects are created in bulk liquid media (resin) using a mathematical algorithm developed for computer tomography. In this method, the back-projection of 2D masks forms a 3D object. Constructive interference creates areas with high light intensity where oxygen is quickly depleted. When the depletion level exceeds a certain threshold value, polymerization chain reaction activates. In this method, the control and detection of $pO_2$ levels by the means of EPR oximetry will permit fine-tuning of the mechanical properties of 3D-printed or bioprinted parts.[27,29]

Monitoring oxygen distribution with the help of EPR imaging during the polymerization process

may provide important information to improve the resolution and quality of 3D prints. Timing of light exposure and intensity can be optimized for a given oxygen concentration and diffusion coefficient within the printing volume. This optimization would ensure a high-threshold nonlinear oxygen depletion response to light, which is an important factor in the creation of high-resolution 3D objects.

Micro EPR imaging[30] can be used to optimize the microfabrication process. Several techniques have been developed that take advantage of oxygen inhibitory properties in 3D printing.[31,32] Computational modeling of oxygen diffusion and the polymerization process is used to optimize manufacturing.[33] EPR imaging can directly measure oxygen concentration with a spatial resolution of 30 μm or better[30,34] to verify and fine-tune the theoretical models.

To date, only 2D spectral-spatial EPR imaging has been used to observe the polymerization process.[35] Changes in EPR spectra of radicals along a single spatial dimension were measured, and no oxygen imaging has been done. Allylic radical formation resulting from light-induced resin polymerization[35] was visualized. Such spatial images are not informative for most 3D printing applications.

In comparison, this article demonstrates 4D spectral-spatial imaging results. These images comprise of three Cartesian dimensions ($x,y,z$) and a single spectral dimension (EPR intensity as a function of the magnetic field). Absorption EPR spectra are analyzed to give local oxygen concentration. This is done by nonlinear curve fitting. The width of the EPR spectrum is a linear function of oxygen partial pressure. The fitting procedure transforms 4D spectral-spatial images into 3D oxygen maps. In this article, we present the first oxygen images in 3D prints containing photoinduced resin.

Commercially available Formlabs "Clear" resin, a photopolymerizable, methacrylic acid ester-based liquid substance, was used in the imaging experiments described here. We routinely use the Formlabs Form 3 printer (Formlabs, Somerville, MA) to manufacture various parts of a locally-built EPR imaging system with this resin.[3] The introduction of stable EPR probes into the resin makes it possible to monitor both the static oxygen concentration and its kinetics.

Three types of proof-of-concept experiments were conducted. First, we observed the propagation of the polymerization front upon exposure to light and subsequent depletion of oxygen. In the second experiment, the process of postcuring was imaged. Oxygen levels in the print were imaged before and after postcuring. Finally, oxygen partial pressure in a 3D-printed object was imaged. Towards this goal, an Ender-3 Pro printer was modified to extrude resin instead of plastic.

The EPR oximetry imaging method described in this article is not limited to a specific resin type

but can be used with any photopolymerizable resin. Oxygen inhibition is encountered in a very wide range of polymerization reactions. We envision applications of EPR imaging to 3D bioprinting, where a wider range of probes [16,17,36–39] can be used to monitor changes in both the prints and the cell environment. We hope that this article will introduce EPR imaging into the field of 3D printing and stimulate interdisciplinary collaborations.

## 3.3 Materials and Methods

### 3.3.1 EPR imaging system

A recently described modular rapid scan (RS) EPR imaging system [3] was used. The RS EPR method has been demonstrated to provide superior sensitivity in comparison with the traditional first-harmonic continuous-wave EPR method. [40,41] Crucially, a larger number of projections can be acquired per unit of time, which is critical for imaging.

### 3.3.2 Modification of 3D printer

An open-source extrusion-type 3D printer (Creality Ender-3 Pro, Shenzhen Creality 3D, China) was modified for stereolithography resin extrusion (Fig. 3.1). The extruder and its stepper motor were replaced with a linear actuator (S20-3030-B; Actuonix, Canada) and a 1 mL Luer-Lok Tip syringe (BD, NJ) with a 23-gauge stainless steel blunt needle (Component Supply Co., TN) to extrude the resin. A light source (405nm, MTPCB-0842; Marktech Optoelectronics, NY) was added to photopolymerize the resin. The firmware of the 3D printer was reprogrammed to control the movement and extrusion of the actuator. Bed and extruder heating were disabled.

### 3.3.3 Sample preparation

Commercial "Clear" resin (FLGP CL04; Formlabs) was used for oximetric EPR imaging. This resin is a mixture of methacrylic acid esters and a diphenyl (2,4,6-trimethylbenzoyl) phosphine oxide (DPTBP) as a photoinitiator. [42] "Clear" resin, as its name suggests, is a semitransparent viscous substance. Upon exposure to blue light (405nm wavelength), the resin solidifies forming semitransparent solid 3D objects.

To permit oxygen EPR imaging, lithium octa-n-butoxynaphthalocyanine (LiNc-BuO) microcrystals were dispersed in the resin at various concentrations. LiNc-BuO is an oxygen-sensing particulate micro-

crystalline probe used for longitudinal oximetry measurements in preclinical and clinical EPR applications.[43–47] Crystal sizes varied in the range of approximately 2-10µm (Fig. 3.2). LiNc-BuO (molecular weight [MW]=1297 g/mol) crystals are hydrophobic compounds and can be easily dispersed, but not solubilized, in the methacrylic acid ester resin using a shaker and/or sonicator. LiNc-BuO was found to be insoluble in the monomer. Long-term storage of the prepared mixture (3 months) does not compromise the probe.

The resin containing LiNc-BuO was calibrated with respect to the linewidth in response to oxygen partial pressure. Several oxygen EPR probes, India ink, for example, characteristically display nonlinear dependence of linewidth on oxygen. However, LiNc-BuO has been found to have a linear dependence of linewidth on oxygen concentration, independent of crystal size and preparation procedure.[48]

Three processes were investigated: (1) propagation of the front of photopolymerization in a glass bottle filled with liquid resin; (2) postcuring of a polymerized resin; and (3) imaging of a 3D printed spiral object.

- A 2mL glass bottle with a diameter of 11 mm was filled with "Clear" resin containing LiNc-BuO at 0.5 mg/mL. High-efficiency violet LZ1-00UB00 LED (LED Engin, San Jose, CA) was used as the light source for photopolymerization. This small-size surface-mount chip (4.4 x 4.4 mm footprint) emits within a narrow wavelength band between 385 and 410 nm. This range matches that specified for the commercial Formlabs resin that was used in the imaging experiments. As explained above, the light-emitting diode (LED) activates the photoinitiator and starts the polymerization chain reaction after the depletion of oxygen. The bottle containing the probe/resin mixture was placed in the EPR resonator of the RS EPR imaging system[3] (Fig. 3.3a), and the LED was positioned underneath the sample. As a result, light (and polymerization) propagated in the upward direction (Fig. 3.3b). The LED was placed slightly off the bottle's center. Three images were acquired after 3, 7, and 82 s of the total light exposure time. Images were measured directly following light exposure to observe the propagation of the polymerization front throughout the sample.

- A solid cylinder was formed by sequential (layer by layer) photopolymerization of the probe/resin mixture of 8 mg/mL LiNc-BuO. The probe/resin mixture was added into a cup (ID=8.8 mm) and photopolymerized to form five layers, each ∼0.9 mm in height, resulting in a polymerized cylinder

with a height of 4.5 mm. Each layer was exposed to LED light for 2s.

- A 3D spiral was designed with OpenSCAD and printed on the modified Ender-3 Pro with a mixture of resin and 32mg/mL LiNc-BuO crystals. The spiral was printed with 20 layers of 0.04 mm height, resulting in a structure $\sim$0.8 mm high. The spiral had an outside diameter of 10 mm. Each layer was exposed to LED light for 2 s.

### 3.3.4 Rapid scan EPR imaging

A locally-built, modular, RS EPR system operating at 800 MHZ[3] was used for spectroscopy and imaging. RS EPR signals were acquired and then postprocessed into absorption spectra.[49] For both spectroscopy and imaging, the scan frequency was 9400 Hz. Scan amplitude varied depending on the gradient values. After the 4D spectral-spatial images were reconstructed, EPR spectra corresponding to each voxel were line fitted to extract linewidth and integral spin intensity proportional to the spin-probe concentration. MATLAB Curve Fitting app was used to fit EPR absorption spectra which had a Lorentzian shape. As a result of the fitting, 3D linewidth and concentration maps were obtained. The linewidth maps were translated into oxygen images using the known linear dependence of LiNc-BuO spectral width (SW) as a function of $O_2$ partial pressure, $pO_2$.[45]

## 3.4 Results

### 3.4.1 Imaging of the photopolymerization process

LiNc-BuO oxygen probe was homogeneously distributed throughout the bottle volume. However, areas containing fully oxygenated probes are not visible in the image. This happens due to two reasons. First, the oxygenated (at ambient partial oxygen pressure) LiNc-BuO probe has a broad relative to the image SW EPR linewidths. Magnetic field scans at low gradient values are comparable to SW. RS deconvolution algorithm was designed to treat any line comparable in width to the scan amplitude as background. As a result, broad spectral components are subtracted from the reconstructed EPR absorption lines.[50] It is also important to note that due to the inverse relationship between RS EPR linewidth and intensity, degassed LiNc-BuO crystals will produce x 25 higher intensity spectra compared to an analogous sample in an ambient 21% oxygen atmosphere.

The cumulative result of these effects is a sharp spatial contrast in EPR intensity images (Fig. 3.4a-c). The advancement of the polymerization front is visible with increased light exposure. A total of 2546 projections were acquired per image, including 26 intermediate zero-gradient spectra. These spectra were used for fine adjustments of the drifting external magnetic field, which occurs as a result of heating of the permanent magnet by the gradient coils. Total image acquisition time was ∼2.5 min. A maximum gradient of 3 G/cm was used to measure projections.

Four-dimensional spectral-spatial images were reconstructed using a previously described algorithm[51] developed by Komarov and Hirata. The four dimensions in the images include three Cartesian coordinates ($x, y, z$) and a spectral coordinate (EPR spectrum as a function of the scanning magnetic field). As a result, each voxel (3D volume element within the image) contains an EPR spectrum (4th dimension). During the postprocessing, the spectra are line fitted to extract two (or more depending on the probe being used) parameters: oxygen and spin probe intensity. Only Figure 3.4a-c shows spin intensity 3D maps extracted from the 4D EPR images. The rest of the results show 3D cross-sections and histograms obtained from 3D $pO_2$ maps. These maps were extracted from 4D EPR images by the means of the EPR line fitting procedure.

Due to the off-center placement of the LED source, the polymerization process propagates asymmetrically. In addition, light reflection and refraction along the glass bottle walls expose resin close to the wall surface to higher intensity LED irradiation. As a result, polymerization near the walls advanced further compared to the center of the bottle. The depletion of oxygen throughout the volume is shown in the histograms (Fig. 3.4g-i). The LED circuit was nonmagnetic and contained a minimal amount of metal so no effect on the EPR resonator and scanning field was detectable.

### 3.4.2 Imaging of postcuring process

It is common practice to postprocess photopolymerized 3D printed objects by simultaneous light exposure and heating (often to 60°C). Postcuring is required to achieve the desired mechanical properties of 3D prints. Timing and light intensity need to be optimized for the object's size and the resin used. In industrial use, an unnecessarily long curing process may reduce the manufacturing throughput and the profit margins, as a result. In addition, overexposure may damage the polymer chains, weakening the mechanical strength of a printed object. EPR imaging is proposed as a tool to monitor and optimize the postcuring process. Toward this goal, a cylinder (described in Sample preparation 2) was polymerized

under ultraviolet (UV) light and imaged with EPR. The cylinder was then postcured for 30 min at 60°C in the Form Cure machine (Formlabs) and imaged again. Voxels in 4D EPR images measured before and after postcuring were processed to obtain two 3D oxygen maps using line-fitting procedure described above. The difference between the two $O_2$ maps (Fig. 3.5) demonstrates changes in oxygen distribution as a result of curing. Histograms of $pO_2$ distribution in the photopolymerized cylinder before and after postcuring are presented in Figure 3.6.

### 3.4.3  Imaging of 3D printed spiral

Oxygen distribution in a 3D-printed spiral was visualized. Oxygen concentration was found to be higher at the perimeter of the spiral, and in areas with less resin. The CAD model (Fig. 3.7a) was sliced with Simplify3D and printed using the modified Ender-3 Pro printer (Fig. 3.1). The 3D printed spiral shown in Figure 3.7b was imaged. The oxygen image (2D slice from 3D map) shown in Figure 3.7c demonstrates a close resemblance in shape with the printed object.

## 3.5  Discussion

Proof-of-concept experiments were conducted to demonstrate the capability of EPR imaging to monitor the light-induced consumption of oxygen in the resin used for 3D stereolithography (SLA) printing. It was shown that the incorporation of oxygen-sensitive EPR probe into SLA resin permits monitoring of the photopolymerization and postcuring processes. The latter can be used to time- and quality-optimize postprocessing of 3D printed objects. EPR imaging can also be used to visualize internal structures of the prints that are not visible to the naked eye. In this capacity, EPR can serve as an alternative to other imaging modalities, such as computed X-ray tomography (CT). In comparison with CT, EPR, in addition to spatial information, measures oxygen distribution in the printed objects.

To demonstrate the feasibility of 3D printing using SLA resin impregnated with an EPR probe, an extrusion-type 3D printer was modified (Fig. 3.1). A spiral model (Fig. 3.7) was printed and imaged. The quality of the print was not perfect given the use of a modified plastic printer. However, the EPR image captured the real shape quite accurately. The introduction of oxygen-sensing probes into resins used in the commercial 3D SLA systems will improve the print quality and permit oxygen imaging.

The imaging technology described in this article has the potential to become a practical tool for

the developers of 3D printers and resins. Imaging of oxygen will provide a unique insight into the polymerization process not achievable with any other technique. Further studies need to be conducted to optimize probe concentration. It needs to be low enough to ensure no interference with the printing process and high enough to produce high-quality oxygen images. Faster image acquisition will be more informative in capturing oxygen diffusion real-time, during printing. Increasing gradient strength to about 30G/cm will produce oxygen mapping throughout the entire sample, including fully oxygenated (21%) regions. However, in this case, the reduction of sensitivity may need to be compensated by increasing the probe concentration. Imaging of the postprint curing process may become a valuable tool to optimize both duration and light intensity.

The monitoring of oxygen and other important biomarkers in bioprinted structures in another future direction for this technology. We hope that this article will serve as a starting point for future collaborative research with manufacturers and developers of 3D printers and other interested parties. For example, we envision the application of EPR oximetry to 3D printing in dental research and clinical practices.[52,53] EPR imaging could be additionally used to study polymer degradation over time caused by atmospheric oxygen.[54]

## 3.6   Disclaimer

The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health.

## 3.7   Author Disclosure Statement

No competing financial interests exist.

## 3.8   Funding Information

This work was supported by the NIH grants R01-EB023888, U54GM104942, and P20GM121322. This project was supported by the RAP WVU program for undergraduate students (V.S.).

# References

[1] Jiang Y, Wang Y, He H, Feinerman A, and Pan Y (2020) Constrained Window Design in Projection Stereolithography for Continuous Three-Dimensional Printing. 3D Printing and Additive Manufacturing 7(4):163–169. ISSN 2329-7662, 2329-7670. doi:10.1089/3dp.2019.0134. URL https://www.liebertpub.com/doi/10.1089/3dp.2019.0134.

[2] Tumbleston JR, Shirvanyants D, Ermoshkin N, Janusziewicz R, Johnson AR, Kelly D, Chen K, Pinschmidt R, Rolland JP, Ermoshkin A, Samulski ET, and DeSimone JM (2015) Continuous liquid interface production of 3D objects. Science 347(6228):1349–1352. ISSN 0036-8075, 1095-9203. doi:10.1126/science.aaa2397. URL https://www.science.org/doi/10.1126/science.aaa2397.

[3] Tseytlin O, Guggilapu P, Bobko AA, AlAhmad H, Xu X, Epel B, O'Connell R, Hoblitzell EH, Eubank TD, Khramtsov VV, Driesschaert B, Kazkaz E, and Tseytlin M (2019) Modular imaging system: Rapid scan EPR at 800 MHz. Journal of Magnetic Resonance 305:94–103. ISSN 10907807. doi:10.1016/j.jmr.2019.06.003. URL https://linkinghub.elsevier.com/retrieve/pii/S1090780719301107.

[4] Epel B, Krzykawska-Serda M, Tormyshev V, Maggio MC, Barth ED, Pelizzari CA, and Halpern HJ (2017) Spin Lattice Relaxation EPR pO2 Images May Direct the Location of Radiation Tumor Boosts to Enhance Tumor Cure. Cell Biochemistry and Biophysics 75(3-4):295–298. ISSN 1085-9195, 1559-0283. doi:10.1007/s12013-017-0825-2. URL http://link.springer.com/10.1007/s12013-017-0825-2.

[5] Elas M, Magwood JM, Butler B, Li C, Wardak R, DeVries R, Barth ED, Epel B, Rubinstein S, Pelizzari CA, Weichselbaum RR, and Halpern HJ (2013) EPR Oxygen Images Predict Tumor Control by a 50% Tumor Control Radiation Dose. Cancer Research 73(17):5328–5335. ISSN 0008-5472, 1538-7445. doi:10.1158/0008-5472.CAN-13-0069. URL https://aacrjournals.org/cancerres/article/73/17/5328/584734/EPR-Oxygen-Images-Predict-Tumor-Control-by-a-50.

[6] Matsumoto K, Kishimoto S, Devasahayam N, Chandramouli GVR, Ogawa Y, Matsumoto S, Krishna MC, and Subramanian S (2018) <span style="font-variant:small-caps;" >EPR</span> -based

oximetric imaging: a combination of single point-based spatial encoding and <span style="font-variant:small-caps;">T</span> $_1$ weighting. Magnetic Resonance in Medicine 80(5):2275–2287. ISSN 0740-3194, 1522-2594. doi:10.1002/mrm.27182. URL https://onlinelibrary.wiley.com/doi/10.1002/mrm.27182.

[7] Taguchi A, DeVience S, Driesschaert B, Khramtsov VV, and Hirata H (2020) *In vitro* simultaneous mapping of the partial pressure of oxygen, pH and inorganic phosphate using electron paramagnetic resonance. The Analyst 145(9):3236–3244. ISSN 0003-2654, 1364-5528. doi:10.1039/D0AN00168F. URL http://xlink.rsc.org/?DOI=D0AN00168F.

[8] Fujii H, Itoh K, Pandian RP, Sakata M, Kuppusamy P, and Hirata H (2007) Measuring Brain Tissue Oxygenation under Oxidative Stress by ESR/MR Dual Imaging System. Magnetic Resonance in Medical Sciences 6(2):83–89. ISSN 1347-3182, 1880-2206. doi:10.2463/mrms.6.83. URL https://www.jstage.jst.go.jp/article/mrms/6/2/6_2_83/_article.

[9] Weil JA and Bolton JR (2007) *Electron paramagnetic resonance: elementary theory and practical applications*. Wiley-Interscience, Hoboken, N.J, 2nd ed edition. ISBN 978-0-471-75496-1. OCLC: ocm69020990.

[10] Schweiger A and Jeschke G (2001) *Principles of pulse electron paramagnetic resonance*. Oxford University Press, Oxford, UK ; New York. ISBN 978-0-19-850634-8.

[11] Poncelet M, Driesschaert B, Tseytlin O, Tseytlin M, Eubank TD, and Khramtsov VV (2019) Dextran-conjugated tetrathiatriarylmethyl radicals as biocompatible spin probes for EPR spectroscopy and imaging. Bioorganic & Medicinal Chemistry Letters 29(14):1756–1760. ISSN 0960894X. doi:10.1016/j.bmcl.2019.05.017. URL https://linkinghub.elsevier.com/retrieve/pii/S0960894X19303142.

[12] Epel B, Maggio MC, Barth ED, Miller RC, Pelizzari CA, Krzykawska-Serda M, Sundramoorthy SV, Aydogan B, Weichselbaum RR, Tormyshev VM, and Halpern HJ (2019) Oxygen-Guided Radiation Therapy. International Journal of Radiation Oncology*Biology*Physics 103(4):977–984. ISSN 03603016. doi:10.1016/j.ijrobp.2018.10.041. URL https://linkinghub.elsevier.com/retrieve/pii/S0360301618339476.

[13] Kishimoto S, Matsumoto KI, Saito K, Enomoto A, Matsumoto S, Mitchell JB, Devasahayam N, and Krishna MC (2018) Pulsed Electron Paramagnetic Resonance Imaging: Applications in the Studies of Tumor Physiology. Antioxidants & Redox Signaling 28(15):1378–1393. ISSN 1523-0864, 1557-7716. doi:10.1089/ars.2017.7391. URL http://www.liebertpub.com/doi/10.1089/ars.2017.7391.

[14] Matsuo M, Kawai T, Kishimoto S, Saito K, Munasinghe J, Devasahayam N, Mitchell JB, and Krishna MC (2018) Co-imaging of the tumor oxygenation and metabolism using electron paramagnetic resonance imaging and 13-C hyperpolarized magnetic resonance imaging before and after irradiation. Oncotarget 9(38):25089–25100. ISSN 1949-2553. doi:10.18632/oncotarget.25317. URL https://www.oncotarget.com/lookup/doi/10.18632/oncotarget.25317.

[15] Epel B, Sundramoorthy SV, Krzykawska-Serda M, Maggio MC, Tseytlin M, Eaton GR, Eaton SS, Rosen GM, Kao JP, and Halpern HJ (2017) Imaging thiol redox status in murine tumors in vivo with rapid-scan electron paramagnetic resonance. Journal of Magnetic Resonance 276:31–36. ISSN 10907807. doi:10.1016/j.jmr.2016.12.015. URL https://linkinghub.elsevier.com/retrieve/pii/S1090780716302798.

[16] Khramtsov VV, Bobko AA, Tseytlin M, and Driesschaert B (2017) Exchange Phenomena in the Electron Paramagnetic Resonance Spectra of the Nitroxyl and Trityl Radicals: Multifunctional Spectroscopy and Imaging of Local Chemical Microenvironment. Analytical Chemistry 89(9):4758–4771. ISSN 0003-2700, 1520-6882. doi:10.1021/acs.analchem.6b03796. URL https://pubs.acs.org/doi/10.1021/acs.analchem.6b03796.

[17] Bobko AA, Eubank TD, Driesschaert B, Dhimitruka I, Evans J, Mohammad R, Tchekneva EE, Dikov MM, and Khramtsov VV (2017) Interstitial Inorganic Phosphate as a Tumor Microenvironment Marker for Tumor Progression. Scientific Reports 7(1):41233. ISSN 2045-2322. doi:10.1038/srep41233. URL https://www.nature.com/articles/srep41233.

[18] Bobko AA, Dhimitruka I, Zweier JL, and Khramtsov VV (2014) Fourier Transform EPR Spectroscopy of Trityl Radicals for Multifunctional Assessment of Chemical Microenvironment. Angewandte Chemie 126(10):2773–2776. ISSN 00448249. doi:10.1002/ange.201310841. URL https://onlinelibrary.wiley.com/doi/10.1002/ange.201310841.

[19] Bobko AA, Dhimitruka I, Zweier JL, and Khramtsov VV (2014) Inside Cover: Fourier Transform EPR Spectroscopy of Trityl Radicals for Multifunctional Assessment of Chemical Microenvironment (Angew. Chem. Int. Ed. 10/2014). Angewandte Chemie International Edition 53(10):2512–2512. ISSN 14337851. doi:10.1002/anie.201401080. URL https://onlinelibrary.wiley.com/doi/10.1002/anie.201401080.

[20] Bobko AA, Dhimitruka I, Komarov DA, and Khramtsov VV (2012) Dual-Function pH and Oxygen Phosphonated Trityl Probe. Analytical Chemistry 84(14):6054–6060. ISSN 0003-2700, 1520-6882. doi:10.1021/ac3008994. URL https://pubs.acs.org/doi/10.1021/ac3008994.

[21] Khramtsov VV and Zweier JL (2010) Functional in vivo EPR Spectroscopy and Imaging Using Nitroxide and Trityl Radicals. In Hicks RG, editor, *Stable Radicals*, 537–566. John Wiley & Sons, Ltd, Chichester, UK. ISBN 978-0-470-66697-5 978-0-470-77083-2. doi:10.1002/9780470666975.ch16. URL https://onlinelibrary.wiley.com/doi/10.1002/9780470666975.ch16.

[22] Komarov DA, Ichikawa Y, Yamamoto K, Stewart NJ, Matsumoto S, Yasui H, Kirilyuk IA, Khramtsov VV, Inanami O, and Hirata H (2018) In Vivo Extracellular pH Mapping of Tumors Using Electron Paramagnetic Resonance. Analytical Chemistry 90(23):13938–13945. ISSN 0003-2700, 1520-6882. doi:10.1021/acs.analchem.8b03328. URL https://pubs.acs.org/doi/10.1021/acs.analchem.8b03328.

[23] Yokoyama T, Taguchi A, Kubota H, Stewart NJ, Matsumoto S, Kirilyuk IA, and Hirata H (2019) Simultaneous T2* mapping of 14N- and 15N-labeled dicarboxy-PROXYLs using CW-EPR-based single-point imaging. Journal of Magnetic Resonance 305:122–130. ISSN 10907807. doi:10.1016/j.jmr.2019.06.012. URL https://linkinghub.elsevier.com/retrieve/pii/S1090780719301193.

[24] Yasui H, Kawai T, Matsumoto S, Saito K, Devasahayam N, Mitchell JB, Camphausen K, Inanami O, and Krishna MC (2017) Quantitative imaging of pO$_2$ in orthotopic murine gliomas: hypoxia correlates with resistance to radiation. Free Radical Research 51(9-10):861–871. ISSN 1071-5762, 1029-2470. doi:10.1080/10715762.2017.1388506. URL https://www.tandfonline.com/doi/full/10.1080/10715762.2017.1388506.

[25] Bártolo P, editor (2011) *Stereolithography: materials, processes and applications*. Springer, New York. ISBN 978-0-387-92903-3. OCLC: ocn305125487.

[26] Andrzejewska E (2001) Photopolymerization kinetics of multifunctional monomers. Progress in Polymer Science 26(4):605–665. ISSN 00796700. doi:10.1016/S0079-6700(01)00004-1. URL https://linkinghub.elsevier.com/retrieve/pii/S0079670001000041.

[27] Yin H, Ding Y, Zhai Y, Tan W, and Yin X (2018) Orthogonal programming of heterogeneous micro-mechano-environments and geometries in three-dimensional bio-stereolithography. Nature Communications 9(1):4096. ISSN 2041-1723. doi:10.1038/s41467-018-06685-1. URL https://www.nature.com/articles/s41467-018-06685-1.

[28] Kelly BE, Bhattacharya I, Heidari H, Shusteff M, Spadaccini CM, and Taylor HK (2019) Volumetric additive manufacturing via tomographic reconstruction. Science 363(6431):1075–1079. ISSN 0036-8075, 1095-9203. doi:10.1126/science.aau7114. URL https://www.science.org/doi/10.1126/science.aau7114.

[29] Gojzewski H, Guo Z, Grzelachowska W, Ridwan MG, Hempenius MA, Grijpma DW, and Vancso GJ (2020) Layer-by-Layer Printing of Photopolymers in 3D: How Weak is the Interface? ACS Applied Materials & Interfaces 12(7):8908–8914. ISSN 1944-8244, 1944-8252. doi:10.1021/acsami.9b22272. URL https://pubs.acs.org/doi/10.1021/acsami.9b22272.

[30] Halevy R, Tormyshev V, and Blank A (2010) Microimaging of Oxygen Concentration near Live Photosynthetic Cells by Electron Spin Resonance. Biophysical Journal 99(3):971–978. ISSN 00063495. doi:10.1016/j.bpj.2010.05.002. URL https://linkinghub.elsevier.com/retrieve/pii/S0006349510005643.

[31] Dendukuri D, Panda P, Haghgooie R, Kim JM, Hatton TA, and Doyle PS (2008) Modeling of Oxygen-Inhibited Free Radical Photopolymerization in a PDMS Microfluidic Device. Macromolecules 41(22):8547–8556. ISSN 0024-9297, 1520-5835. doi:10.1021/ma801219w. URL https://pubs.acs.org/doi/10.1021/ma801219w.

[32] Jeong HE and Suh KY (2008) On the role of oxygen in fabricating microfluidic channels with

ultraviolet curable materials. Lab on a Chip 8(11):1787. ISSN 1473-0197, 1473-0189. doi:10. 1039/b810348h. URL http://xlink.rsc.org/?DOI=b810348h.

[33] Alvankarian J and Majlis BY (2015) Exploiting the Oxygen Inhibitory Effect on UV Curing in Microfabrication: A Modified Lithography Technique. PLOS ONE 10(3):e0119658. ISSN 1932-6203. doi:10.1371/journal.pone.0119658. URL https://dx.plos.org/10.1371/journal. pone.0119658.

[34] Hashem M, Weiler-Sagie M, Kuppusamy P, Neufeld G, Neeman M, and Blank A (2015) Electron spin resonance microscopic imaging of oxygen concentration in cancer spheroids. Journal of Magnetic Resonance 256:77–85. ISSN 10907807. doi:10.1016/j.jmr.2015.04.012. URL https://linkinghub.elsevier.com/retrieve/pii/S1090780715001007.

[35] Levêque P, Leprince JG, Bebelman S, Devaux J, Leloup G, and Gallez B (2012) Spectral spatial electron paramagnetic resonance imaging as a tool to study photoactive dimethacrylate-based dental resins. Journal of Magnetic Resonance 220:45–53. ISSN 10907807. doi:10.1016/j.jmr. 2012.04.014. URL https://linkinghub.elsevier.com/retrieve/pii/S1090780712001723.

[36] Sanzhaeva U, Poncelet M, Tseytlin O, Tseytlin M, Gencheva M, Eubank TD, Khramtsov VV, and Driesschaert B (2020) Synthesis, Characterization, and Application of a Highly Hydrophilic Triarylmethyl Radical for Biomedical EPR. The Journal of Organic Chemistry 85(16):10388–10398. ISSN 0022-3263, 1520-6904. doi:10.1021/acs.joc.0c00557. URL https://pubs.acs.org/doi/ 10.1021/acs.joc.0c00557.

[37] Poncelet M, Driesschaert B, Bobko AA, and Khramtsov VV (2018) Triarylmethyl-based biradical as a superoxide probe. Free Radical Research 52(3):373–379. ISSN 1071-5762, 1029-2470. doi: 10.1080/10715762.2017.1369058. URL https://www.tandfonline.com/doi/full/10.1080/ 10715762.2017.1369058.

[38] Driesschaert B, Poncelet M, and Sanzhaeva U (2018) Design, Synthesis, and Characterization of New Triarylmethyl (TAM) Radicals for Biomedical EPR Applications. In *Milestone Presentations*. Snowbird, UT.

[39] Bobko AA, Eubank TD, Driesschaert B, and Khramtsov VV (2018) In Vivo EPR As-

sessment of pH, pO$_2$, Redox Status, and Concentrations of Phosphate and Glutathione in the Tumor Microenvironment. Journal of Visualized Experiments (133):56624. ISSN 1940-087X. doi:10.3791/56624. URL https://www.jove.com/t/56624/in-vivo-epr-assessment-ph-po2-redox-status-concentrations-phosphate.

[40] Biller JR, Tseitlin M, Mitchell DG, Yu Z, Buchanan LA, Elajaili H, Rosen GM, Kao JPY, Eaton SS, and Eaton GR (2015) Improved Sensitivity for Imaging Spin Trapped Hydroxyl Radical at 250 MHz. ChemPhysChem 16(3):528–531. ISSN 14394235. doi:10.1002/cphc.201402835. URL https://onlinelibrary.wiley.com/doi/10.1002/cphc.201402835.

[41] Mitchell DG, Tseitlin M, Quine RW, Meyer V, Newton ME, Schnegg A, George B, Eaton SS, and Eaton GR (2013) X-band rapid-scan EPR of samples with long electron spin relaxation times: a comparison of continuous wave, pulse and rapid-scan EPR. Molecular Physics 111(18-19):2664–2673. ISSN 0026-8976, 1362-3028. doi:10.1080/00268976.2013.792959. URL http://www.tandfonline.com/doi/abs/10.1080/00268976.2013.792959.

[42] Park HK, Shin M, Kim B, Park JW, and Lee H (2018) A visible light-curable yet visible wavelength-transparent resin for stereolithography 3D printing. NPG Asia Materials 10(4):82–89. ISSN 1884-4049, 1884-4057. doi:10.1038/s41427-018-0021-x. URL https://www.nature.com/articles/s41427-018-0021-x.

[43] Hou H, Khan N, Gohain S, Kuppusamy ML, and Kuppusamy P (2018) Pre-clinical evaluation of OxyChip for long-term EPR oximetry. Biomedical Microdevices 20(2):29. ISSN 1387-2176, 1572-8781. doi:10.1007/s10544-018-0272-x. URL http://link.springer.com/10.1007/s10544-018-0272-x.

[44] Frank J, Gündel D, Drescher S, Thews O, and Mäder K (2015) Injectable LiNc-BuO loaded microspheres as in vivo EPR oxygen sensors after co-implantation with tumor cells. Free Radical Biology and Medicine 89:741–749. ISSN 08915849. doi:10.1016/j.freeradbiomed.2015.10.401. URL https://linkinghub.elsevier.com/retrieve/pii/S0891584915010709.

[45] Kmiec MM, Tse D, Mast JM, Ahmad R, and Kuppusamy P (2019) Implantable microchip containing oxygen-sensing paramagnetic crystals for long-term, repeated, and multisite in vivo oximetry.

Biomedical Microdevices 21(3):71. ISSN 1387-2176, 1572-8781. doi:10.1007/s10544-019-0421-x. URL http://link.springer.com/10.1007/s10544-019-0421-x.

[46] Hou H, Khan N, and Kuppusamy P (2017) Measurement of pO2 in a Pre-clinical Model of Rabbit Tumor Using OxyChip, a Paramagnetic Oxygen Sensor. In Halpern HJ, LaManna JC, Harrison DK, and Epel B, editors, *Oxygen Transport to Tissue XXXIX*, volume 977, 313–318. Springer International Publishing, Cham. ISBN 978-3-319-55229-3 978-3-319-55231-6. doi:10.1007/978-3-319-55231-6_41. URL http://link.springer.com/10.1007/978-3-319-55231-6_41. Series Title: Advances in Experimental Medicine and Biology.

[47] Jarvis L, Williams B, PE S, EY C, CV A, H H, Schreiber W, Wood V, Flood A, Swartz H, and Kuppusamy P (2016) EPR Oximetry using OxyChip: Initial Results on Phase I Clinical Trial. 34. Krakow, Poland.

[48] Pandian RP, Dolgos M, Marginean C, Woodward PM, Hammel PC, Manoharan PT, and Kuppusamy P (2009) Molecular packing and magnetic properties of lithium naphthalocyanine crystals: hollow channels enabling permeability and paramagnetic sensitivity to molecular oxygen. Journal of Materials Chemistry 19(24):4138. ISSN 0959-9428, 1364-5501. doi:10.1039/b901886g. URL http://xlink.rsc.org/?DOI=b901886g.

[49] Tseytlin M (2017) Full cycle rapid scan EPR deconvolution algorithm. Journal of Magnetic Resonance 281:272–278. ISSN 10907807. doi:10.1016/j.jmr.2017.06.008. URL https://linkinghub.elsevier.com/retrieve/pii/S1090780717301635.

[50] Tseitlin M, Mitchell DG, Eaton SS, and Eaton GR (2012) Corrections for sinusoidal background and non-orthogonality of signal channels in sinusoidal rapid magnetic field scans. Journal of Magnetic Resonance 223:80–84. ISSN 10907807. doi:10.1016/j.jmr.2012.07.023. URL https://linkinghub.elsevier.com/retrieve/pii/S1090780712002650.

[51] Komarov DA and Hirata H (2017) Fast backprojection-based reconstruction of spectral-spatial EPR images from projections with the constant sweep of a magnetic field. Journal of Magnetic Resonance 281:44–50. ISSN 10907807. doi:10.1016/j.jmr.2017.05.005. URL https://linkinghub.elsevier.com/retrieve/pii/S1090780717301313.

[52] Nunes TG, Ceballos L, Osorio R, and Toledano M (2005) Spatially resolved photopolymerization kinetics and oxygen inhibition in dental adhesives. Biomaterials 26(14):1809–1817. ISSN 01429612. doi:10.1016/j.biomaterials.2004.06.012. URL https://linkinghub.elsevier.com/retrieve/pii/S0142961204005514.

[53] Pereira SG, Telo JP, and Nunes TG (2008) Towards a controlled photopolymerization of dental dimethacrylate monomers: EPR studies on effects of dilution, filler loading, storage and aging. Journal of Materials Science: Materials in Medicine 19(9):3135–3144. ISSN 0957-4530, 1573-4838. doi:10.1007/s10856-008-3434-1. URL http://link.springer.com/10.1007/s10856-008-3434-1.

[54] Naveed KuR, Wang L, Yu H, Ullah RS, Haroon M, Fahad S, Li J, Elshaarani T, Khan RU, and Nazir A (2018) Recent progress in the electron paramagnetic resonance study of polymers. Polymer Chemistry 9(24):3306–3335. ISSN 1759-9954, 1759-9962. doi:10.1039/C8PY00689J. URL http://xlink.rsc.org/?DOI=C8PY00689J.

## 3.9 Figures
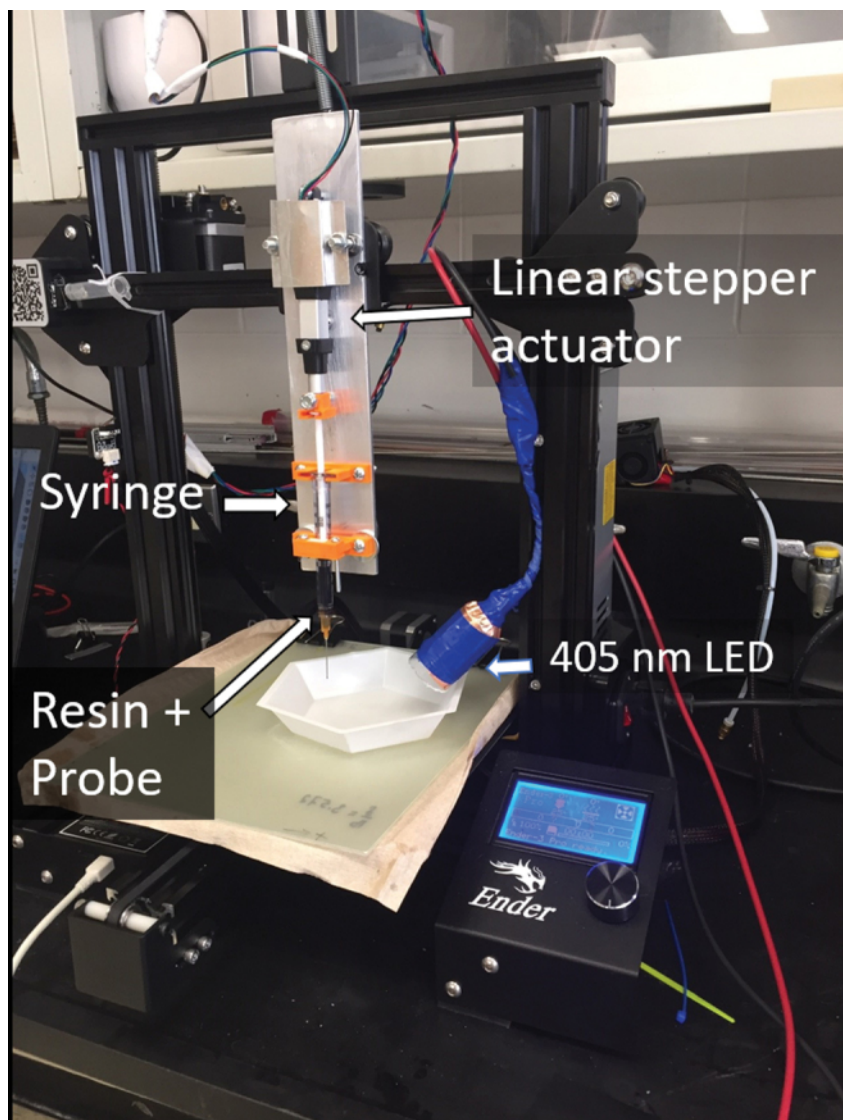


Figure 3.1: Modified Ender-3 Pro 3D Printer. A stepper motor responsible for plastic filament extrusion was replaced by a linear actuator. The linear actuator, connected to a syringe plunger, permits both extrusion and retraction of a mixture of Clear FLGP CL04 resin and LiNc-BuO crystals. A 385-410 nm light source is attached to the printer frame that initiates photopolymerization of the resin. 3D, three-dimensional; LiNc-BuO, lithium octa-n-butoxynaphthalocyanine. Color images are available online.
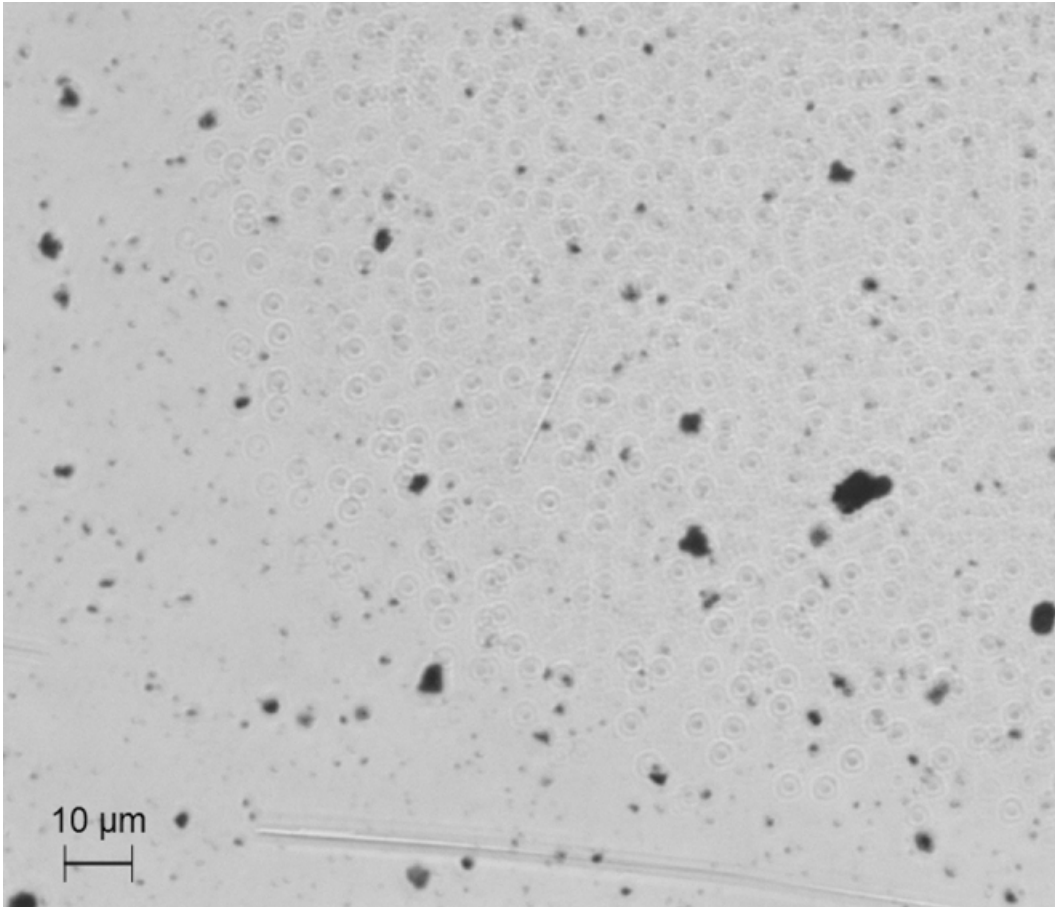
Figure 3.2: Microscopic images of LiNc-BuO crystals dispersed in Formlabs "Clear" resin.
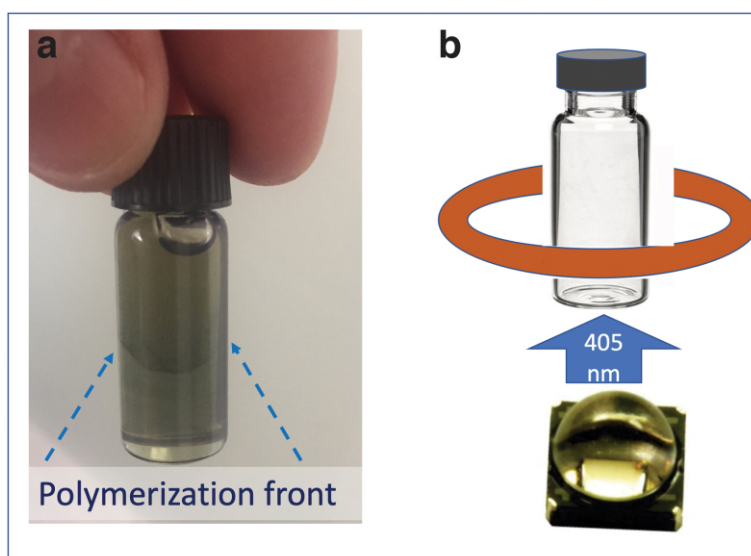
Figure 3.3: A bottle filled with a mixture of oxygen-sensitive EPR probe LiNc-BuO and "Clear" resin. **(a)** The photo of the bottle shows the suspension of LiNc-BuO in the resin. After the light exposure, a visible liquid-solid interface is formed. **(b)** Sketch showing the sample containing a bottle placed in the resonator loop (*red*) and LED beneath the bottle. Polymerization propagates from the bottom up. EPR, electron paramagnetic resonance; LED, light-emitting diode. Color images are available online.
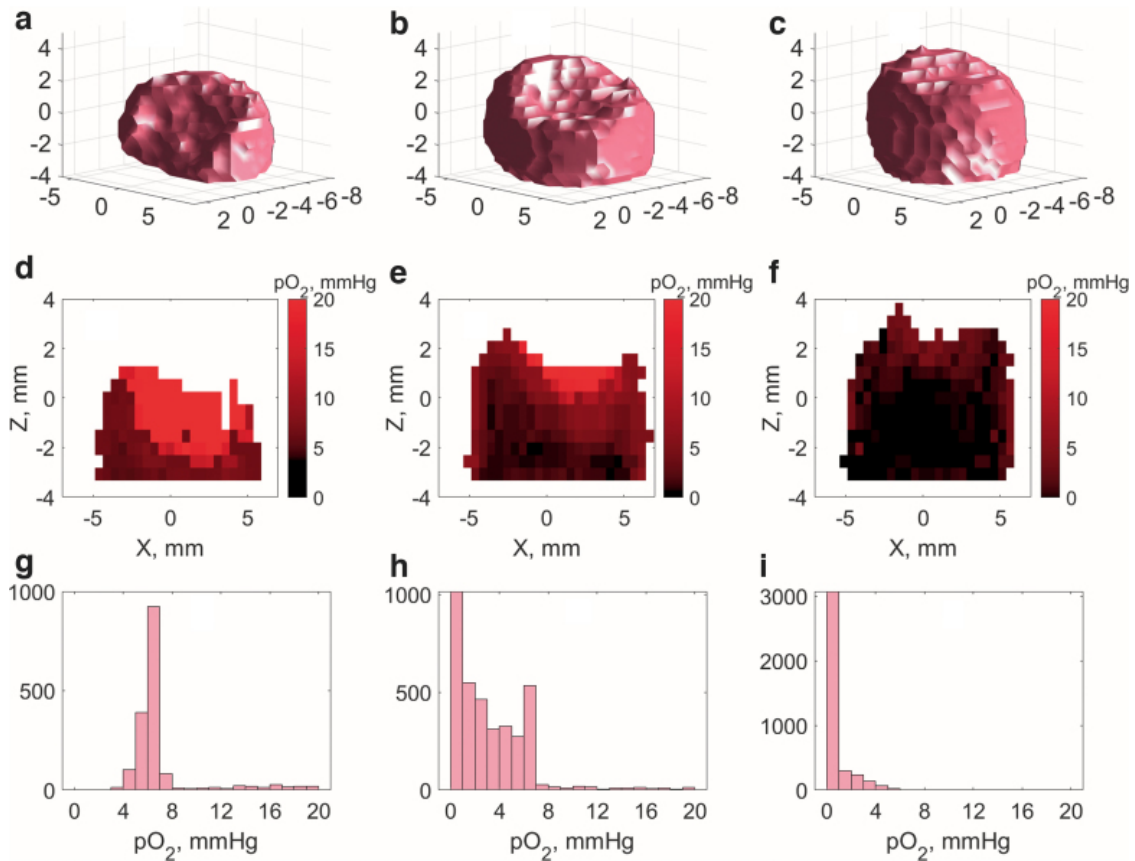
Figure 3.4: Spin probe concentration and oxygen partial pressure ($pO_2$) data obtained from three 4D EPR images acquired after 3 (**a, d, g**); 7 (**b, e, h**); and 82 (**c, f, i**) seconds of light exposure. Isosurfaces of the concentration maps are shown in **a-c** subplots. 2D coronal slices of oxygen distribution (**d, e, f**) and whole image histograms (**g, h, i**) are provided for the 3-, 7-, and 82-s images, respectively. Bin size in all histograms is 1 mmHg. 2D, two-dimensional. Color images are available online.
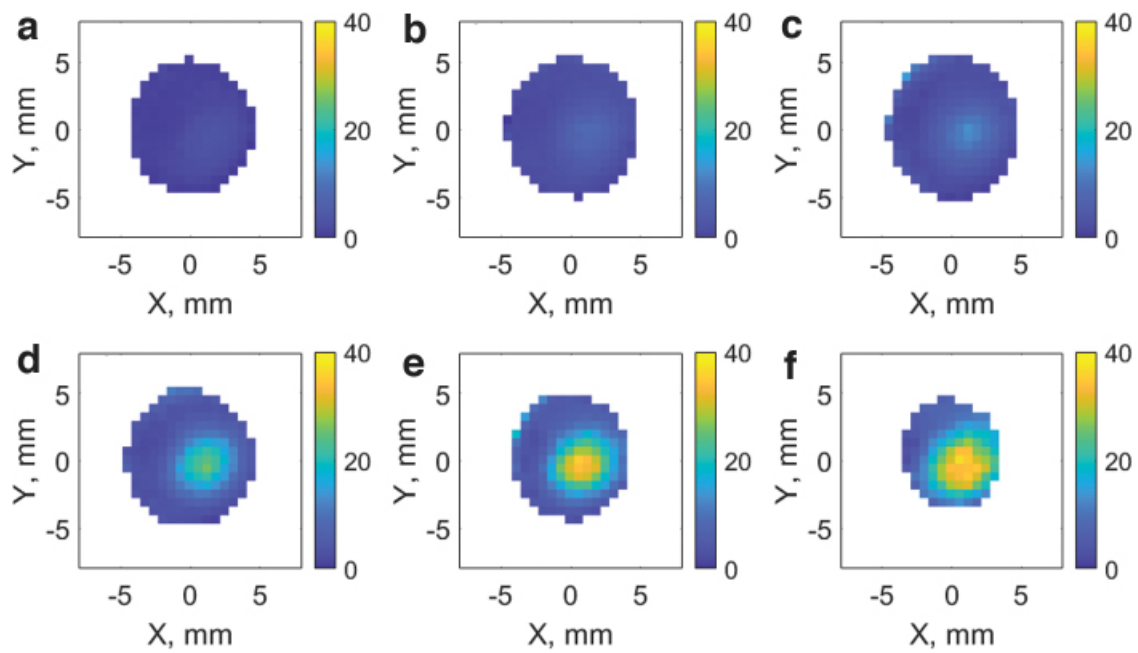
Figure 3.5: Change in oxygen partial pressure in the units of mmHg across a photo-polymerized cylinder as the result of the curing process. Slices **a-f** correspond to the z-depth of 159, 95, 32 ,-32, -95, -195 μm, respectively. Color images are available online.
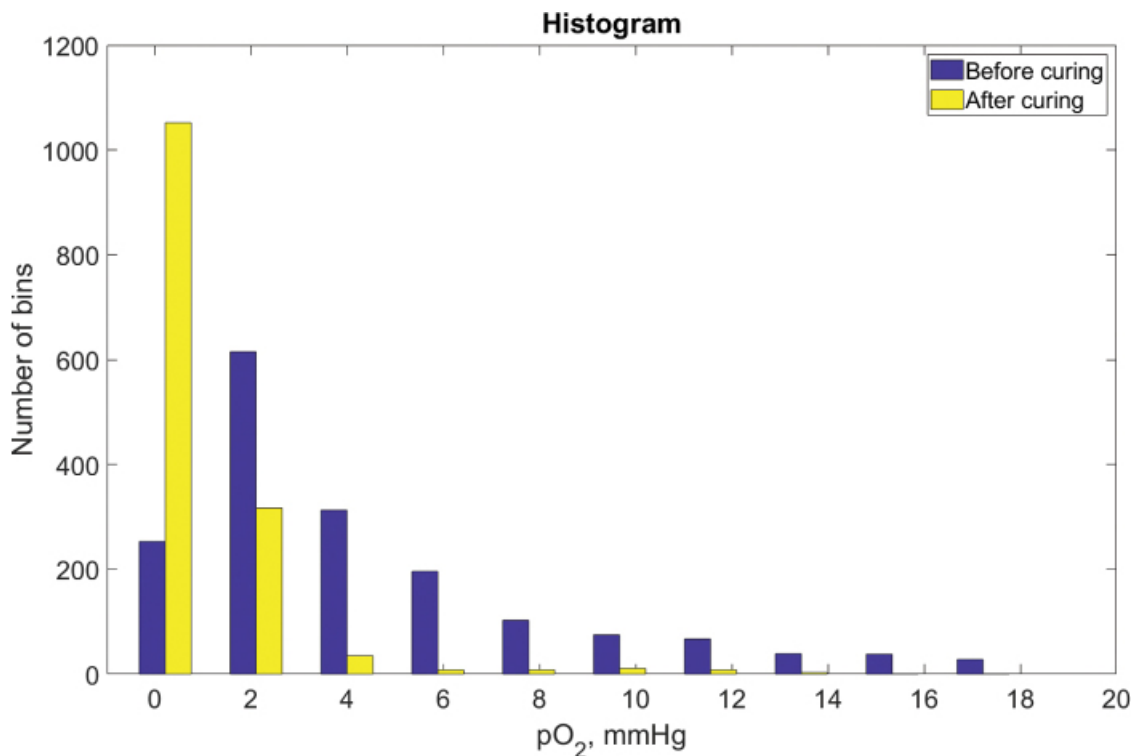
Figure 3.6: Histograms of $pO_2$ distribution in the polymerized cylinder before and after postcuring. The number of voxels in the precuring and postcuring oxygen images are 1727 and 1442, respectively. The difference in the numbers is the result of the thresholding procedure that removes voxels with low signal-to-noise ratios. The voxel volume is approximately 3-4 $mm^3$. Color images are available online.
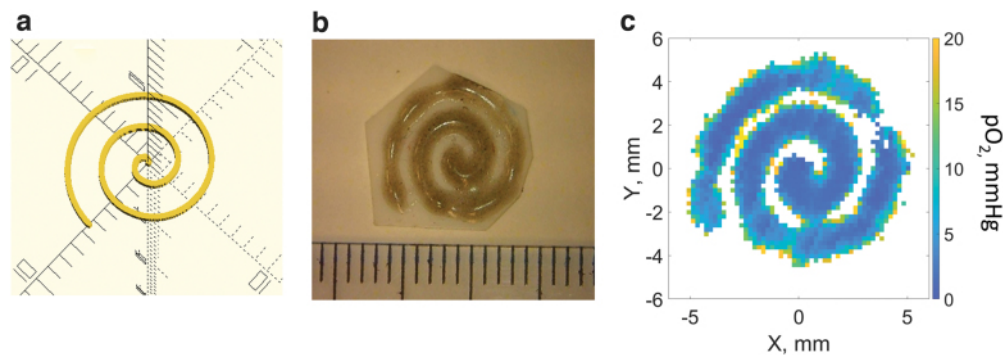


Figure 3.7: OpenSCAD spiral model **(a)**, photo of a 3D printed part (marks on the ruler are in millimeters) **(b)**, 2D slice from 3D image of oxygen distribution in the spiral **(c)**. Color images are available online.

# Chapter 4

# Rapid scan EPR mapping of oxygen flux in a cell-laden bioprinted model using Saccharomyces cerevisiae and murine macrophages

**Authors:** Ryan C. O'Connell[1,2], Sajad Sarvari[1], Duncan McGee[3], Vignesh Sivashankar[3], Oxana Tseytlin[1,2], Tim Eubank[2,4], Ethan Meadows [5,6], Elena Pugacheva[1,7,8], Benoit Driesschaert[2], John M Hollander [5,6], Mark Tseytlin[1,2,7]

**Affiliations:**

[1] Department of Biochemistry and Molecular Medicine, West Virginia University, West Virginia, USA

[2] In Vivo Multifunctional Magnetic Resonance Center at Robert C. Byrd Health Sciences Center, West Virginia University, Morgantown, West Virginia, USA

[3] Statley College of Engineering and Mineral Resources, West Virginia University, Morgantown, West Virginia, USA

[4] Department of Microbiology, Immunology, and Cell Biology, West Virginia University, Morgantown, West Virginia, USA

[5] Department of Human Performance, West Virginia University, Morgantown, West Virginia, USA

[6] Mitochondria and Bioenergetics Working Group, West Virginia University, Morgantown, West Virginia,

USA

[7] West Virginia University Cancer Institute, Morgantown West Virginia, USA

[8] Department of Radiation Oncology, West Virginia University, Morgantown, West Virginia, USA

**Note:** This chapter includes text and figures planned for submission for publication to a journal.

## 4.1 Abstract

Bioprinting and other 3D culture techniques promise to develop complex *in vitro* tissue models to answer medical and biological questions. An unsolved question in this field is the amount of oxygen delivered to cells in thick bioprints, due to limited diffusion. Several methods such as optical fiber optic probes and microscopy can be used for investigating standard 2D cultures, but these methods give limited spatial information or insufficient depth penetration in 3D culture. Electron paramagentic resonance imaging (EPRI) is proposed to map changing oxygen levels in these models. Here, a spherical simulation of oxygen consumption is generated using the heat equation. To give basis for comparison, fiber optic probe and Seahorse measurements of oxygen consumption rate (OCR) are taken of *Saccharomyces cerevisiae* yeast using varying concentrations of a nutrient source, sodium pyruvate. Measurements are also taken with peritoneum-derived murine macrophages. Using a home-built bioprinter, 3D cell culture models containing yeast or macrophages are printed and imaged with EPRI. OCR rates are calculated by analyzing consecutive EPR images in the time dimension. In all OCR measurement methods with yeast, a difference in OCR is evident depending on the presence of sodium pyruvate. Taken as a whole, this research is a first demonstration of mapping 3D OCR.

## 4.2 Introduction

Bioprinting is an exciting and rapidly developing direction for additive manufacturing, where a 3D model is constructed through the 3D printing of a cell-laden hydrogel called bioink.[1] This technology has the capability of constructing whole organs with high resolution (1-10μm).[2,3] There are promising developments in the application of bioprinting towards transplants/implants, drug testing and cancer models.[4–6] To attain more physiologically-relevant models, the bioprinting field is moving towards more complex, functional, and physiologically-sized organ models.

One of the limiting factors in the development of physiologically-sized organ models is the limit of diffusion for oxygen, nutrients, and waste.[7,8] The term *limit of diffusion* refers to the depth oxygen can reach before being consumed by cells in a natural or artificial tissue. In bioprinting, depending on the cell density and type of materials used, the diffusion limit is between 40μm for scaffold-free bioprints to 350μm for acellular structures.[9,10] Beyond this point, cell viability decreases as the delivery of oxygen,

nutrients, and waste becomes insufficient to support the cells.[7]

To overcome this limitation, two main strategies are employed. The first strategy is supplying oxygen to cells by adding oxygen-releasing substances to the bioink. Calcium peroxide or sodium percarbonate show steady release of oxygen over several days, although there are the drawbacks of the release of hydrogen peroxide and a spike of oxygen release directly following printing, potentially damaging the printed cells.[11] Catalase can be added to break down the hydrogen peroxide into water and oxygen.[12] Another method to supply oxygen to cells in bioprints is by printing with algae and cyanobacteria, which release oxygen as a product of photosynthesis.[13] This method has been used in conjunction with HepG2 hepatic tumor cells to demonstrate steady release of oxygen and increased cell viability.[14]

Similar to natural tissue, bioprinted models can supply oxygen and nutrients and remove cell waste through vasculature. In the body, there is a spectrum of vasculature size, ranging from 2-5mm for arteries to 5-20µm for capillaries. In bioprinting, the smallest vascular size should be less than the limit of diffusion (40-350µm) to allow for dense functional models.[15] However, most cell-laden vascular models are larger than 500µm, with the smallest cell-laden vascular bioprint to date being 150µm in diameter.[16] Acellular capillary bioprints with diameter 5µm have been printed, introducing the possibility for cell-laden bioprinting of capillaries of this size.[17]

Cell viability and single-point oxygen measurements have been used to gauge the success of these methods for providing oxygen and nutrients to the cells.[7,18–20] Increase in cell viability shows that the cells have sufficient oxygen and nutrient delivery to survive, but does not indicate if oxygen and nutrient delivery have reached physiologic levels. Similarly, single-point oxygen measurements have been used to demonstrate the success of microchannels and perfusion in increasing oxygen concentrations in the interior of bioprints at a single point, but do not information about the overall oxygen concentration throughout the bioprints.

Several imaging modalities provide maps of oxygenation through tissue. Infrared fluorescent microscopy can quantitatively map oxygen, among many other parameters, with high resolution ($\sim$200µm) and up to 3mm depth.[21] The depth limitations of microscopy limit its capability for thicker bioprints such as whole organ models. Positron emission tomography (PET) can also be used to measure oxygen with positronium or $^{15}$O radiotracers, along with glucose consumption rates using Fleuro-2-deoxy-D-glucose, a glucose analog.[22–24] PET is however limited in low spatial and oxygen resolutions. Doppler-ultrasound can be used to image perfusion rate with good resolution.[25] MRI can be used to measure relative or

quantitative oxygen concentration in certain conditions.[26,27] Finally, electron paramagnetic resonance imaging (EPRI) has been used to measure quantitative oxygen concentration and partial pressure *in vivo*.[28–30]

EPRI has the advantage of deep tissue imaging ($\sim$10 cm), multifunctional probes, and fast imaging times (rapid scan EPRI, RS EPRI). Because EPRI probes directly measure oxygen, pH, and other parameters, EPRI is a quantitative technique for mapping the microenvironment (ME).[31] Through the use of a new soluble probe, the probe for *in vivo* concurrent measurement of p**H**, p**O**$_2$ and **P**i in **E**xtracellular space (HOPE), multiple parameters of the ME relevant to bioprinting can be mapped.[32] Other probes, like deuterated Finland trityl (dFT) and its derivatives (Ox063 and Ox071) provide oxygen mapping with good resolution.[33]

Quantitative mapping and determination of oxygen flux in bioprinted models of *Saccharomyces cerevisiae* is demonstrated here. These measurements are made possible through the development of stable, automated digital tuning, coupling, and data acquisition on our home-built modular RS EPRI system, making long-term imaging experiments for determining oxygen flux possible.[34–36] The incorporation of EPR probe into the bioink enables long-term oxygen mapping.

## 4.3   Materials and Methods

### 4.3.1   Bioprinter design

The bioprinter design was based on a previous modified 3D printer design for plastic resin (figure 4.1)[35] using the **f**reeform **r**eversible **e**mbedding of **s**uspended **h**ydrogels (FRESH) bioprinting method.[37] A Creality Ender 3 Pro 3D printer was modified by the replacement of the heated extrusion head with a linear extruder. This extruder had the capability to print bioink by pushing the plunger of a syringe.

### 4.3.2   FRESH 2.0 synthesis

FRESH 2.0 support bath was made using an existing protocol. Briefly, 2% gelatin type B, 0.25% Plurinic F-127, 0.1% gum arabic, 50% ethanol and the remainder water, were mixed together at 45°C until dissolved. This solution was then taken off heat and stirred at 950 RPM for 16 hours. After 16 hours of stirring, the solution was aliquoted into 50mL centrifuge tubes and centrifuged at 300 RCF for 5 minutes. The supernatant was removed, and a 10mM CaCl$_2$ solution was added to fill the centrifuge

tubes to 45mL. This wash procedure was repeated at least three times, with centrifugation at 1000 RCF. The resulting FRESH solution was stored at 4°C until use, at which time the FRESH solution was centrifuged at 1000 RCF, supernatant removed, and FRESH beads moved to the printing chamber.

### 4.3.3 Bioprinting protocol

The bioprinted model was designed using MATLAB, resulting in *Gcode*, the protocol the bioprinter uses to print the model (see Appendix for code). A tube was generated with outer radius 7.5mm, inner radius 3mm and height 8mm. A representative CAD model was displayed in figure 4.2. Bioprinting was conducted in a biosafety cabinet to maintain sterile conditions. An alternative to placing the entire bioprinter in the biosafety cabinet would be to construct a biosafety cabinet around the bioprinter. Custom biosafety cabinets are standard in commercial bioprinters.

The bioink was prepared as follows: for the experiments using *Saccharomyces cerevisiae*, cells were diluted to 0.25 optical density (O.D.) using a UV-vis spectrometer at 600nm light. 500μL of this solution was centrifuged at 4000 RCF for 5 minutes, and supernatant discarded. 750μL of 6% sodium alginate was added, resulting in a final alginate concentration of 4.5%. EPR oxygen-sensitive probe (dFT) was added to give final concentration of 1mM. After bioprinting, the resulting dFT concentration in the bioink was ~200μM due to incorporation of FRESH bath into the bioprint and diffusion of EPR probe into the support bath after printing. The final dFT concentration of 200μM dFT was calculated by dissolving a bioprinted model and comparing UV-Vis absorption at 467nm against standardized solutions of dFT. Sodium pyruvate was added in figure 5b to give final pyruvate concentration of 1%. Sodium pyruvate was also added to the FRESH bath in this case, resulting in FRESH bath pyruvate concentration of 1%. Finally, water was added to make 1mL total volume. For the macrophage experiment, 25,000 cells were combined with 750μL of 6% sodium alginate dissolved in calcium-free EMEM medium with added FBS, sodium pyruvate, and HEPE buffer. 1mM Ox063 and more culture media were added to make 1mL total volume of bioink. Final concentration of Ox063 in the bioprint was approximately 200μM, calculated with absorbance of the dissolved bioprint. After printing finishes, the print was allowed to rest in the FRESH bath for a period of 6 hours to allow strong crosslinking.

### 4.3.4 Cell culture

*Saccharomyces cerevisiae* strain RM11$\alpha$ yeast are cultured on YPD agar in standard 100mm cell culture plates at 4°C. Yeast were chosen to demonstrate OCR mainly because of their ease-of-use. Yeast can be grown on the countertop or refrigerator at room temperature using basic yeast agar. 24hrs before the experiment, yeast were inoculated into 10mL of a custom YPGal liquid media consisting 2% peptone, 1% yeast extract and 2% galactose to upregulate the aerobic respiration pathway. The media containing yeast was stirred on a hot plate set to 30°C and 350 RPM.

MDA-MB-231 cells were cultured in L-15 medium supplemented with 2mM glutamine, 15% fetal bovine serum and 1% penecillin/streptomycin antibiotic solution, and kept in a cell incubator at 37°C, 5% $CO_2$. After bioprinting and imaging, the bioprint containing MDA-MB-231 cells was kept in L-15 medium in the cell incubator for 24 hours.

### 4.3.5 Oxygen consumption measurements

Yeast were collected from liquid culture by centrifugation at 1000 RCF and washed twice in distilled water. Yeast were starved by mixing in distilled water for six hours before the experiment. The yeast were diluted to an OD (650 nm) of 1.00 using a UV-Vis spectrometer. This correlates to roughly $7 \times 10^7$ cells per mL. Cells were additionally counted using a hemocytometer.

To measure OCR in the presence of varying concentrations of pyruvate under stirring conditions (stirring removes the factor of diffusion of oxygen and pyruvate through undisturbed solution), 20mL cells (see Appendix for cell concentrations) in solution consisting of one of the following concentrations of sodium pyruvate (1%, 0.4%, 0.2% and 0.04%) were placed in a 25mL Erlenmeyer flask. 4mL viscous mineral oil was added to the surface of the yeast solution to slow oxygen diffusion from the external environment. The solution was stirred vigorously using a stir bar on a hot plate at 500 RPM with the heating element turned off. Measurements (4 technical replicates per group) were collected using a fiber-optic oxygen sensor. Data was analyzed using MATLAB software. Groups were analyzed for statistical differences using a 1-way ANOVA and pairwise comparisons test.

To measure OCR in the presence of varying concentrations of pyruvate in undisturbed conditions (stirring only between technical replicates), an Agilent XFe24 Seahorse Analyzer was used. Cells were centrifuged and adhered to the 24 well plate. The plates were filled with 500µL of water. For yeast

measurements, injection of 50μL sodium pyruvate solution was used give final well concentrations of 1%, 0.4%, 0.2%, and 0.04% pyruvate. Measurements were collected for 10, 15 or 60 minutes, with mixing of the media for 2 minutes between measurements.

As a control for EPRI OCR measurement of macrophages, it was necessary to compare the OCR of macrophages to that of yeast to determine the concentration of macrophages to use in the bioprint. For peritoneal macrophages no injections of pyruvate were used. Rather, macrophages (25,000, 50,000, 100,000 or 250,000 macrophages per well) were in a dMEM media without phenol red during the Seahorse experiment.

### 4.3.6 Peritoneal macrophage isolation

1 mL of sterile 3% Brewer thioglycolate medium was injected into the peritoneal cavity of a C57BL/6J mouse. After four days, the mouse was euthanized via isoflurane overdose with secondary euthanasia method of opening the pleural cavity. The skin around the incision area was sterilized with 70% isopropanol solution. Next, the outer skin of the peritoneum was cut to expose the inner skin lining the peritoneal cavity. 5 mL ice cold PBS (with 3% FCS) was injected into the peritoneal cavity using a 27G needle, and the peritoneum was massaged to dislodge any attached cells. Then, a 25G needle was inserted to aspirate the fluid containing the macrophages. The resulting cell suspension was centrifuged at 132 RCF (800 RPM on our machine) for 10 minutes at 4°C. A live/dead cell count was conducted with Trypan blue on a hemocytometer.

### 4.3.7 RS EPRI setup

A previously described RS EPRI system was used.[34,36] Briefly, the the system was a modular RS EPR imager at 800MHz with automatic control of tuning, coupling, and data acquisition. Home-built MAT-LAB software was also used (see Appendix). Images were collected with 2000 averages per spectrum, internal averaging conducted by the digitizer. Bioprints were placed inside of a 14mm inner diameter cylindrical vial, with the outside edges of the bioprint in contact with the vial. 500μL petroleum jelly was melted at 50°C and added to remove as much air as possible in the center and top of the tube bioprint. The rest of the volume in the vial was filled with dental impression, which cures over a period of 3-5 minutes. The vial containing the sealed bioprint was placed in the resonator. The vial was

centered in the RS EPRI system by changing gradients in x, y, z directions and observing the spectrum. Finally, images were collected with 26 blocks containing 126 projections and a zero gradient spectrum. To capture faster changes in kinetics, where large changes in concentration occur within 30 minutes, a slightly different collection method was used, where 16 blocks of 126 projections are collected. The last 10 blocks were not collected in these images, reducing the amount of time required for each image. In the last image of the experiment, all 26 blocks of projections were collected. The last 10 blocks of projections were mostly responsible for the collection of spatial information rather than the collection of spectral information. During image reconstruction, this spatial data is assumed to remain constant in the bioprinted model. This method of image collection allows for $\sim 62\%$ faster data collection, enabling the measurement of faster kinetics.

## 4.4 Results

### 4.4.1 Simulation of oxygen diffusion and consumption by cells in 3D culture

Oxygen flux is dependent on several factors, including diffusion coefficient, cell size, cell number, and consumption rate by the cells. To investigate the role of each of these factors towards oxygen flux in a bioprint, a simulated model was generated (figure 4.3). Oxygen diffusion and consumption in a homogeneous bioprint where cells are equidistant in a spherical coordinate system can be described using the following heat equation:

$$\frac{\partial u}{\partial t} = \frac{D(x)}{x^2} \frac{\partial}{\partial x} \left( x^2 \frac{\partial u}{\partial x} \right) + f(t) \tag{4.1}$$

The concentration of oxygen is given as u, time as t, distance x. $D(x)$ is diffusion coefficient, where the first three values represent the diffusion coefficient of the plasma membrane and/or cell wall, and the rest of the values represent the diffusion coefficient of the extracellular matrix. Finally, $f(t)$ is the consumption of oxygen by the cells. Based on experimental data (see mixing raw data in Appendix), $f(t)$ is a linear equation unless limited by diffusion (there are no more oxygen molecules in the sphere $x(1)$ to consume).

This theoretical model of oxygen flux was simulated numerically in MATLAB (see Appendix for code). Simulations compare the effects of internal oxygen consumption rate (OCR) on oxygen concentration

over time (figure 4.4 a, b), external OCR measured by the viewer (figure 4.4 c, d), cell concentration (compare figure 4.4 a to b and c to d), and the number, N, of increments in x, the distance from the center of the cell to the halfway point to the next cell. Internal OCR is the amount of oxygen consumed by the cell per second, and is represented by $f(x)$ in equation 4.1. External OCR is the OCR rate that would be measured by an observer, represented by $\frac{du}{dt}$. As expected, internal OCR positively correlates with external OCR and cell concentration. External OCR was also linear, with curvature only near 0 mM $O_2$. However, there was a large difference ($\sim 50\%$) in external OCR with respect to N.

To match the simulation more closely with EPRI results, an extended simulation was completed over the period of a week with lower internal OCR and N=1000 (see figure 4.5). Two hours of the simulation was completed over the period of a week (solid lines). The simulation was extrapolated for a period of 20 hours (dotted lines).

### 4.4.2 Yeast metabolic measurements in 2D culture

Yeast oxygen consumption rates under varying concentrations of sodium pyruvate were measured in an Erlenmeyer flask under mixing conditions to minimize the contribution of oxygen diffusion (figure 4.6). Starved yeast were placed in distilled water containing one of the following concentrations of sodium pyruvate: 1%, 0.4%, 0.2%, or 0.04% by mass. Oxygen concentration in the yeast solution was measured over time (see Appendix). The rate was extracted from the data using MATLAB, and adjusted by the amount of cells in solution. As expected, the rate of yeast oxygen consumption increased as pyruvate concentration increased (figure 4.7). Statistics are shown in Tables 4.1 and 4.2.

To confirm yeast oxygen consumption rates, additional measurements were collected using Seahorse. In this case, the yeast were adhered to the bottom of a 24-well plate. A solution of sodium pyruvate in distilled water was mixed by the machine, then the oxygen probe moved in close proximity to the cells, and oxygen concentration over time was measured without mixing. Measurements were collected for 10 minutes (figures 4.8 and 4.9, tables 4.1 and 4.2) and 60 minutes (figures 4.10 and 4.11, tables 4.3 and 4.4). The first minute of data collection was used to generate OCR and statistical analyses. Oxygen consumption increased with increasing pyruvate concentrations, matching with data in figure 4.7.

### 4.4.3   3D bioprinting of a cell-laden model

Loss in cell viability is a common issue among extrusion bioprinters.[38] To demonstrate the ability for our homebuilt bioprinter (figure 4.1) to construct cell-laden constructs with live cells, MDA-MB-231 GFP and RFP positive cells were printed into a thick tube construct (figure 4.2). These cells were chosen to demonstrate bioprinting with human cells in the homebuilt bioprinter and for their ability to switch between aerobic and anaerobic respiration. The cells were imaged 6 and 24 hours after printing, seen in figure 4.12 a and b, respectively, using an Olympus spinning disc confocal microscope. While signs of apoptosis were present in some cells, most cells remained viable. Because the confocal microscope cannot penetrate the full depth of the bioprint without sectioning, and because we only wanted to see if some cells survived the printing process, we did not numerically measure cell viability.

### 4.4.4   Mapping oxygen diffusion in 3D bioprinted culture

To better understand diffusion in 3D culture, an acellular thick tube construct was printed. Alginate bioink containing 1mM dFT was used. After printing, dFT diffused from the print into the FRESH support bath, giving a final probe concentration of $< 200\mu M$ in the print. A solution of Pluronic F-125 bioink containing glucose and GOX was injected into the construct, consuming oxygen. In this case, the diffusion of oxygen out of the print towards the GOX was investigated. The print was then sealed from oxygen using 500μL petroleum jelly and dental impression mold. Figure 4.13 maps change in oxygen partial pressure as a function of time. At time 0, a sharp gradient is visible in the print. Oxygen concentration decreases to near-zero, then rises slightly after two hours, perhaps as a result of small air bubbles around the perimeter of the construct.

### 4.4.5   Mapping oxygen flux in 3D bioprinted culture with Saccharomyces cerevisiae

As a proof-of-principle demonstration, two thick tube constructs laden with yeast was printed. One construct contained sodium pyruvate at 1% concentration. 1% pyruvate was because it maintained a similar nutrient density as the media the yeast were grown in. The second construct did not contain sodium pyruvate. Pyruvate was used as an energy source for the yeast to force aerobic respiration. To make the constructs, yeast diluted to 0.25 optical density were further diluted twice more for a resulting concentration of $3 * 10^6$ cells per mL. Both bioprints were continuously imaged over several hours.

Combining the images together in the time dimension allows for analysis of kinetics. Both yeast only (figure 4.14 and yeast + pyruvate in figure 4.15 show decrease in oxygen concentration over time. OCR was calculated for each voxel in the time dimension (figure 4.16 and d, e of figures 4.14 and 4.15). By comparing average OCR in 3D culture, it was possible to deduce that yeast in the bioprint containing 1% sodium pyruvate consumed more oxygen over time. Finally, OCR was calculated for eight trials using EPR spectrometry (figure 4.17). Linear OCR rate was calculated by measuring EPR spectra over time in bioprints containing *Saccharomyces cerevisiae* with and without pyruvate (n=4 in each group). Student's t-test showed a difference between pyruvate and no pyruvate groups (p=0.0478).

### 4.4.6 Comparison of OCR measurement methods with Saccharomyces cerevisiae

Each OCR measurement method is compared in table 4.7. Additionally, a one-way ANOVA and Tukey HSD Post-hoc test was performed on the summarized data (tables 4.8 and 4.9). Importantly, Seahorse and EPR spectral measurements were similar, while EPRI measurements more closely matched the Stirring experimental results. Most methods were found to be different from each other, resulting in slightly different OCR for the given conditions, though all methods were in the same magnitude of each other. These differences could be a result of the different environment (e.g. stirring or mixed environment compared to not-stirred environment), or due to error. For example, at high gradients required for EPRI spatial resolution, the measured data become noisy and the potential for error increases.

### 4.4.7 Murine macrophage metabolic measurements in 2D culture

As a comparison with yeast, and moving towards mammalian cells, OCR of murine macrophages was measured using Seahorse in figure 4.18. The number of cells per well was varied between 25,000 and 250,000 macrophages. Similar to the yeast Seahorse data, OCR was not linear over longer periods of time. The first minute of data collection was used to generate OCR and statistical analyses.

### 4.4.8 Mapping oxygen flux in 3D bioprinted culture with murine macrophages

To demonstrate oxygen consumption by mammalian cells in 3D culture, a cell-laden bioprint containing 25,000 macrophages per mL was generated. Peritoneal-derived macrophages were chosen because they can switch from aerobic to anaerobic respiration, do not proliferate, and can be polarized. These

aspects of macrophages are planned for further investigation in bioprinting in our lab. EPR imaging was conducted over 18 hours (figure 4.20. Over this time period, oxygen partial pressure inside the construct decreased from levels comparable to air partial pressure (159mmHg) to below 50mmHg at the end of imaging. By combining the images, OCR was extracted from the data (figure 4.20 e). OCR for peritoneal macrophages was less than the yeast models, but the number of cells were also much less.

## 4.5  Discussion

This research is presented as a first demonstration of OCR mapping in 3D culture. To achieve this, a spherical model of oxygen diffusion and consumption by cells was generated to explore the relationships between cell concentration, diffusion coefficient at the surface of the cell, diffusion coefficient of the extracellular matrix, and oxygen consumption rate inside the cell. Next, to give a basis of comparison for 3D OCR, and example rates for the spherical model, OCR was calculated using Seahorse as well as in a stirred solution using a fiber optic probe. Using a home-built bioprinter, cell-laden and acellular 3D constructs were printed with 4.5% alginate. Viability of cells was confirmed by printing fluorescent-labeled cancer cells. RS EPR imaging was conducted over several hours for bioprints containing *Saccharomyces cerevisiae* yeast or macrophages. Images were combined together to add the dimension of time, resulting in 5D spectral-spatial-temporal imaging. Comparing changes in the oxygen maps over time, 3D OCR was able to be extracted. Finally, the 3D OCR was compared in bioprinted yeast with and without pyruvate.

Measuring OCR and mapping oxygen in bioprinted models will be important as the size and complexity of models increases. Already, researchers are including vasculature into bioprints to allow cells to survive in thicker models.[7,39,40] The need to measure the success of the methods to deliver oxygen to cells deep in the model remains. This research represents the first efforts towards this goal.

A model of spherical diffusion was generated using MATLAB, comparing the effects of internal oxygen consumption rate, cell concentration, and spherical shell incremental size, N. Increase in observed OCR was expected for increasing internal OCR and cell concentration. However, changes in N also resulted in large changes in observed OCR, indicating that the value N needs to be increased until it no longer plays a role in observed OCR. The simulation works by calculating oxygen concentration in each shell, $u(x)$ at each time point, $t$. The number of iterations required for calculating $u(x)$ for $N = 400$ and

135

1e6 cells over a simulation time period of one second is 150,470 iterations, from the equation $i = \frac{D}{dx^2}$, where $i$ is iterations, $D$ is diffusion coefficient, and $dx$ is the shell size increment determined from N and the cell concentration. For N of 1000, the number of iterations required increases exponentially to 943,290, plus the compounding factor of computing more points. Therefore, while the simulation takes 6 hours to run on our computers using $N = 200$, it will take several days, months, or years to run simulations with N greater than 400.

For N of 1000, 5,000 or 10,000 the time required is approximately 9 days, 162,700 days and ~1 million days, respectively, assuming a cubic relationship of N to time required. A simulation of $N = 1000$ was carried out in figure 4.5. Notably, the simulation of 1e-4 mM $O_2$/s/cell at N=1000 is closer to 1e-4 mM $O_2$/s/cell with N=400 than with N=200 (figure 4.4), indicating convergence of values with increasing N. Such simulations with high values of N could be carried out on a supercomputer, but supercomputers reduce time the most using parallel computing. Iterations to calculate $u(t)$ currently are conducted using serial processing. Despite the limitations described above, some interesting conclusions can be deduced from the computational model. First, during the simulation oxygen concentration is relatively flat with respect to distance away from the cell, x. Measured OCR is also relatively linear over time. This is predicted because of the small distances between cells (100μm using $10^6$ cells). Interestingly, there is nonlinearity when oxygen concentrations near zero, something we also noticed in all experiments.

Nonlinearity of oxygen concentration from the Seahorse was more apparent than in the other OCR methods. This could be due in part to an oxygen gradient forming at the bottom of the Seahorse wells, with oxygen consumed at the level of the cells being replaced by oxygen diffusing from elsewhere in the wells. As oxygen level near the cells decreases, oxygen diffusion towards the cells increases, creating nonlinearity in the measurements. While this affects OCR measurements using our experimental setup, it may not have a significant effect during more normal operation.[41] This nonlinearity made selection of OCR rate for comparison in the bar graphs challenging (figures 4.9, 4.11, 4.19). Different rates are observed at different times during the experiment. For this reason, OCR for the bar graphs and statistical analyses were selected during the first minute of data collection after the mixing cycle.

EPRI was used to extract 3D OCR from bioprinted models, showing a difference in OCR between different types of cells and yeast with or without pyruvate. Resolution was approximately 250μm, on the same magnitude as the distance between yeast cells in the study. This resolution gives a good overview of oxygen concentrations throughout the print. In the future, we hope the computational model can

predict how the oxygen maps appear on much smaller scales, since diffusion is important on the scale of tens to hundreds of micrometers. To achieve EPR imaging at this scale, higher gradients or probes with more narrow linewidth could be used. Additionally, images were acquired every 18 minutes with 2000 averages and 16 blocks of projections, limited by the need for good EPR signal-to-noise (SNR). Large changes in oxygen concentration much faster than this may result in distorted images. However, RS EPRI allows for image collection much faster than this (up to 2000 times faster if only one average is needed), with the requirement that SNR is higher. This can be achieved by using higher concentration of EPR probe, or improving the sensitivity of the EPR imager. Such improvement would allow for mapping changes on the time scale of milliseconds and seconds as opposed to tens of minutes.

Both single EPRI and EPR measurement in *Saccharomyces cerevisiae* with and without puruvate showed statistical difference in OCR. However, high variation is noted in the experimental pyruvate group. Intriguingly, OCR measured by EPR spectra was higher than OCR measured through EPRI, even when using the same data. "Pyruvate 2" and "No Pyruvate 1" in figure 4.17 are the spectral components from imaging data seen in figure 4.16. The reason for this difference in measured OCR remains to be determined, but could be due in part to increased background as EPR imaging gradients are applied, resulting in less accurate linewidth measurements biased towards narrow linewidths. The authors are currently exploring this issue.

Important similarities and differences were seen between the OCR methods (see table 4.7). The simulation data collected for the table used the same cell concentration as the bioprint, with internal oxygen consumption rate, $f(x)$, as the variable changed to attain OCR similar to the bioprints. The Seahorse measurements in figure 4.8 appeared to closely match the stirring measurements in figure 4.7 as well as EPRI measurements. Slightly higher measurements for the Seahorse OCR data could be explained by the fact that the cells are glued to the bottom of the well plates - they are concentrated on the bottom where the oxygen sensor is placed, while in the Stirring and EPRI measurements, the cells are evenly dispersed throughout the entire volume being measured.

During EPR imaging, bioprints were kept in room conditions: air concentration $CO_2$ and room temperature. These are normal conditions for yeast cells, but mammalian cells, including murine macrophages, need to be maintained under physiologic conditions. Metabolism under these conditions will be different than under physiologic conditions. Currently, our lab is developing a bioprinting chamber to maintain these conditions during EPR imaging. This chamber will allow the study of bioprints

with mammalian cells under controlled, physiologic conditions.

The methodology demonstrated here is proposed to quantify the success of methods of delivering oxygen to cells in thick bioprints. In future studies, the evaluation of improvements of oxygen delivery in response to vascularization and the release of oxygen through peroxides are proposed.

## 4.6   Acknowledgements

# References

[1] Raees S, Ullah F, Javed F, Akil HM, Jadoon Khan M, Safdar M, Din IU, Alotaibi MA, Alharthi AI, Bakht MA, Ahmad A, and Nassar AA (2023) Classification, processing, and applications of bioink and 3D bioprinting: A detailed review. International Journal of Biological Macromolecules 232:123476. ISSN 01418130. doi:10.1016/j.ijbiomac.2023.123476. URL https://linkinghub.elsevier.com/retrieve/pii/S0141813023003689.

[2] Li W, Wang M, Ma H, Chapa-Villarreal FA, Lobo AO, and Zhang YS (2023) Stereolithography apparatus and digital light processing-based 3D bioprinting for tissue fabrication. iScience 26(2):106039. ISSN 25890042. doi:10.1016/j.isci.2023.106039. URL https://linkinghub.elsevier.com/retrieve/pii/S2589004223001165.

[3] Mirdamadi E, Tashman JW, Shiwarski DJ, Palchesko RN, and Feinberg AW (2020) FRESH 3D Bioprinting a Full-Size Model of the Human Heart. ACS Biomaterials Science & Engineering 6(11):6453–6459. ISSN 2373-9878, 2373-9878. doi:10.1021/acsbiomaterials.0c01133. URL https://pubs.acs.org/doi/10.1021/acsbiomaterials.0c01133.

[4] Szklanny AA, Machour M, Redenski I, Chochola V, Goldfracht I, Kaplan B, Epshtein M, Simaan Yameen H, Merdler U, Feinberg A, Seliktar D, Korin N, Jaroš J, and Levenberg S (2021) 3D Bioprinting of Engineered Tissue Flaps with Hierarchical Vessel Networks (VesselNet) for Direct Host-To-Implant Perfusion. Advanced Materials 33(42):2102661. ISSN 0935-9648, 1521-4095. doi:10.1002/adma.202102661. URL https://onlinelibrary.wiley.com/doi/10.1002/adma.202102661.

[5] Engel M, Belfiore L, Aghaei B, and Sutija M (2022) Enabling high throughput drug discovery in 3D cell cultures through a novel bioprinting workflow. SLAS Technology 27(1):32–38. ISSN 24726303. doi:10.1016/j.slast.2021.10.002. URL https://linkinghub.elsevier.com/retrieve/pii/S2472630321000029.

[6] Sharma R, Restan Perez M, Da Silva VA, Thomsen J, Bhardwaj L, Andrade TAM, Alhussan A, and Willerth SM (2023) 3D bioprinting complex models of cancer. Biomaterials Science 11(10):3414–

3430. ISSN 2047-4830, 2047-4849. doi:10.1039/D2BM02060B. URL http://xlink.rsc.org/?DOI=D2BM02060B.

[7] Kolesky DB, Homan KA, Skylar-Scott MA, and Lewis JA (2016) Three-dimensional bioprinting of thick vascularized tissues. Proceedings of the National Academy of Sciences 113(12):3179–3184. ISSN 0027-8424, 1091-6490. doi:10.1073/pnas.1521342113. URL https://pnas.org/doi/full/10.1073/pnas.1521342113.

[8] Grebenyuk S, Abdel Fattah AR, Kumar M, Toprakhisar B, Rustandi G, Vananroye A, Salmon I, Verfaillie C, Grillo M, and Ranga A (2023) Large-scale perfused tissues via synthetic 3D soft microfluidics. Nature Communications 14(1):193. ISSN 2041-1723. doi:10.1038/s41467-022-35619-1. URL https://www.nature.com/articles/s41467-022-35619-1.

[9] Sekine W and Haraguchi Y (2011) Thickness limitation and cell viability of multi-layered cell sheets and overcoming the diffusion limit by a porous-membrane culture insert. Journal of Biochips & Tissue Chips 01(01). ISSN 21530777. doi:10.4172/2153-0777.S1-007. URL https://www.omicsonline.org/thickness-limitation-and-cell-viability-of-multilayered-cell-sheets-and-overcoming-the-diphp?aid=9796.

[10] Fritschen A, Acedo Mestre M, Scholpp S, and Blaeser A (2023) Influence of the physico-chemical bioink composition on the printability and cell biological properties in 3D-bioprinting of a liver tumor cell line. Frontiers in Bioengineering and Biotechnology 11:1093101. ISSN 2296-4185. doi:10.3389/fbioe.2023.1093101. URL https://www.frontiersin.org/articles/10.3389/fbioe.2023.1093101/full.

[11] Ke D, Kengla C, Lee SJ, Yoo JJ, Zhu X, and Murphy SV (2022) Release Kinetics and In Vitro Characterization of Sodium Percarbonate and Calcium Peroxide to Oxygenate Bioprinted Tissue Models. International Journal of Molecular Sciences 23(12):6842. ISSN 1422-0067. doi:10.3390/ijms23126842. URL https://www.mdpi.com/1422-0067/23/12/6842.

[12] Gholipourmalekabadi M, Zhao S, Harrison BS, Mozafari M, and Seifalian AM (2016) Oxygen-Generating Biomaterials: A New, Viable Paradigm for Tissue Engineering? Trends in Biotech-

nology 34(12):1010–1021. ISSN 01677799. doi:10.1016/j.tibtech.2016.05.012. URL https://linkinghub.elsevier.com/retrieve/pii/S0167779916300580.

[13] Ortega JS, Corrales-Orovio R, Ralph P, Egaña JT, and Gentile C (2023) Photosynthetic microorganisms for the oxygenation of advanced 3D bioprinted tissues. Acta Biomaterialia 165:180–196. ISSN 17427061. doi:10.1016/j.actbio.2022.05.009. URL https://linkinghub.elsevier.com/retrieve/pii/S1742706122002781.

[14] Maharjan S, Alva J, Cámara C, Rubio AG, Hernández D, Delavaux C, Correa E, Romo MD, Bonilla D, Santiago ML, Li W, Cheng F, Ying G, and Zhang YS (2021) Symbiotic Photosynthetic Oxygenation within 3D-Bioprinted Vascularized Tissues. Matter 4(1):217–240. ISSN 25902385. doi:10.1016/j.matt.2020.10.022. URL https://linkinghub.elsevier.com/retrieve/pii/S2590238520305762.

[15] Mir A, Lee E, Shih W, Koljaka S, Wang A, Jorgensen C, Hurr R, Dave A, Sudheendra K, and Hibino N (2023) 3D Bioprinting for Vascularization. Bioengineering 10(5):606. ISSN 2306-5354. doi:10.3390/bioengineering10050606. URL https://www.mdpi.com/2306-5354/10/5/606.

[16] Kolesky DB, Truby RL, Gladman AS, Busbee TA, Homan KA, and Lewis JA (2014) 3D Bioprinting of Vascularized, Heterogeneous Cell-Laden Tissue Constructs. Advanced Materials 26(19):3124–3130. ISSN 09359648. doi:10.1002/adma.201305506. URL https://onlinelibrary.wiley.com/doi/10.1002/adma.201305506.

[17] Arakawa C, Gunnarsson C, Howard C, Bernabeu M, Phong K, Yang E, DeForest CA, Smith JD, and Zheng Y (2020) Biophysical and biomolecular interactions of malaria-infected erythrocytes in engineered human capillaries. Science Advances 6(3):eaay7243. ISSN 2375-2548. doi:10.1126/sciadv.aay7243. URL https://www.science.org/doi/10.1126/sciadv.aay7243.

[18] Alemdar N, Leijten J, Camci-Unal G, Hjortnaes J, Ribas J, Paul A, Mostafalu P, Gaharwar AK, Qiu Y, Sonkusale S, Liao R, and Khademhosseini A (2017) Oxygen-Generating Photo-Cross-Linkable Hydrogels Support Cardiac Progenitor Cell Survival by Reducing Hypoxia-Induced Necrosis. ACS Biomaterials Science & Engineering 3(9):1964–1971. ISSN 2373-9878, 2373-9878. doi:10.1021/acsbiomaterials.6b00109. URL https://pubs.acs.org/doi/10.1021/acsbiomaterials.6b00109.

[19] Figueiredo L, Le Visage C, Weiss P, and Yang J (2020) Quantifying Oxygen Levels in 3D Bio-printed Cell-Laden Thick Constructs with Perfusable Microchannel Networks. Polymers 12(6):1260. ISSN 2073-4360. doi:10.3390/polym12061260. URL https://www.mdpi.com/2073-4360/12/6/1260.

[20] García-Lizarribar A, Villasante A, Lopez-Martin JA, Flandez M, Soler-Vázquez MC, Serra D, Herrero L, Sagrera A, Efeyan A, and Samitier J (2023) 3D bioprinted functional skeletal muscle models have potential applications for studies of muscle wasting in cancer cachexia. Biomaterials Advances 150:213426. ISSN 27729508. doi:10.1016/j.bioadv.2023.213426. URL https://linkinghub.elsevier.com/retrieve/pii/S2772950823001498.

[21] Wesseler MF, Johansen MN, Kızıltay A, Mortensen KI, and Larsen NB (2022) Optical 4D oxygen mapping of microperfused tissue models with tunable *in vivo* -like 3D oxygen microenvironments. Lab on a Chip 22(21):4167–4179. ISSN 1473-0197, 1473-0189. doi:10.1039/D2LC00063F. URL http://xlink.rsc.org/?DOI=D2LC00063F.

[22] Shibuya K, Saito H, Nishikido F, Takahashi M, and Yamaya T (2020) Oxygen sensing ability of positronium atom for tumor hypoxia imaging. Communications Physics 3(1):173. ISSN 2399-3650. doi:10.1038/s42005-020-00440-z. URL https://www.nature.com/articles/s42005-020-00440-z.

[23] Fan AP, An H, Moradi F, Rosenberg J, Ishii Y, Nariai T, Okazawa H, and Zaharchuk G (2020) Quantification of brain oxygen extraction and metabolism with [15O]-gas PET: A technical review in the era of PET/MRI. NeuroImage 220:117136. ISSN 10538119. doi:10.1016/j.neuroimage.2020.117136. URL https://linkinghub.elsevier.com/retrieve/pii/S1053811920306224.

[24] Xiang X, Wind K, Wiedemann T, Blume T, Shi Y, Briel N, Beyer L, Biechele G, Eckenweber F, Zatcepin A, Lammich S, Ribicic S, Tahirovic S, Willem M, Deussing M, Palleis C, Rauchmann BS, Gildehaus FJ, Lindner S, Spitz C, Franzmeier N, Baumann K, Rominger A, Bartenstein P, Ziegler S, Drzezga A, Respondek G, Buerger K, Perneczky R, Levin J, Höglinger GU, Herms J, Haass C, and Brendel M (2021) Microglial activation states drive glucose uptake and FDG-PET alterations in neurodegenerative diseases. Science Translational Medicine 13(615):eabe5640. ISSN

1946-6234, 1946-6242.  doi:10.1126/scitranslmed.abe5640.  URL https://www.science.org/doi/10.1126/scitranslmed.abe5640.

[25] Wang F, Jin P, Feng Y, Fu J, Wang P, Liu X, Zhang Y, Ma Y, Yang Y, Yang A, and Feng X (2021) Flexible Doppler ultrasound device for the monitoring of blood flow velocity. Science Advances 7(44):eabi9283.  ISSN 2375-2548.  doi:10.1126/sciadv.abi9283.  URL https://www.science.org/doi/10.1126/sciadv.abi9283.

[26] Matsumoto Ki, Bernardo M, Subramanian S, Choyke P, Mitchell JB, Krishna MC, and Lizak MJ (2006) MR assessment of changes of tumor in response to hyperbaric oxygen treatment. Magnetic Resonance in Medicine 56(2):240–246.  ISSN 0740-3194, 1522-2594.  doi:10.1002/mrm.20961. URL https://onlinelibrary.wiley.com/doi/10.1002/mrm.20961.

[27] Salem A, Little RA, Latif A, Featherstone AK, Babur M, Peset I, Cheung S, Watson Y, Tessyman V, Mistry H, Ashton G, Behan C, Matthews JC, Asselin MC, Bristow RG, Jackson A, Parker GJ, Faivre-Finn C, Williams KJ, and O'Connor JP (2019) Oxygen-enhanced MRI Is Feasible, Repeatable, and Detects Radiotherapy-induced Change in Hypoxia in Xenograft Models and in Patients with Non–small Cell Lung Cancer.  Clinical Cancer Research 25(13):3818–3829.  ISSN 1078-0432, 1557-3265.  doi:10.1158/1078-0432. CCR-18-3932.  URL https://aacrjournals.org/clincancerres/article/25/13/3818/81902/Oxygen-enhanced-MRI-Is-Feasible-Repeatable-and.

[28] Elas M, Ahn KH, Parasca A, Barth ED, Lee D, Haney C, and Halpern HJ (2006) Electron Paramagnetic Resonance Oxygen Images Correlate Spatially and Quantitatively with Oxylite Oxygen Measurements. Clinical Cancer Research 12(14):4209–4217. ISSN 1078-0432, 1557-3265. doi:10.1158/1078-0432.CCR-05-0446. URL https://aacrjournals.org/clincancerres/article/12/14/4209/284777/Electron-Paramagnetic-Resonance-Oxygen-Images.

[29] Subramanian S, Devasahayam N, McMillan A, Matsumoto S, Munasinghe JP, Saito K, Mitchell JB, Chandramouli GV, and Krishna MC (2012) Reporting of quantitative oxygen mapping in EPR imaging. Journal of Magnetic Resonance 214:244–251. ISSN 10907807. doi:10.1016/j.jmr.2011.11.013. URL https://linkinghub.elsevier.com/retrieve/pii/S1090780711004794.

[30] Gallez B (2021) Oxygenation Status in Normal Tissues, Pathological Tissues and Malignant Tumors: A pO2 Database Based on Electron Paramagnetic Resonance (EPR) Oximetry Measurements. Applied Magnetic Resonance 52(10):1395–1450. ISSN 0937-9347, 1613-7507. doi:10.1007/s00723-021-01358-7. URL https://link.springer.com/10.1007/s00723-021-01358-7.

[31] Bobko AA, Eubank TD, Driesschaert B, and Khramtsov VV (2018) In Vivo EPR Assessment of pH, pO$_2$, Redox Status, and Concentrations of Phosphate and Glutathione in the Tumor Microenvironment. Journal of Visualized Experiments (133):56624. ISSN 1940-087X. doi:10.3791/56624. URL https://www.jove.com/t/56624/in-vivo-epr-assessment-ph-po2-redox-status-concentrations-phosphate.

[32] Gluth TD, Poncelet M, Gencheva M, Hoblitzell EH, Khramtsov VV, Eubank TD, and Driesschaert B (2022) Biocompatible Monophosphonated Trityl Spin Probe, HOPE71, for In Vivo Measurement of pO$_2$, pH, and [P$_i$] by Electron Paramagnetic Resonance Spectroscopy. Analytical Chemistry acs.analchem.2c03476. ISSN 0003-2700, 1520-6882. doi:10.1021/acs.analchem.2c03476. URL https://pubs.acs.org/doi/abs/10.1021/acs.analchem.2c03476.

[33] Poncelet M, Huffman JL, Khramtsov VV, Dhimitruka I, and Driesschaert B (2019) Synthesis of hydroxyethyl tetrathiatriarylmethyl radicals OX063 and OX071. RSC Advances 9(60):35073–35076. ISSN 2046-2069. doi:10.1039/C9RA08633A. URL http://xlink.rsc.org/?DOI=C9RA08633A.

[34] Tseytlin O, Guggilapu P, Bobko AA, AlAhmad H, Xu X, Epel B, O'Connell R, Hoblitzell EH, Eubank TD, Khramtsov VV, Driesschaert B, Kazkaz E, and Tseytlin M (2019) Modular imaging system: Rapid scan EPR at 800 MHz. Journal of Magnetic Resonance 305:94–103. ISSN 10907807. doi:10.1016/j.jmr.2019.06.003. URL https://linkinghub.elsevier.com/retrieve/pii/S1090780719301107.

[35] Tseytlin O, O'Connell R, Sivashankar V, Bobko AA, and Tseytlin M (2021) Rapid Scan EPR Oxygen Imaging in Photoactivated Resin Used for Stereolithographic 3D Printing. 3D Printing and Additive Manufacturing 8(6):358–365. ISSN 2329-7662, 2329-7670. doi:10.1089/3dp.2020.0170. URL https://www.liebertpub.com/doi/10.1089/3dp.2020.0170.

[36] O'Connell RC, Tseytlin O, Bobko AA, Eubank TD, and Tseytlin M (2022) Rapid scan EPR: Automated digital resonator control for low-latency data acquisition. Journal of Magnetic Reso-

nance 345:107308. ISSN 10907807. doi:10.1016/j.jmr.2022.107308. URL https://linkinghub.elsevier.com/retrieve/pii/S1090780722001665.

[37] Hinton TJ, Jallerat Q, Palchesko RN, Park JH, Grodzicki MS, Shue HJ, Ramadan MH, Hudson AR, and Feinberg AW (2015) Three-dimensional printing of complex biological structures by freeform reversible embedding of suspended hydrogels. Science Advances 1(9):e1500758. ISSN 2375-2548. doi:10.1126/sciadv.1500758. URL https://www.science.org/doi/10.1126/sciadv.1500758.

[38] Boularaoui S, Al Hussein G, Khan KA, Christoforou N, and Stefanini C (2020) An overview of extrusion-based bioprinting with a focus on induced shear stress and its effect on cell viability. Bioprinting 20:e00093. ISSN 24058866. doi:10.1016/j.bprint.2020.e00093. URL https://linkinghub.elsevier.com/retrieve/pii/S2405886620300208.

[39] Zhang Y, Yu Y, Akkouch A, Dababneh A, Dolati F, and Ozbolat IT (2015) In vitro study of directly bioprinted perfusable vasculature conduits. Biomaterials Science 3(1):134–143. ISSN 2047-4830, 2047-4849. doi:10.1039/C4BM00234B. URL http://xlink.rsc.org/?DOI=C4BM00234B.

[40] Kumar H, Dixit K, Sharma R, MacDonald ME, Sinha N, and Kim K (2023) Closed-loop vasculature network design for bioprinting large, solid tissue scaffolds. Biofabrication 15(2):024104. ISSN 1758-5082, 1758-5090. doi:10.1088/1758-5090/acb73c. URL https://iopscience.iop.org/article/10.1088/1758-5090/acb73c.

[41] Plitzko B and Loesgen S (2018) Measurement of Oxygen Consumption Rate (OCR) and Extracellular Acidification Rate (ECAR) in Culture Cells for Assessment of the Energy Metabolism. BIO-PROTOCOL 8(10). ISSN 2331-8325. doi:10.21769/BioProtoc.2850. URL https://bio-protocol.org/e2850.

## 4.7 Figures
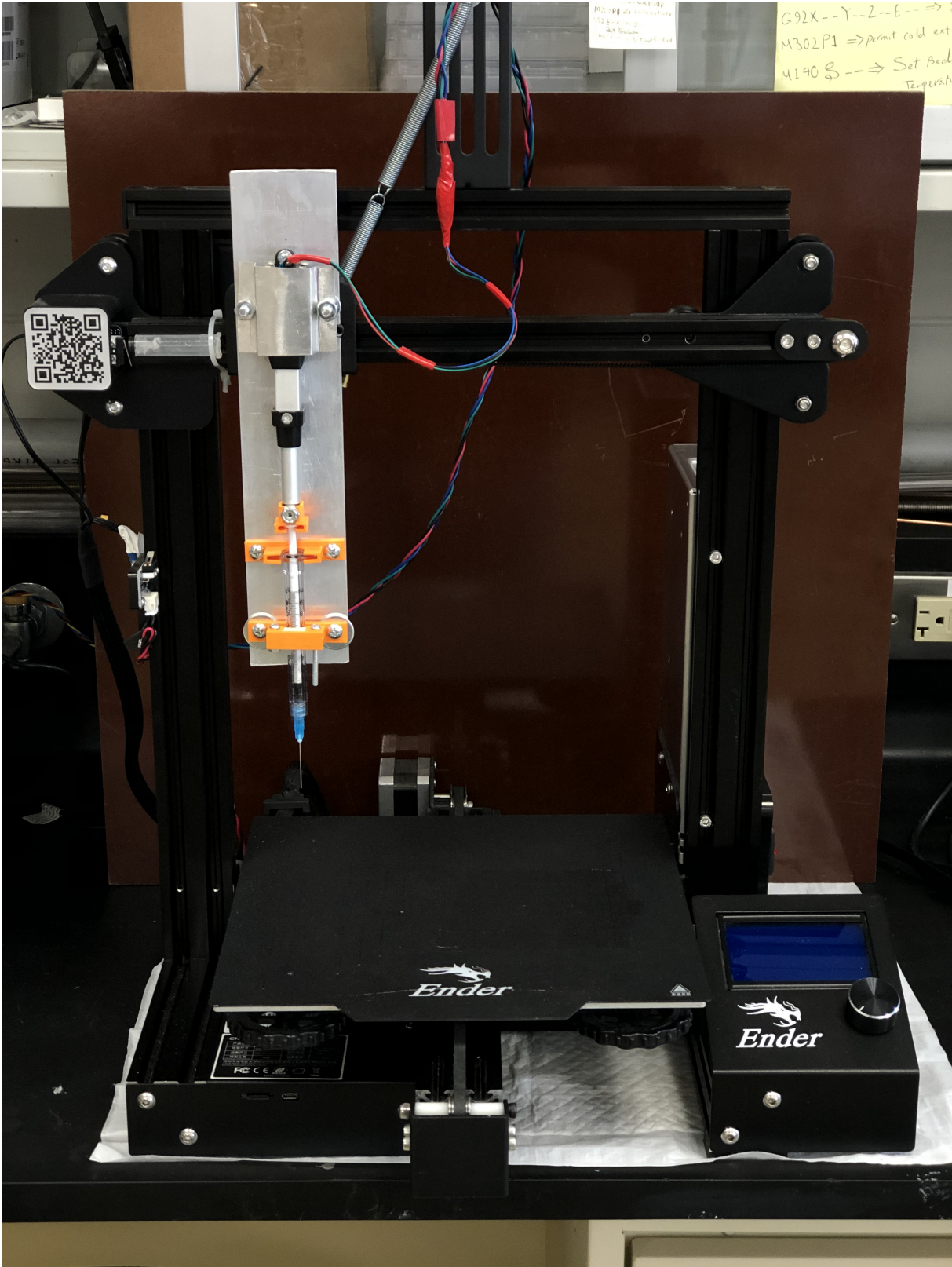
Figure 4.1: Extrusion bioprinter design
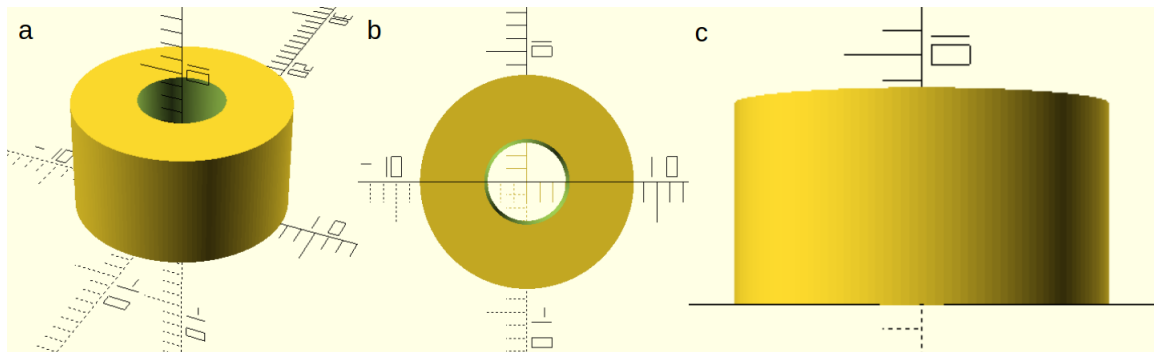
Figure 4.2: CAD model of tube bioprint.
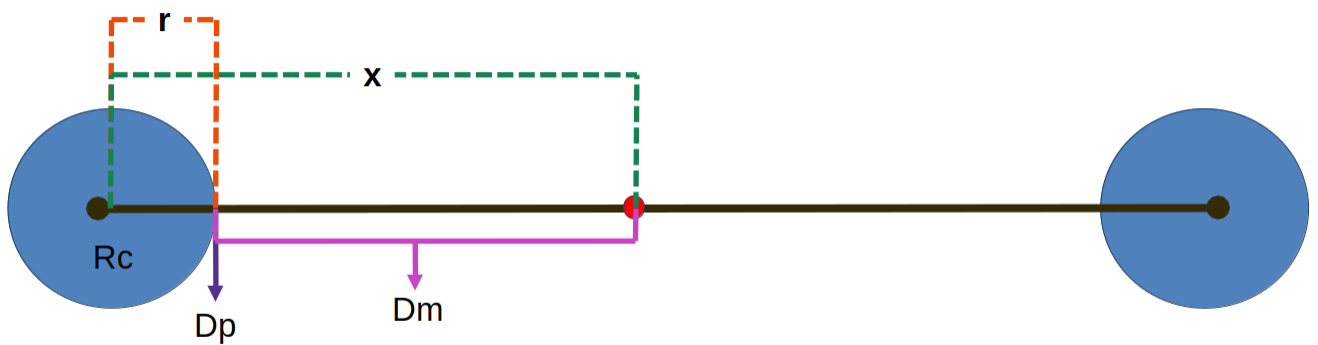


Figure 4.3: Diagram of a theoretical model of oxygen flux. To model the flux of oxygen in 3D culture, assume cells are arranged equidistant in a spherical coordinate system, X. The maximum value of X is the halfway point between two cells, marked as a red circle. This maximum value is dependent on the concentration of cells in 3D culture. The cells have some radius r, and a rate of oxygen consumption, Rc. For simplicity, the extracellular matrix is assumed to have a constant diffusion coefficient, Dm. The plasma membrane and/or cell wall of the cell has a different diffusion coefficient, Dp.
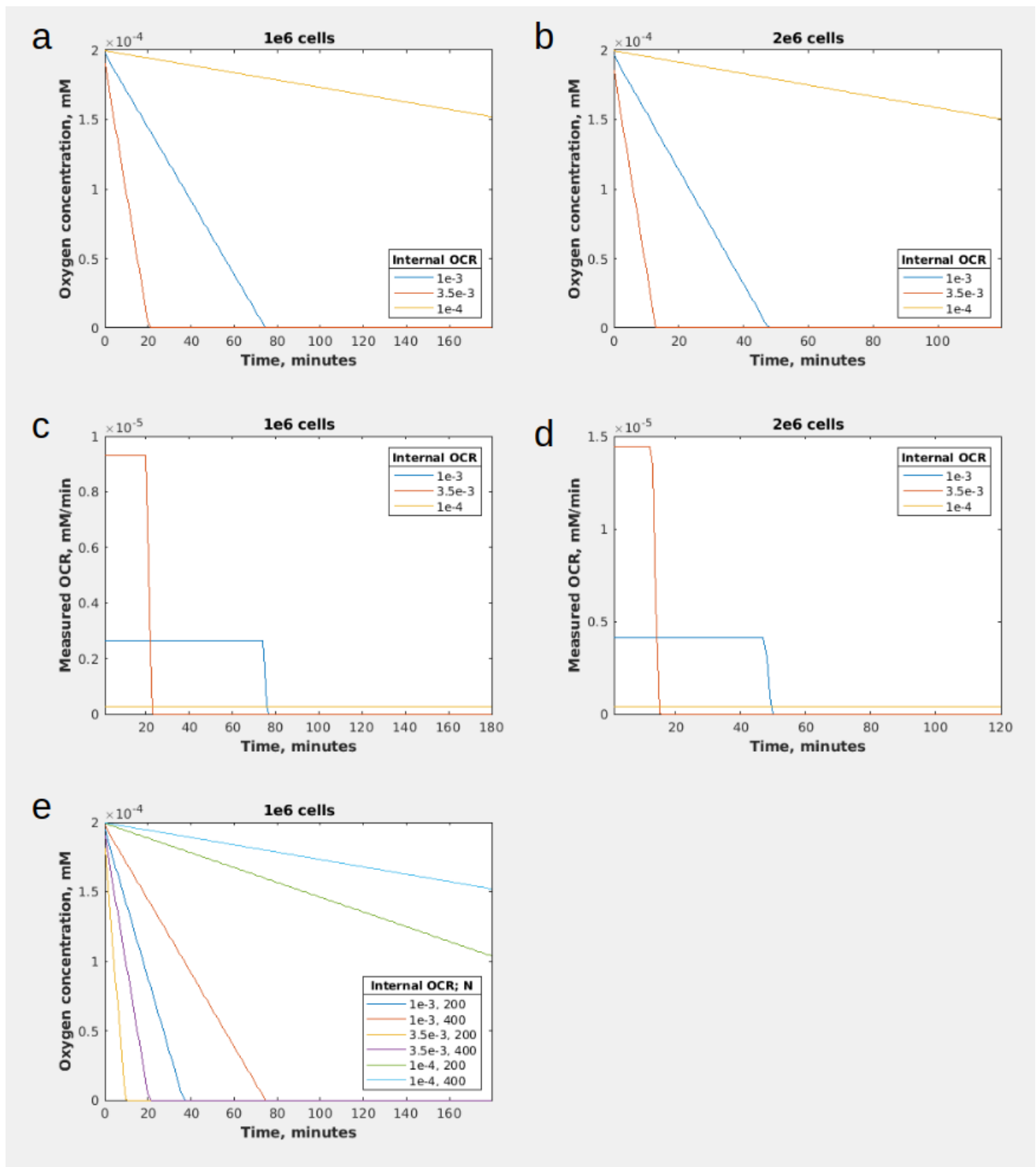
Figure 4.4: 3D oxygen diffusion and consumption simulation. a-b: Predicted oxygen concentration over time with varying internal oxygen consumption rates (OCR) and a) $1*10^6$ cells or b) $2*10^6$ cells per mL. c-d: External OCR with varying internal OCR and c) $1*10^6$ or d) $2*10^6$ cells per mL. e: Comparison of the predicted oxygen concentration over time with varying internal OCR and N. N is defined as the number of increments in x, the spherical distance from the center of the cell and the next cell. Internal OCR is a constant in the simulation, defined as the amount of oxygen (mM $O_2$ consumed per second per cell. External OCR is the rate measured by an observer, in units mM $O_2$ consumed per cell per minute.
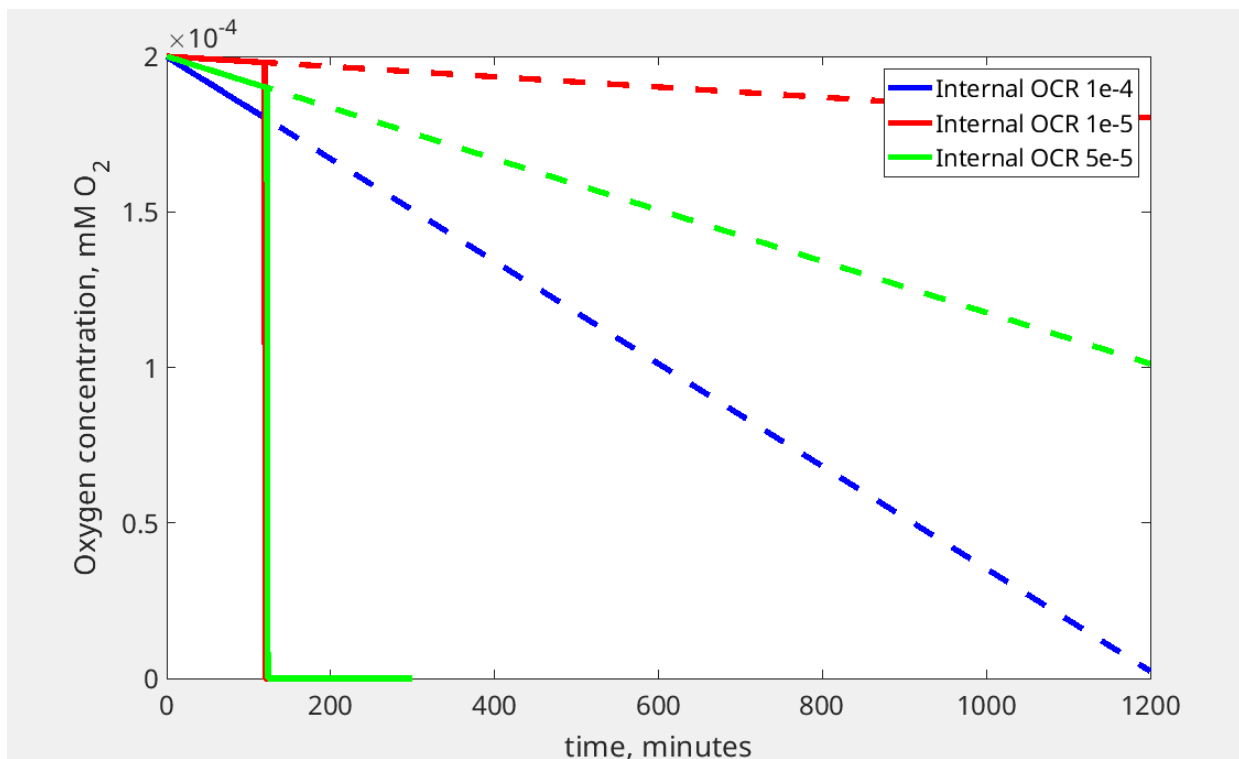
Figure 4.5: 3D oxygen consumption simulation with lower rates and higher N. Predicted oxygen concentration over time with varying internal oxygen consumption rates and $2*10^6$ cells per mL. Solid lines represents simulated data, while dotted lines represent continuation of simulated data assuming continuation of linearity.
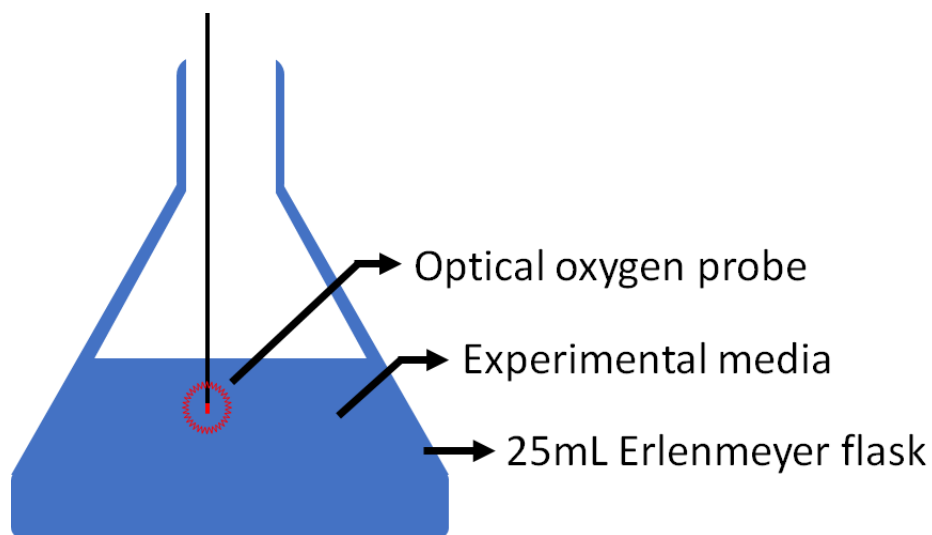


Figure 4.6: Experimental setup for yeast oxygen measurement under stirring conditions.
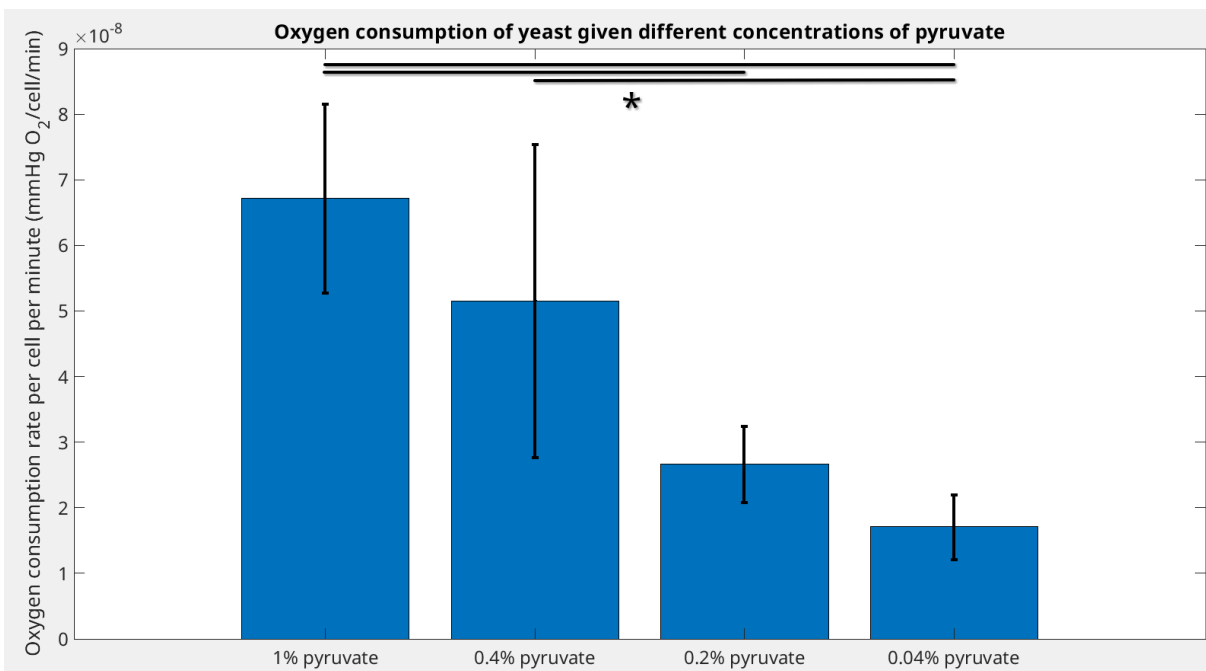
Figure 4.7: Yeast oxygen consumption rates under stirring conditions. Yeast OCR was measured with different concentrations of pyruvate. There were 4 technical replicates in each group. One-way ANOVA and pairwise comparisons test calculated using MATLAB (see tables 4.1 and 4.2 for results and p-values).
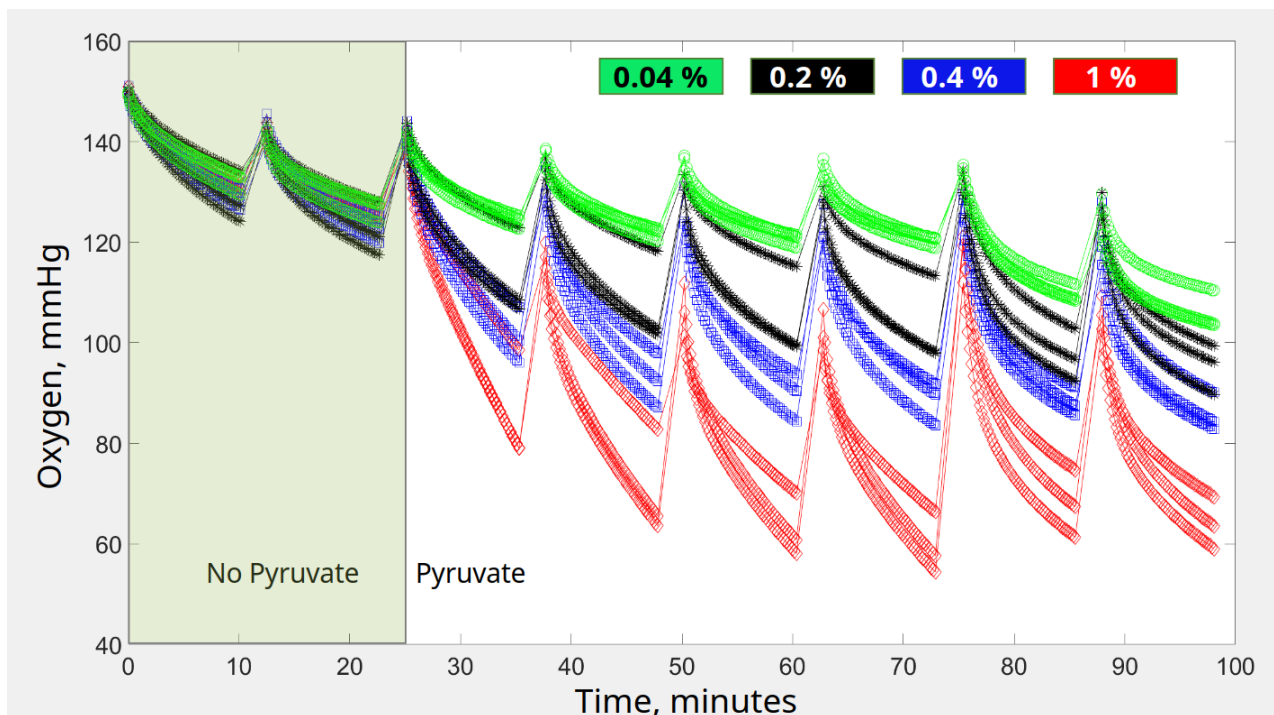
Figure 4.8: Seahorse yeast oxygen measurement under different pyruvate concentrations. Numbers in the legend refer to concentrations of pyruvate. Each line in the same color is a technical replicate, n=3. Seahorse works through cycles of mixing and measurement. Mixing does not occur during measurement. The mixing period can be seen when there is a sharp increase in oxygen $pO_2$ (one example of this is at the demarcation between no pyruvate and pyruvate). From each technical replicate, two cycles of measurement without pyruvate and six cycles of measurement with added pyruvate were taken.
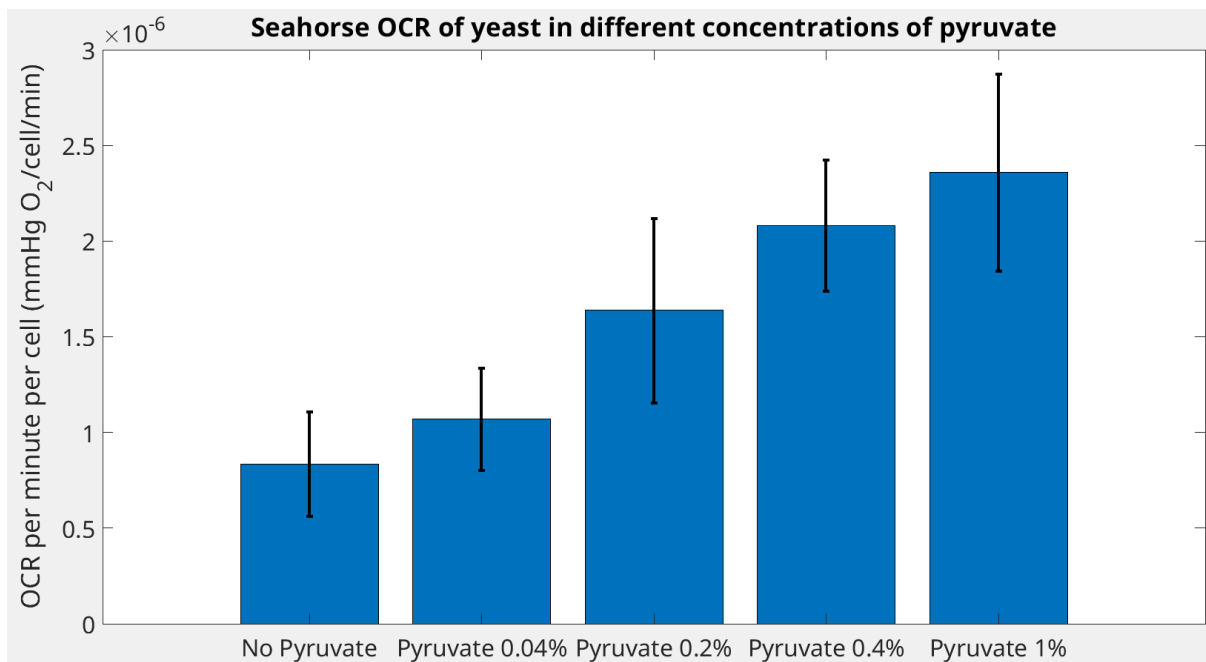
Figure 4.9: Graph of OCR from Seahorse (see figure 4.8). OCR was calculated from the last five points in each replicate after injection of pyruvate solution. Pyruvate 1%, 0.4%, 0.2% and 0.04% groups had 18 technical replicates each, while the background group had 24 and No Pyruvate had 48. All groups are statistically different, with p∼0, except when comparing Pyruvate 0.04% to No Pyruvate and Pyruvate 1% to Pyruvate 0.4%. Results from statistical analysis are shown in tables 4.3 and 4.4.
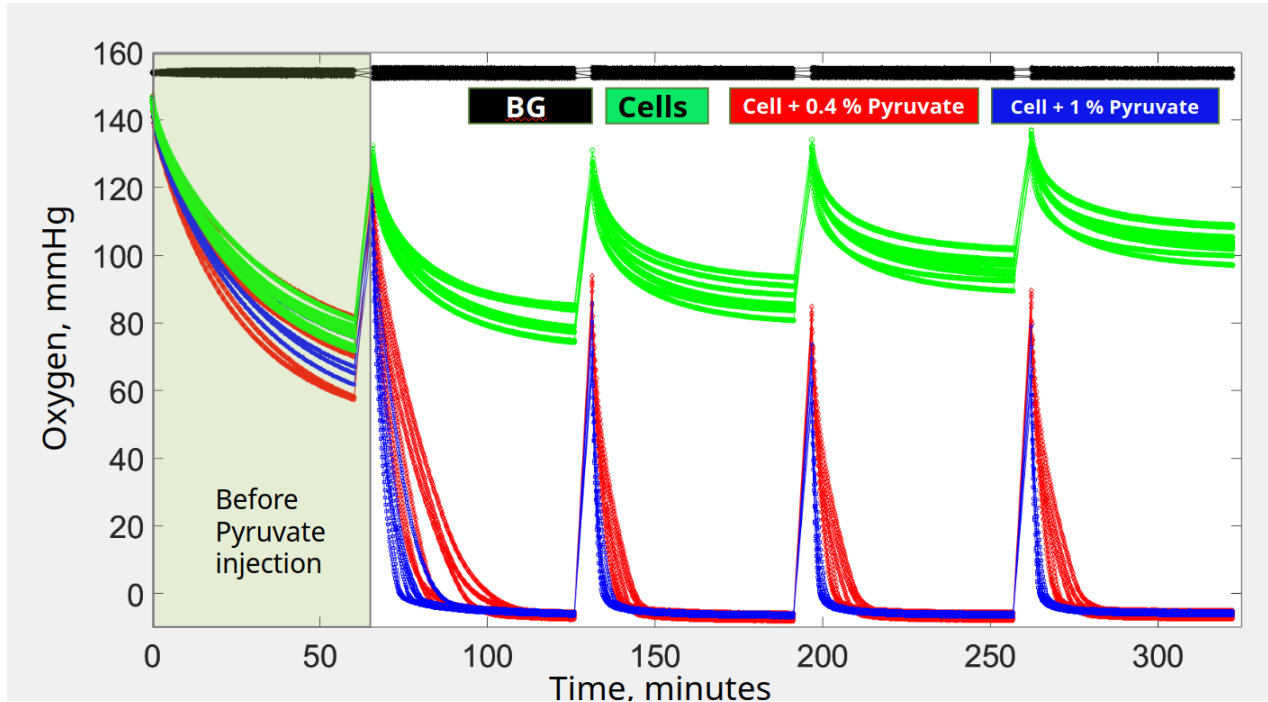
Figure 4.10: Longer-term Seahorse yeast oxygen measurement. Each line in the same color is a technical replicate, n=6. The cells group is cells only, with no added pyruvate. BG is background, the well contains only water.
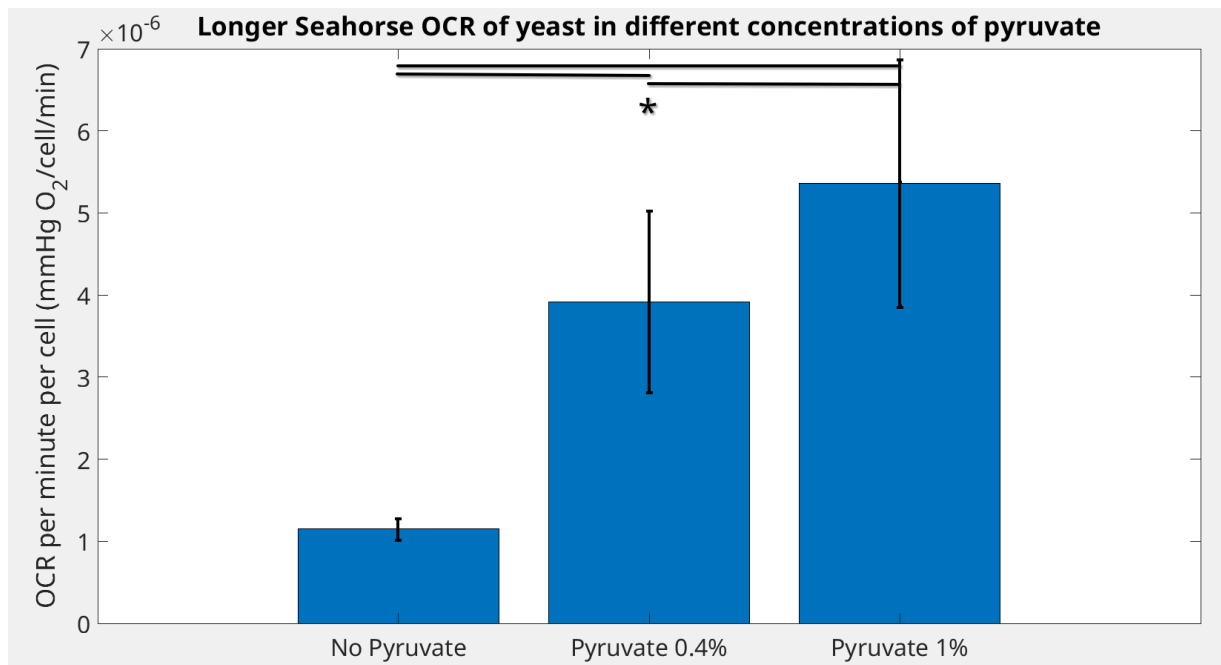


Figure 4.11: Graph of OCR for longer-term Seahorse yeast measurement (see figure 4.10). OCR was calculated from the first 10 points in each replicate after injection of pyruvate solution. Pyruvate 1% and 0.4% had 24 replicates each, while Background had 16 and No Pyruvate had 32. All groups are statistically different. Results from statistical analysis are shown in tables 4.5 and 4.6.
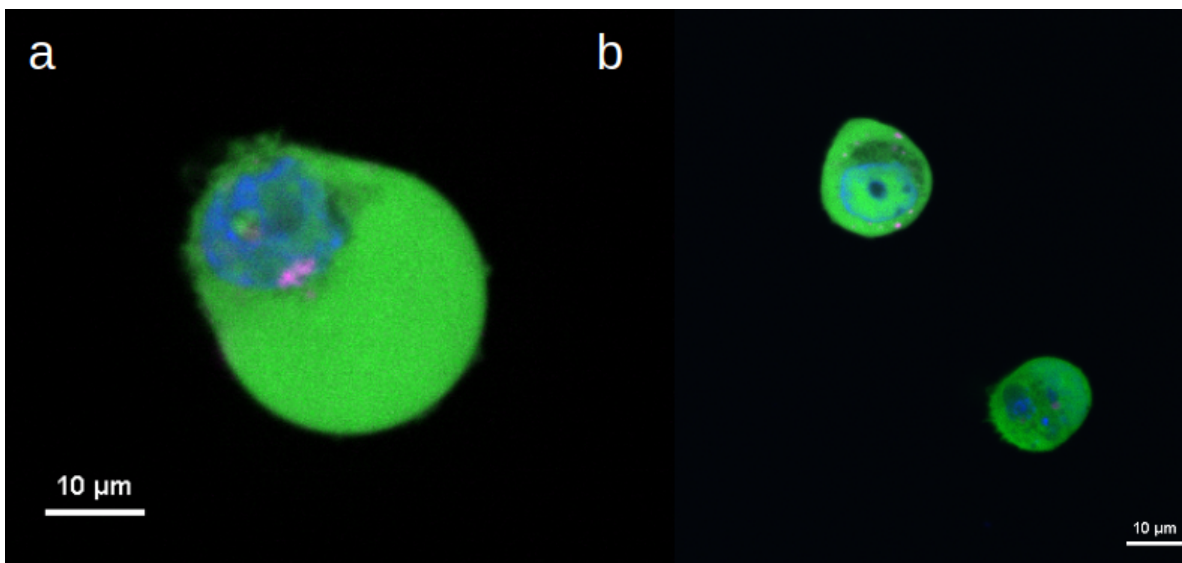
Figure 4.12: Bioprinted cancer cells a) day of bioprinting b) day after bioprinting. Cells are the MDA-MB-231 epithelial human breast cancer cell line. Green is GFP, pink is RFP, and blue is Nucspot live 650 nuclear stain.
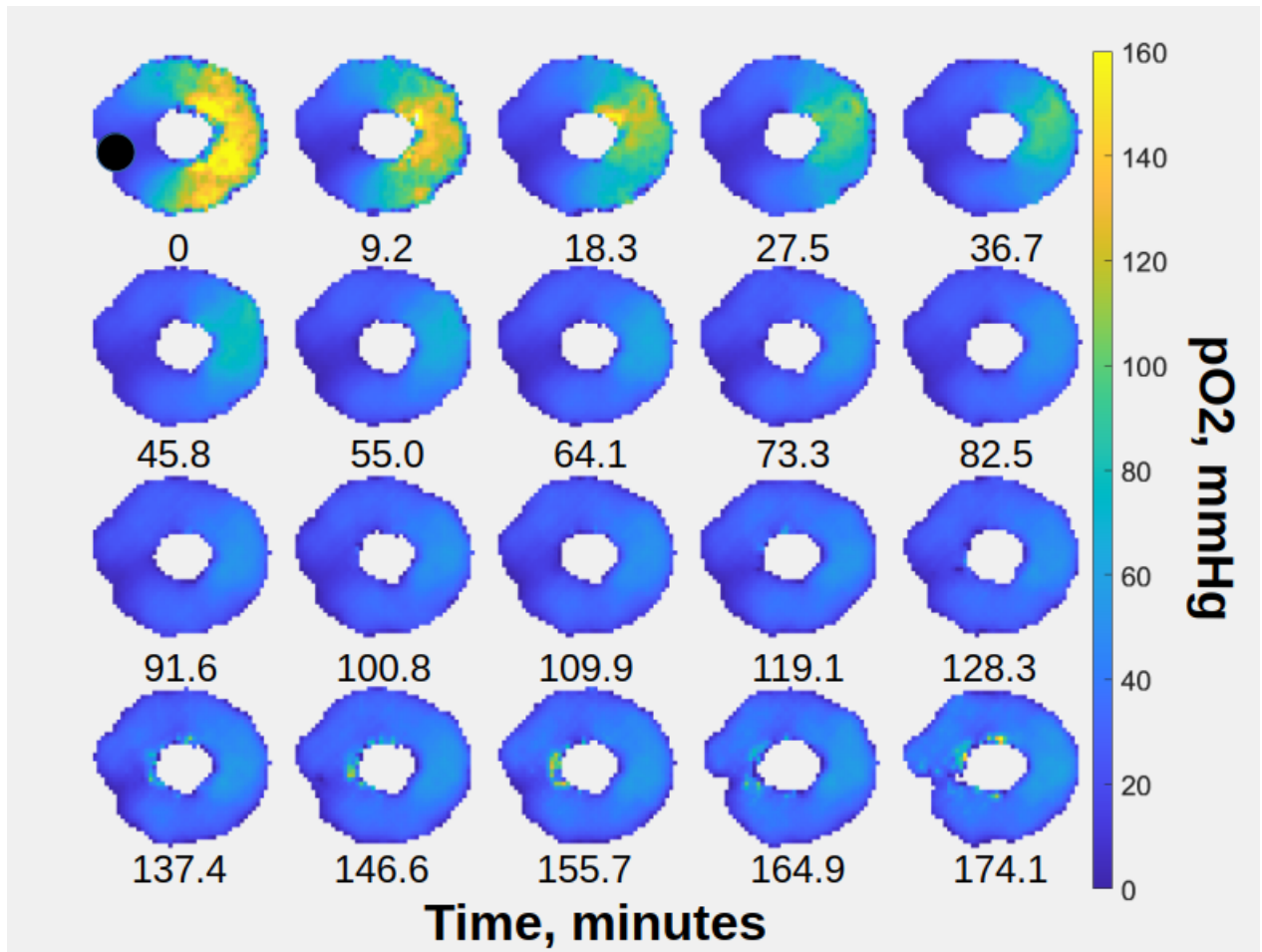
Figure 4.13: Bioprinted model of oxygen diffusion. Oxygen maps show the change in oxygen concentration as a function of time. EPR images were collected continuously over the period of three hours. The black circle in image 1 illustrates the injection site for the GOX solution.
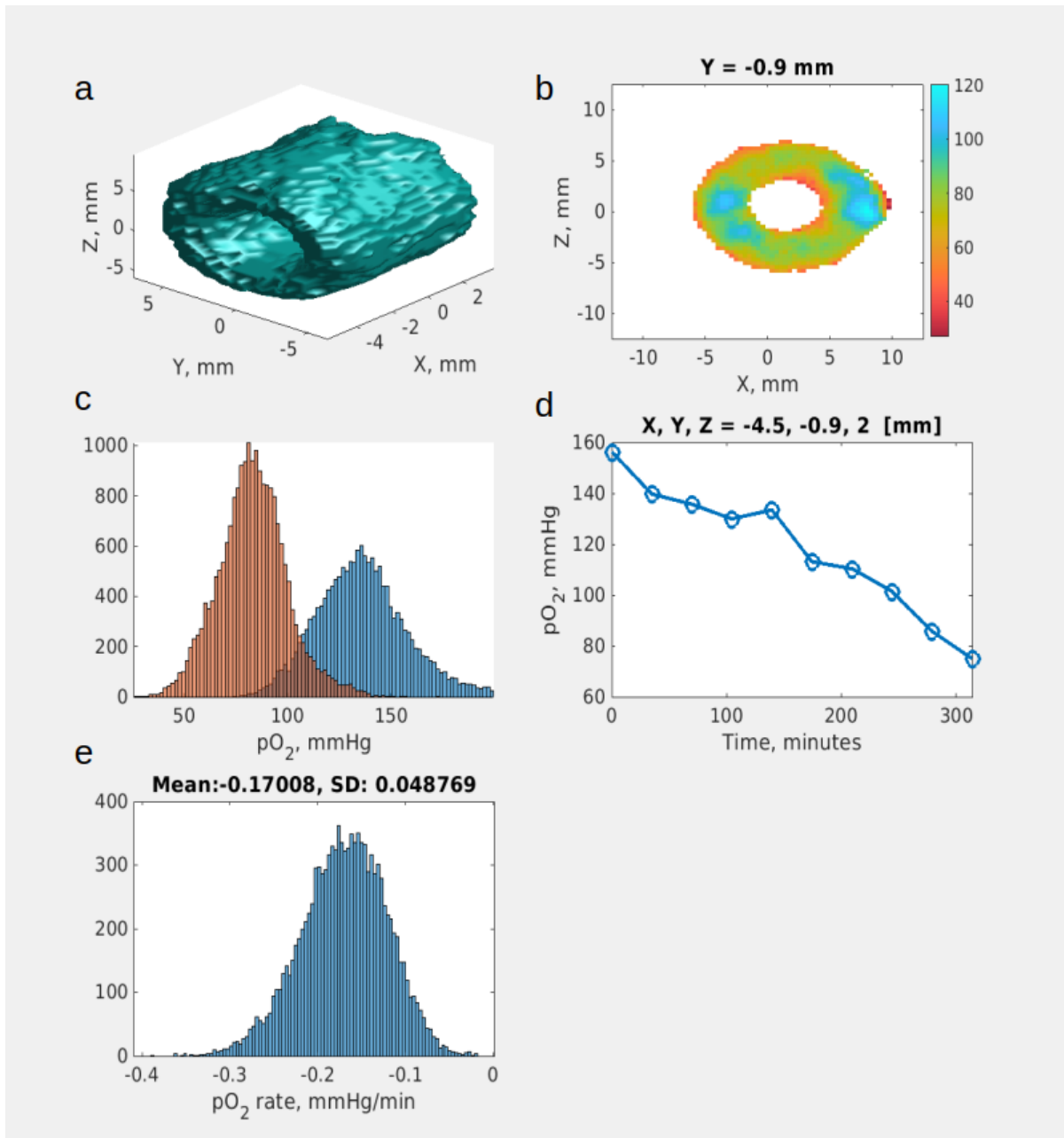
Figure 4.14: EPRI kinetics of yeast in a bioprint. Data are a single experiment (collection of 10 images) taken over the period of approximately 5.5 hours. Data show OCR measurement of yeast (OD 0.25 diluted twice in distilled water) embedded in a bioprint and starved for 6 hours before experiment. a: surface model of reconstructed EPR image showing bioprint as seen by EPRI. b: representative slice of last EPR image in the set of images, taken approximately 5.5 hours after the first image collected. Color bar shows $pO_2$ in mmHg. c: histogram of first and last image showing distribution of oxygen partial pressure across all voxels in the image. d: representative oxygen partial pressure at one voxel across time. e: histogram of oxygen consumption rate calculated at each voxel.
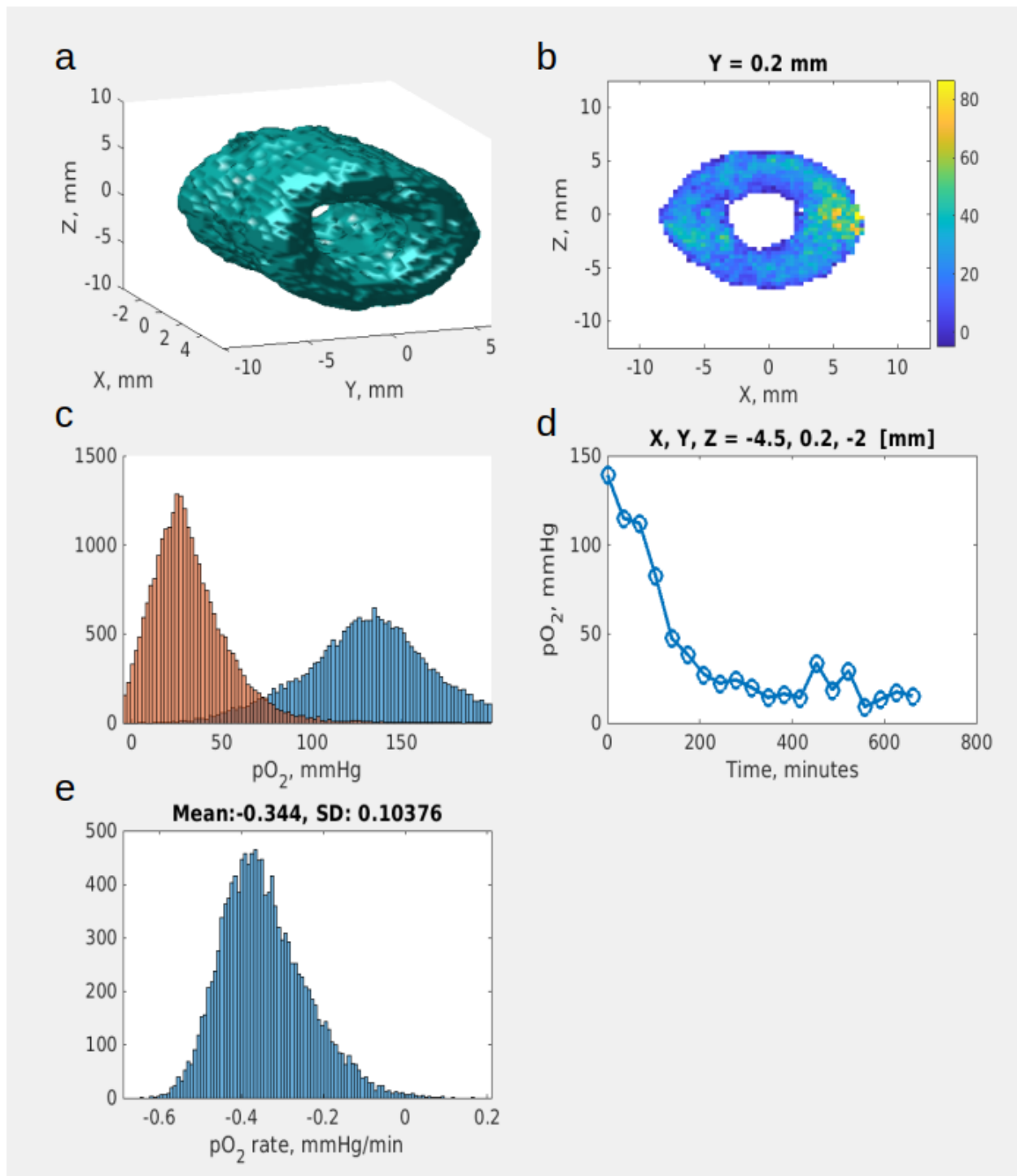
Figure 4.15: EPRI kinetics of yeast in a bioprint containing 1% sodium pyruvate. Data are a single experiment (collection of 20 images) taken over approximately 11 hours. Data show OCR measurement of yeast (OD 0.25 diluted twice in distilled water) embedded in a bioprint containing 1% pyruvate. Bioprint containing yeast and pyruvate rested to complete crosslinking process for the period of 6 hours before the experiment. a: surface model of reconstructed EPR image. b: representative slice of last EPR image in the set of images. Color bar shows $pO_2$ in mmHg. c: histogram of first and last image showing distribution of oxygen partial pressure across all voxels in the image. d: representative oxygen partial pressure at one voxel across time. e: histogram of oxygen consumption rate calculated at each voxel.
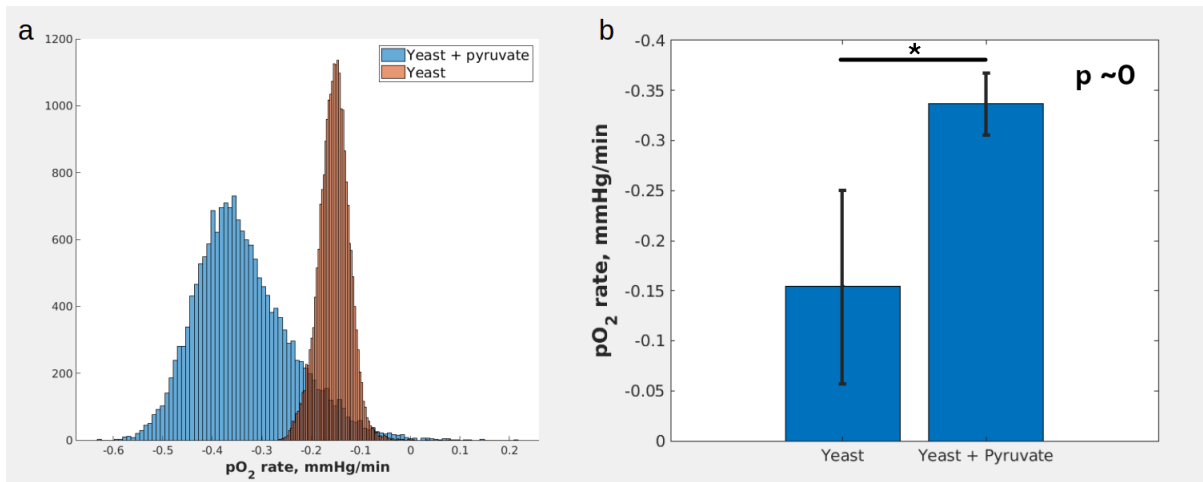
Figure 4.16: EPRI OCR data. a: histogram of yeast with (blue) and without (red) 1% pyruvate in bioprint. Data shown is comparison of figures 4.14e and 4.15e. b: bar chart comparing oxygen consumption rate data and standard deviation of yeast with and without pyruvate. Standard deviation calculated using OCR from each voxel in the images as an individual measurement. Student's t-test used to compare yeast in bioprint with and without pyruvate.
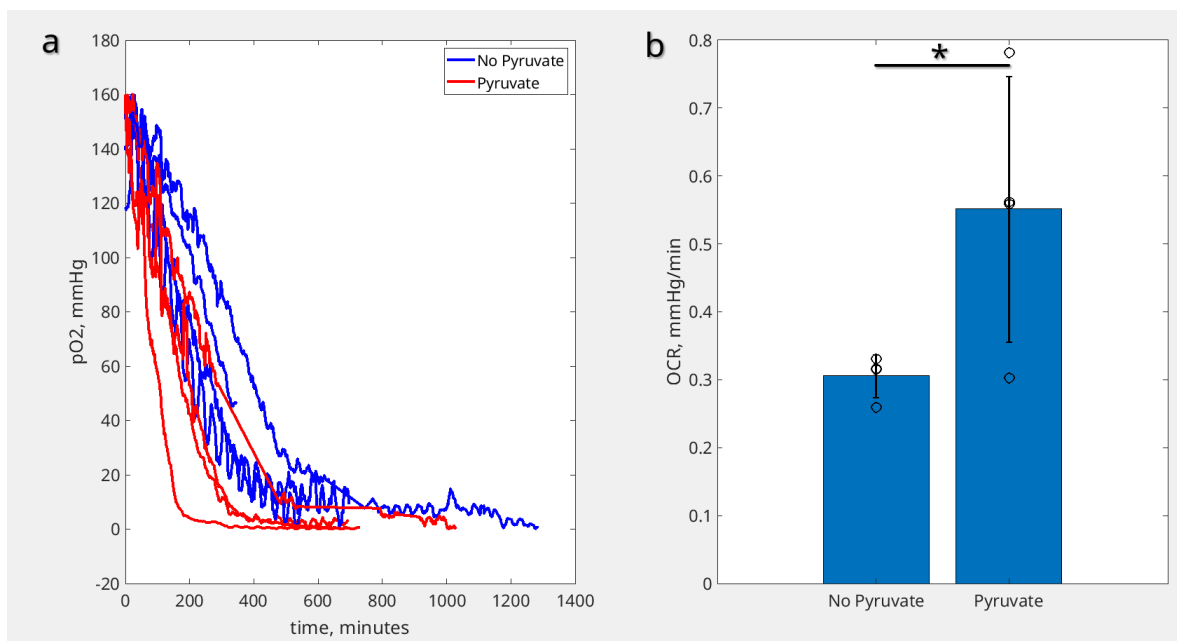


Figure 4.17: Comparison of OCR data. a) EPR spectra of yeast in bioprints with and without 1% pyruvate was collected with four technical replicates in each group (eight bioprints were tested total). Data was smoothed for viewing using moving window average of 10 points. b) OCR was calculated from collected spectra using using LiNc-BuO oxygen sensing probe for two replicates of the pyruvate group, or dFT oxygen sensitive probe in the rest of the replicates. A Student's t-test using MATLAB gives a p value of 0.0478.
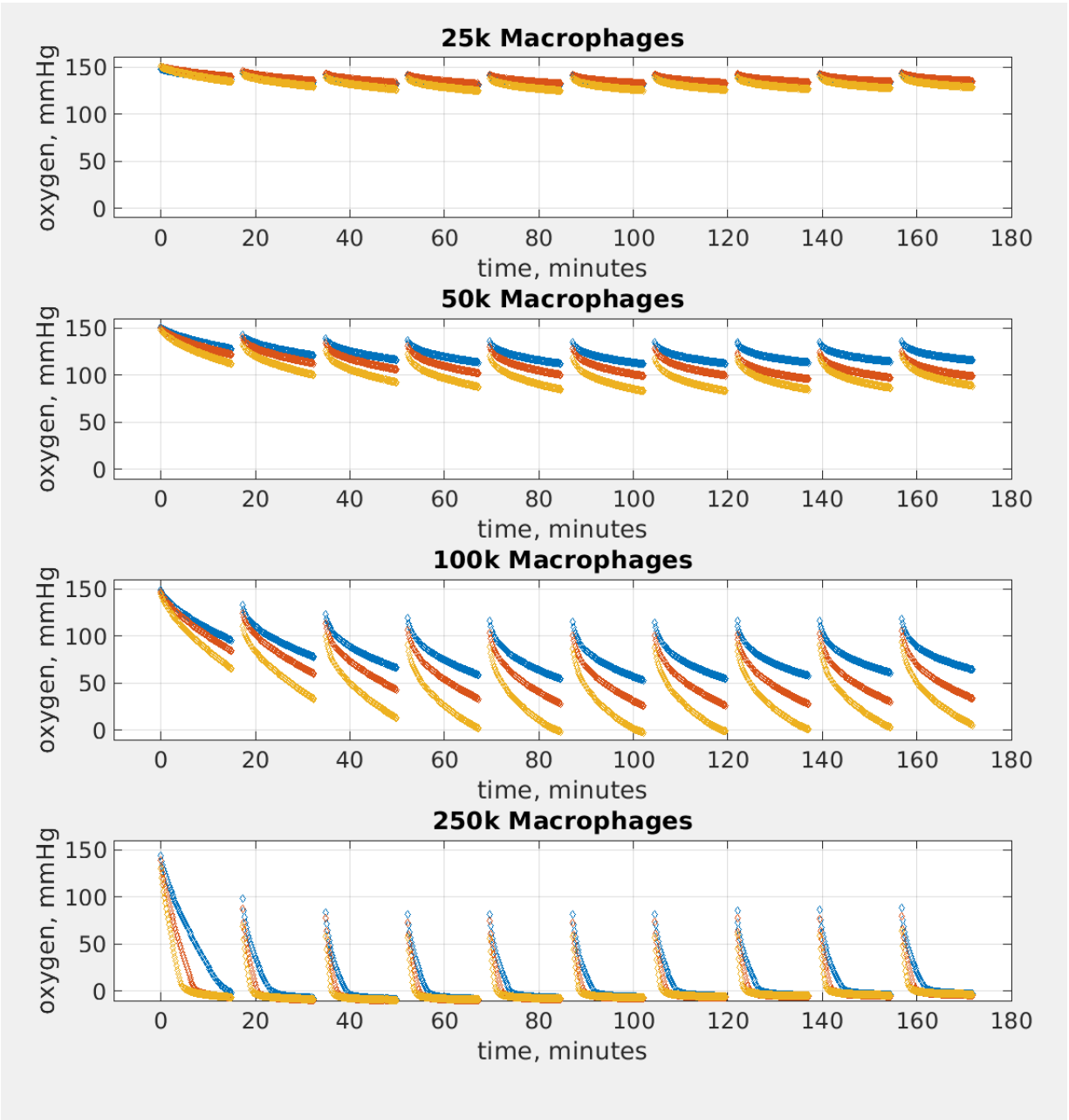
Figure 4.18: Oxygen measurements of murine macrophages in 2D culture using Seahorse (see section 4.3.5). Each line within the graphs represents an individual biological replicate, macrophages taken from a separate mouse (n=3).
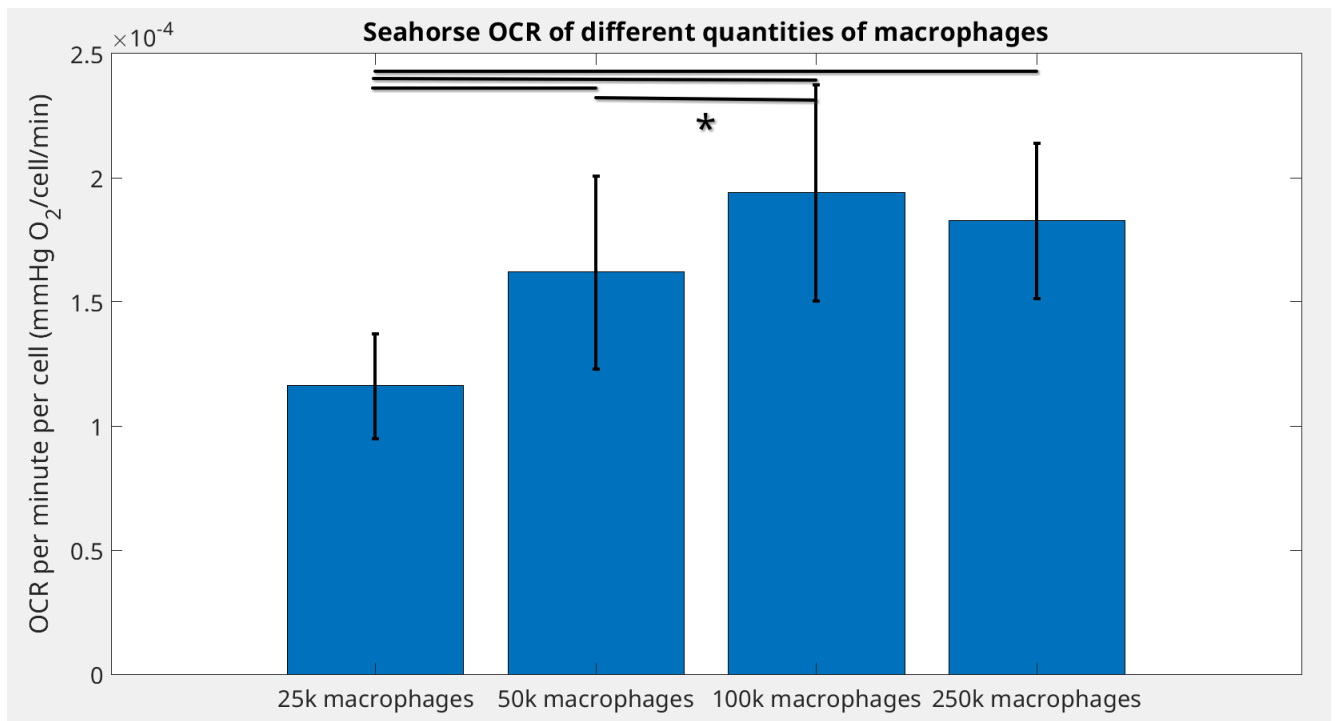
Figure 4.19: OCR measurements of murine macrophages by Seahorse in 2D culture (see figure 4.18). Each group is statistically different, see tables 4.10 and 4.11. The 25,000, 50,000, 100,000 and 250,000 macrophage groups were replicated 30 times each, while Background had 81 replicates.
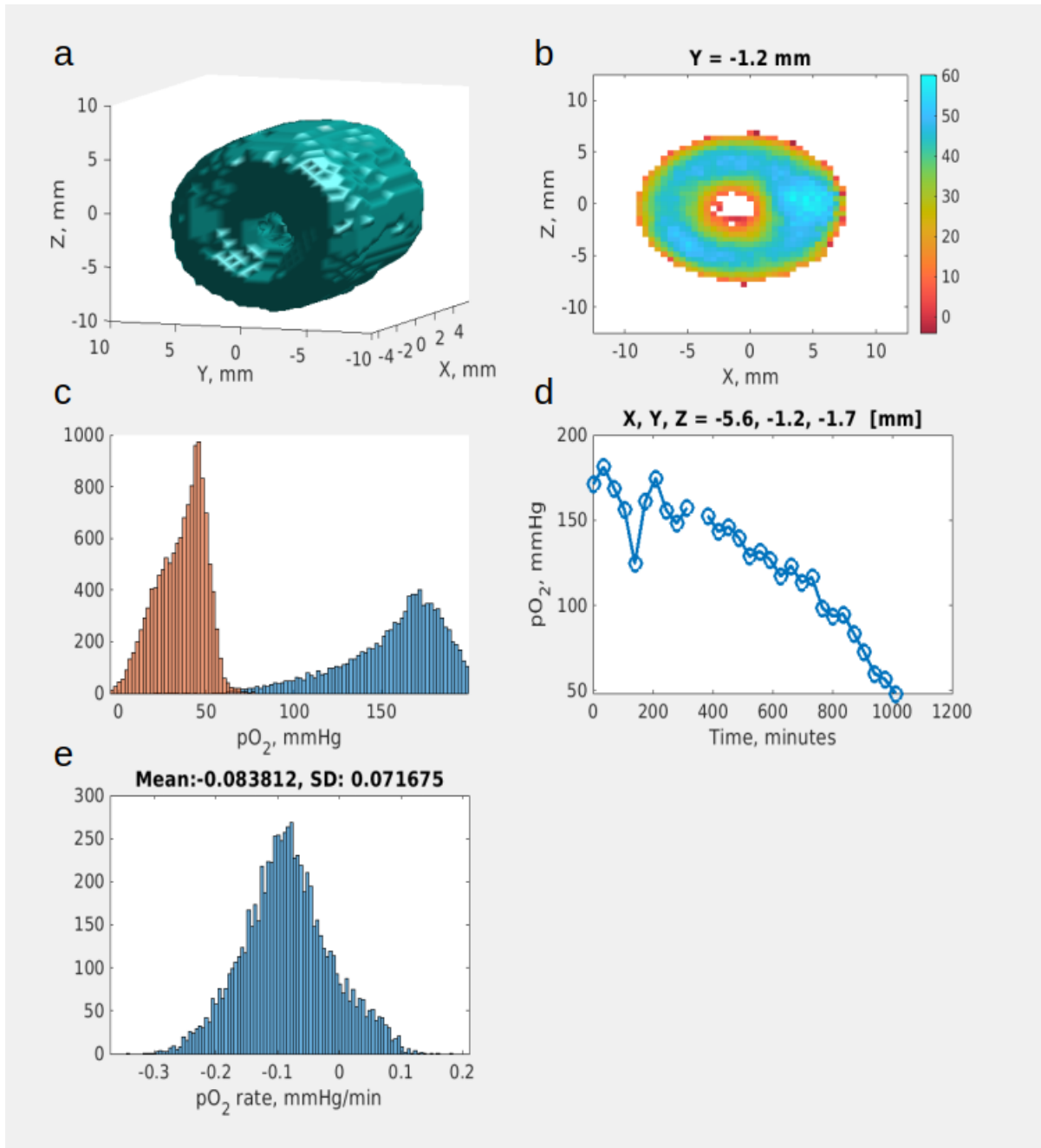
Figure 4.20: EPRI kinetics of murine macrophages in a bioprint. Murine macrophages (25,000 cells/mL) were incorporated into a bioprint as outlined in the Methods section. 30 images were collected over a period of approximately 16.7 hours. a: surface model of reconstructed EPR image. b: representative slice of last EPR image in the set of images. Color bar shows $pO_2$ in mmHg. c: histogram of first and last image showing distribution of oxygen partial pressure across all voxels in the image. d: representative oxygen partial pressure at one voxel across time. e: histogram of oxygen consumption rate calculated at each voxel.

| Source | SS | df | MS | F | Prob>F |
|--------|-----|-----|-----|-----|--------|
| Groups | 6.29359e-15 | 3 | 2.09786e-15 | 10.05 | 0.0014 |
| Error | 2.50375e-15 | 12 | 2.08646e-15 | | |
| Total | 8.79734e-15 | 15 | | | |

Table 4.1: Results of 1-way ANOVA for stirring experiment (see figure 4.7). Calculated using anova1() function in MATLAB.

| Group 1 | Group 2 | P-value |
|---------|---------|---------|
| 1% Pyruvate | 0.4% Pyruvate | 0.4490 |
| 1% Pyruvate | 0.2% Pyruvate | 0.0087 |
| 1% Pyruvate | 0.04% Pyruvate | 0.0018 |
| 0.4% Pyruvate | 0.2% Pyruvate | 0.1230 |
| 0.4% Pyruvate | 0.04% Pyruvate | 0.0248 |
| 0.2% Pyruvate | 0.04% Pyruvate | 0.7859 |

Table 4.2: Pairwise comparisons test for stirring experiment (see figure 4.7). MATLAB multcompare used for analysis.

| Source | SS | df | MS | F | Prob>F |
|--------|-----|-----|-----|-----|--------|
| Groups | 4.3e-11 | 4 | 1.1e-11 | 82.03 | 9.0e-33 |
| Error | 1.5e-11 | 115 | 1.3e-13 | | |
| Total | 5.8e-11 | 119 | | | |

Table 4.3: Results of 1-way ANOVA with unequal sample sizes for Seahorse OCR experiment (see figure 4.9). Calculated using anova1() function in MATLAB.

| Group 1 | Group 2 | P-value |
|---------|---------|---------|
| Pyruvate 1% | Pyruvate 0.4% | 0.16 |
| Pyruvate 1% | Pyruvate 0.2% | 2.8e-7 |
| Pyruvate 1% | Pyruvate 0.04% | 5.2e-18 |
| Pyruvate 1% | No Pyruvate | 7.7e-29 |
| Pyruvate 0.4% | Pyruvate 0.2% | 0.0033 |
| Pyruvate 0.4% | Pyruvate 0.04% | 1.5e-12 |
| Pyruvate 0.4% | No Pyruvate | 2.8e-22 |
| Pyruvate 0.2% | Pyruvate 0.04% | 7.0e-5 |
| Pyruvate 0.2% | No Pyruvate | 1.0e-11 |
| Pyruvate 0.04% | No Pyruvate | 0.14 |

Table 4.4: Pairwise comparisons test for Seahorse OCR measurements (see figure 4.9). Calculated using MATLAB multcompare.

| Source | SS | df | MS | F | Prob>F |
|--------|-----|-----|-----|-----|--------|
| Groups | 2.6e-10 | 2 | 1.3e-10 | 225.2 | 1.1e-24 |
| Error | 8.1e-11 | 77 | 1.1e-12 | | |
| Total | 3.4e-10 | 79 | | | |

Table 4.5: Results of 1-way ANOVA with unequal sample sizes for long-term (60 minutes per measurement period) Seahorse OCR experiment (see figure 4.11). Calculated using anova1()function in MATLAB.

| Group 1 | Group 2 | P-value |
|---------|---------|---------|
| Pyruvate 1% | Pyruvate 0.4% | 1.8e-5 |
| Pyruvate 1% | No Pyruvate | 1.5e-24 |
| Pyruvate 0.4% | No Pyruvate | 4.3e-15 |

Table 4.6: Pairwise comparisons test for longer Seahorse OCR measurement (see figure 4.11). Calculated using MATLAB multcompare.

| Method mean (std) | OCR, Pyruvate 1% $10^{-9}$ mmHg/min/cell | OCR, No Pyruvate $10^{-9}$ mmHg/min/cell |
|-------------------|------------------------------------------|------------------------------------------|
| Stirring | 67.2 (14.4) | 17.1 (4.94) |
| Seahorse | 214 (12.9) | 135 (16.1) |
| EPR | 213 (75.8) | 122 (10.3) |
| EPRI | 91.3 (26.2) | 41.8 (8.36) |

Table 4.7: Comparison of OCR methods. OCR a: Yeast starved for 6 hours then given pyruvate 1%. OCR b: Yeast starved for 6 hours, no added nutrients. Units are mmHg per minute per cell. Seahorse OCR was calculated between 9-10 minutes of data collection.

| Source | SS | df | MS | F | P |
|--------|-----|-----|-----|-----|-----|
| Groups | 1.97e9 | 7 | 2.82e8 | 7400 | <1e-5 |
| Error | 1.16e7 | 30800 | 378 | | |
| Total | 3.14e7 | 30800 | | | |

Table 4.8: One-way ANOVA of different OCR methods. Calculated using 1-way ANOVA using summary data (see statpages.info/anova1sm.html).

| Group 1 | Group 2 | P-value |
|---|---|---|
| Stirring, Pyruvate | Stirring, No Pyruvate | 0.0064 |
| Stirring, Pyruvate | Seahorse, Pyruvate | 0 |
| Stirring, Pyruvate | Seahorse, No Pyruvate | 0 |
| Stirring, Pyruvate | EPR, Pyruvate | 0 |
| Stirring, Pyruvate | EPR, No Pyruvate | 0.0017 |
| Stirring, Pyruvate | EPRI, Pyruvate | 0.20 |
| Stirring, Pyruvate | EPRI, No Pyruvate | 0.1514 |
| Stirring, No Pyruvate | Seahorse, Pyruvate | 0 |
| Stirring, No Pyruvate | Seahorse, No Pyruvate | 0 |
| Stirring, No Pyruvate | EPR, Pyruvate | 0 |
| Stirring, No Pyruvate | EPR, No Pyruvate | 0 |
| Stirring, No Pyruvate | EPRI, Pyruvate | 0 |
| Stirring, No Pyruvate | EPRI, No Pyruvate | 0.1787 |
| Seahorse, Pyruvate | Seahorse, No Pyruvate | 0 |
| Seahorse, Pyruvate | EPR, Pyruvate | 1 |
| Seahorse, Pyruvate | EPR, No Pyruvate | 0 |
| Seahorse, Pyruvate | EPRI, Pyruvate | 0 |
| Seahorse, Pyruvate | EPRI, No Pyruvate | 0 |
| Seahorse, No Pyruvate | EPR, Pyruvate | 0 |
| Seahorse, No Pyruvate | EPR, No Pyruvate | 0.9003 |
| Seahorse, No Pyruvate | EPRI, Pyruvate | 0 |
| Seahorse, No Pyruvate | EPRI, No Pyruvate | 0 |
| EPR, Pyruvate | EPR, No Pyruvate | 0 |
| EPR, Pyruvate | EPRI, Pyruvate | 0 |
| EPR, Pyruvate | EPRI, No Pyruvate | 0 |
| EPR, No Pyruvate | EPRI, Pyruvate | 0.034 |
| EPR, No Pyruvate | EPRI, No Pyruvate | 0 |
| EPRI, Pyruvate | EPRI, No Pyruvate | 0 |

Table 4.9: Tukey HSD Post-hoc Test of different OCR methods. Calculated using summary data (see statpages.info/anova1sm.html).

| Source | SS | df | MS | F | Prob>F |
|---|---|---|---|---|---|
| Groups | 1.06441e-7 | 3 | 3.54804e-8 | 29.36 | 3.37401e-14 |
| Error | 1.40165e-7 | 116 | 1.20832e-9 | | |
| Total | 2.46606e07 | 119 | | | |

Table 4.10: Results of 1-way ANOVA for Seahorse OCR measurement of macrophages (see figure 4.19). Calculated using MATLAB anova1() function.

| Group 1 | Group 2 | P-value |
|---|---|---|
| 25k macrophages | 50k macrophages | 7.5e-6 |
| 25k macrophages | 100k macrophages | 1.6e-13 |
| 25k macrophages | 250k macrophages | 1.2e-10 |
| 50k macrophages | 100k macrophages | 0.003 |
| 50k macrophages | 250k macrophages | 0.10 |
| 100k macrophages | 250k macrophages | 0.59 |

Table 4.11: Pairwise comparisons test for macrophage OCR measurement (see figure 4.19). Calculated using MATLAB multcompare() function.

# Chapter 5

# Discussion

Completion of the dissertation research resulted in several publications, including chapters 2 and 3. Chapter 4 is in preparation for publication. However, several other publications were a result of the dissertation research. Tseytlin et al (2019) describes the original modular rapid scan EPR imager, during which the original ideas for digitization and automation of data acquisition were developed.[1] Next, Mihalik et al (2023) collected *in vivo* oxygenation maps to compare tissue oxygenation before and after low- or high-dose GMCSF treatment using the ADiC EPRI system.[2] The ADiC system allowed faster and easier data collection, giving more time for imaging rather than fine-tuning the machine. Finally, Sarvari et al (2023) measured oxygen depletion by photoinitiator in light-based bioprints, and imaged oxygen production by algae-laden bioprints using the ADiC EPRI system for longitudinal oxygen mapping.[3]

## 5.1 Digitization of the RS EPRI system permits automation and longitudinal imaging

During the construction of our modular rapid (RS) electron paramagnetic resonance (EPR) imaging system,[1] there were several limitations that could be overcome with a digital acquisition system. The time required to tune and couple the resonator by manually adjusting voltages of the resonator's voltage controlled capacitors was on the time scale of several seconds to minutes. Images had to be collected one-by-one, repeating the tuning and coupling process between each image. For long experiments to examine changes over time in a variable, it was required to have one or two personnel monitoring the

EPR system at all times. Additionally, we occasionally noticed large periodic amplitude modulated noise which decreased the signal to noise ratio (SNR) of the imager, which may have been a result of the EPR imager being sensitive to radiofrequencies picked up by the wires between the power supply and the voltage controlled capacitors. Finally, during image collection, no adjustments were able to be made, sometimes resulting in the loss of data from changes in coupling and tuning from animal movement, through drift of magnetic field, or as conditions in the environment changed. These limitations resulted in the loss of much time and less potential data collection.

The objective of digitization of the RS EPRI system through the development of the automated digital control (ADiC) and digital resonator was to overcome the above limitations in Chapter 2. Through the use of digitally tunable capacitors (DTCs) and a printed circuit board (PCB)-based resonator, automated control of coupling was enabled. Replacing the reference microwave RF source with a digital frequency source (DFS) enabled automated tuning. The DTCs that were used (PE64906) allow 32 discrete capacitances each, for a combined 94 possible values if i $\pm$ 1 index values are used. Controlling the ADiC is a Teensy microcontroller (MC). During the coupling cycle, the MC changes the DTC value, sets frequency scans with an arbitrary waveform generator, then measures reflection from the resonator at the rate of 300 measurements per millisecond. MC selects the best DTC value to match resonator impedance. By selecting the best measurement point from the frequency scan, then using a point-to-frequency calibration, the tuning cycle selects the best RF frequency to match the resonance frequency. This process takes place generally within 10ms, dependent on how many DTC indices are tried before optimal coupling is found. This automated tuning and coupling is much faster than manual tuning and coupling, and is very stable - 3000 EPR spectra were collected in Fig. 2.6 over a period of 24 hours without a single failed tuning/coupling cycle. Finally, by introducing a break between image projection collections, tuning/coupling can take place during image collection, as in Fig. 2.7. Here, projections are arranged in blocks, each containing 126 projections. Between each block, magnetic field drift is corrected and tuning/coupling are optimized.

In other EPRI systems, an analog tuning correction is used, called automatic frequency control (AFC). In general terms, AFC is performed by a detector which generates a voltage dependent on the distance between the target frequency (resonance frequency) and the current RF.[4] This voltage induces adjustment of the RF. In continuous wave (CW) EPR, phase detection of the frequency modulation is used to bring RF to resonance frequency.[5] Alternatively, the resonance frequency can be adjusted to

match the RF using varactor diodes in the position of the tuning capacitor.[6] Varactor diodes are electrical components that can be used as voltage controlled capacitors. With their addition to the resonator, automatic coupling can be achieved in conjunction with AFC.[7] AFC is limited to small corrections of frequency[8] and induces change in position of the EPR signal, causing distortion of the EPR spectra or projections.[6] Due to AFC running continuously, this EPR signal distortion results in misleading data (artificial linewidth broadening, etc) or the loss of data. In comparison, ADiC has a global (20MHz in our case) scan for resonance frequency. ADiC only runs when data is not being collected, avoiding distortion of the collected signal.

Integration of digital components into the EPR system is a recent direction in the EPR field.[9] Digital components enable miniaturization, increase repeatability through discretization, and enable automation. Direct digital detection of EPR signal through the use of a digitizer is common in CW, RS, and pulse EPR, and can be used to increase SNR, among other improvements.[10] Digital components such as DFS and DTCs can be used to generate RF and tune/couple the resonator (Figure 2.2). These components are controlled with a digital MC. Alternatively, a digital controller such as MC or a field programmable gate array (FPGA) can control analog components (voltage controlled oscillator, VCO, and voltage controlled capacitors) through the conversion of digital signal to analog voltage using a digital-to-analog converter (DAC).[11] This flexibility of the digital controllers allows for a wider selection of components for use in the design of the EPR system.

Another benefit of digital components is the possibilities of miniaturization. EPR-on-a-chip designs promise to put all the components of an EPR spectrometer on a single PCB, for application in high-frequency EPR with small samples and high spectral resolution.[12,13] Miniature EPR systems also have *in vivo* applications in biodosimetry[14] and cutaneous oxygen measurement.[15] Our own resonator was reduced in size by several centimeters due to the addition of a balun chip replacing a more traditional coaxial transmission line balun (Figure 2.2). Placing the entire EPR RF bridge and resonator on a single PCB would reduce space requirements, cost, and losses due to long transmission lines.

One future direction for further development of an ADiC is finer selection of DTC indices. Close to critical coupling, small capacitance changes give large changes in coupling. In the current ADiC system, two DTCs with maximum difference of index $\pm$ 1 gives 94 possible combinations to select critical coupling. A higher number of possible combinations in the same capacitance range would allow the MC to select coupling closer to true critical coupling of the resonator, resulting in better SNR. As

digital components continue to improve, it is likely that DTCs with more possible combinations will become available. An alternative to waiting would be to design the circuit to allow four capacitors arranged in two pairs. One capacitor would give rough capacitance adjustments while the second gives finer adjustment.

A second limiting factor to the amount of DTC combinations tried during the coupling cycle is the time required for the completion of the coupling cycle itself. In the current ADiC system, each combination requires the MC to collect 300 points as an AWG scans resonance frequency. This cycle takes 1ms for each DTC combination due to the maximum data collection rate of the Teensy 4.1 MC.Improvement in data collection rate creates the possibility of trying more DTC combinations in a given time. FPGAs and application specific integrated circuits (ASICs) allow the use of parallel processing and faster processing speeds. These devices also can be used for direct detection of EPR signal and control of magnetic field strength when combined with fast analog-to-digital converters (ADCs).[16] As a result, the ability of the FPGA or MC to tune, couple and collect data automatically through digital protocols gives the scientist a powerful tool to collect EPR spectra or images without any necessary input from the user. This ability is demonstrated Figures 2.5, 2.6 and 2.7. Finally, EPR data can be stored on a secure digital (SD) card. In sum, the capabilities of MCs or FPGAs almost entirely remove the necessity of a computer for operation of the EPR system.

ADiC can be used to couple and tune the system after large movement from an animal. However, periodic animal movements such as breathing and heartbeat also have effects on coupling and tuning depending on the part of the animal being studied. The rate of mouse breathing under normal conditions is around 255 breaths per minute.[17] This rate is significantly slower under anesthesia, commonly used with EPRI. Mouse heart rate is much faster, around 600 beats per minute.[18] The time for ADiC tuning and coupling cycle along with a cycle of data collection would optimally be a magnitude faster than the breathing or heart rates to account for changes in coupling and tuning these movements cause for *in vivo* experiments. To achieve this, both ADiC and data acquisition would need to be faster than the current system. As mentioned above, using parallel processing with an FPGA or ASIC would allow for faster tuning and coupling cycles. With our current system, a block of projections with enough averaging for good SNR *in vivo* takes between 30 seconds to 2 minutes, corresponding to 1000 and 4000 averages per projection, respectively, for 200µM OX071 (Figure 2.7). Decreasing this time to a scale suitable for correction of resonator conditions for heartbeat and breathing (∼10ms) could be achieved with several

improvements. First, faster data acquisition while maintaining good SNR will require a resonator with higher q-factor and higher RF power. Secondly, several sets of projections can be acquired with low averaging with maintenance of critically coupled conditions (CCC). These repeated projections are later averaged together in the software during image reconstruction. Finer coupling through the use of more DTC indices could result in better SNR, enabling faster data collection through less averaging, though it comes at the cost of more time spent achieving CCC.

In summary, the ADiC system established fast digital coupling, tuning, and data acquisition for RS EPRI. This system overcomes some of the problems associated with traditional AFC and automatic coupling such as distortion of the EPR signal during data collection. Future improvements in ADiC fine coupling and speed can be achieved through parallel processing and upgraded DTCs. The ADiC system is also compatible with other EPR methods such as pulse and CW, though it is optimized for RS EPR.

## 5.2    Oxygen mapping as a tool in 3D printing technology development

In the last decade, 3D printing has rapidly become a useful technique for rapid prototyping and the production of specialized components. One of the more widely available 3D printing methods is light-based printing, where a resin is photopolymerized with light to form a solid structure (see Section 1.7). This photopolymerization process consumes oxygen, a process that can be observed with EPRI through the incorporation of EPR oxygen-sensitive probes in the resin. The original goal of Chapter 3 was to demonstrate a proof-of-concept of the use of EPRI in mapping oxygen concentration in the 3D printing process.

To demonstrate oxygen mapping in 3D prints, lithium n-butoxynaphthalocyanine (LiNc-BuO) was incorporated into resin (Figure 3.2). Oxygen maps were collected of the resin - LiNc-BuO mixture during photopolymerization demonstrating depletion of oxygen at the polymerization front (Figures 3.3, 3.4 and 3.5). Further oxygen depletion was shown as a result of the post-printing process (Figure 3.6). As a proof of concept for imaging printed objects, a 3D printer was modified for use with the LiNc-BuO resin mixture, and an oxygen map was collected for a 3D printed spiral model (Figures 3.1 and 3.2).

One limitation to the use of EPR oxygen mapping in commercial 3D printers is that most of the printers have large resin tanks, requiring a large amount of EPR probe. For example, FormLabs resin comes in 1L containers, requiring 32g of EPR probe if used at the same concentration as Figure 3.7.

This amount of EPR oxygen probe would currently require significant time to synthesize or cost to buy. The solution we utilized was to use a home-built printer which required less volume of resin to operate (Figure 3.1).

Though there is not much previous research mapping oxygen in 3D prints, differential oxygen concentrations are used in 3D printing to prevent the print from sticking to the resin tank.[19] The resin tank is coated in an oxygen-permeable silicone layer that creates a "dead zone" layer preventing photopolymerization directly on the surface of the resin tank. The photopolymerization instead occurs on the build plate. Oxygen mapping using EPR brings the possibility of optimizing this "dead zone" to improve 3D printing speed and quality. To do this, high resolution oxygen EPRI with voxels on the scale of ~10µm will be necessary, as the "dead zone" layer is around 20µm thick.

Another area where oxygen may be important is post-curing. Post-curing of methacrylated resins has been shown to reduce the effects of aging on 3D prints through increased chain crosslinking.[20] Excessive post-curing was shown to result in brittleness. We were able to measure a decrease in oxygen contained in the printed models during the postcuring process, Figure 3.6. By linking oxygen concentration to the amount of post-print curing, optimal conditions may be found for the curing process.

## 5.3 EPRI oxygen mapping, cell consumption rates, and other applications towards bioprinting

EPRI oxygen mapping in 3D culture is proposed as a tool to observe the microenvironment in these models. Observing the oxygen consumption rate (OCR) of cells is an important tool for traditional 2D cell culture in today's research.[21,22] We believe the ability to observe or even map OCR in 3D cultures will be useful. Beyond this application, mapping oxygen concentrations in 3D cultures, particularly bioprinted models, will help on the path towards developing more complex and large *in vitro* 3D tissue and organ models.[23] One unmet need in bioprinting will be to quantify the success of methods to improve delivery of oxygen to cells deep within thick bioprints. Depending on the concentration of cells in culture, the limit of diffusion is between 40-350µm.[24,25] Improving oxygen levels in bioprints through perfused vasculature,[26] oxygen-generating cells,[27] oxygen-generating chemicals,[28] or a combination[29] promise to overcome this challenge. A previous study has measured oxygen concentration in a bioprint using a fiber optic probe, but was limited by the probe to single-point measurements that are invasive

to the model.[30] Another study incorporated oxygen-sensitive nanoparticles into a bioprint.[31] However, microscope depth penetration is limited to ~3mm.

We have recently conducted EPRI oxygen mapping in 3D culture using several types of cells (see Chapter 4). Using home-built bioprinters, it was possible to add EPR oxygen-sensitive probes to the bioink, and create 3D models visible in EPRI. The printers were tested with the MDA-MB-231 model of triple-negative breast cancer. Both 6 and 24 hours after bioprinting, cells remained alive and were able to be imaged under a confocal microscope. By combining images together across the time dimension, 3D OCR was extracted. OCR was found to be different between two bioprints containing yeast, one with pyruvate and one without. These results are consistent with other experiments: 1) fiber optic oxygen probe measurement of yeast stirred in solutions containing varying concentrations of pyruvate, 2) Seahorse measurements of yeast with varying concentrations of pyruvate. Moving towards mammalian cells, OCR of peritoneal macrophages was measured using EPRI. Finally, a computational model of diffusion and oxygen consumption in a spherical coordinate system was generated.

The measurement of OCR in 3D bioprinting using EPRI has exciting potential. However, there are several aspects which can be improved. First, the resolution of the EPR images in the study was approximately 250μm. As mentioned above, depending on cell concentration and diffusion coefficient, the limit of diffusion is between 40-350 μm, about the same as the EPR image resolution. As a result, the oxygen gradient caused by competing diffusion and oxygen consumption by cells would not be seen at this scale. Rather, an average of the oxygen concentration over the volume of the voxel would be observed. Luckily, this average would still be informative, and most vasculature models currently are on the scale of several hundreds of micrometers to millimeters in vasculature diameter, with spacing of several millimeters between vessels. The resolution can be improved, however, through either the use of higher imaging gradients (gradients of 3G were used here, though much higher gradients are possible on our machine), or through the development of an EPR probe with narrower linewidth. Additionally, the computational model can be used in reverse to estimate the oxygen gradient present with a given cell concentration and diffusion coefficient.

Next, each image took between 18-30 minutes to acquire, depending on the collection of 16 blocks of projections for fast kinetic images or the full 26 blocks of projections. Large changes in oxygen concentration faster than this time period will not be recorded, and could perhaps cause distortion in the EPR images. Each projection in these images were collected with 2000 averages. By increasing

the concentration of EPR probe, by increasing the sensitivity of the EPRI system or better coupling of the resonator, less averages will be required resulting in faster image collection. Increasing SNR using the methods described here would also lead to better precision in measured oxygen concentrations. For example, an amount of error or random noise was present in the single voxel measurements of some points in Figures 4.16 d and 4.20 d (between 400-600 minutes and near 200 minutes, respectively). Thankfully, due to the large number of voxels in the images ($50^3$-$70^3$), these single noisy or incorrect points do not affect the overall conclusions.

In the computational simulation, OCR was in the same magnitude as experimental data. However, it was apparent that the number of divisions in distance, x, had a large (~50%) effect on OCR. It is clear that in future experiments, the maximum possible number of divisions, N, will be required. However, as N increases, the time required for computation increases exponentially, because the number of iterations, i, for computing change in oxygen also increases exponentially. In the study, $i = \frac{D}{dx^2}$, where D is diffusion coefficient and dx is computed from N and cell concentration. Additionally, computational time increases even more because of computing oxygen concentrations at N number of positions each iteration. The example used in the study was that an N of 1,000 would take approximately 9 days while an N of 10,000 would take 1 million days for the simulation to compute. At some point, the simulated OCR should converge as N increases, but more research is required to determine when this occurs.

It was noticed that the oxygen concentration measurements of oxygen consumption by cells were more nonlinear in Seahorse data compared to the stirred experiment (Fig. 4.7 and Appendix) or even 3D EPR oxygen mapping. This could be due to oxygen diffusion from the external environment during Seahorse measurement, as well as the absence of stirring during measurement. In the Seahorse experiment, all the cells are at the bottom of the wells. As oxygen is consumed at the bottom of the wells, an oxygen gradient forms. As a result of the heat equation (see equation 4.1), the larger the gradient, the faster oxygen diffuses towards the cells. Therefore, an equilibrium is met: as oxygen is consumed by cells, more oxygen is delivered through diffusion, resulting in the nonlinearity of the Seahorse measurements. Nonlinearity in oxygen concentration over time leads to error in measuring maximum OCR. However, we measured Seahorse data over much longer time periods before mixing than is typically seen in the literature, which would minimize this effect. One note here: slight nonlinearity was also noticed in the simulation data as oxygen concentrations neared zero. This is expected, because in the cell where oxygen is consumed, a finite number of oxygen molecules exist. As oxygen concentrations near zero, the

cell in the simulation consumes all of the oxygen molecules available and has to wait for more oxygen to diffuse into the cell. This is the diffusion-limited condition.

As mentioned above, future studies will apply EPRI oxygen mapping to models of oxygen delivery to cells in the bioprint. Research in the lab is already moving in this direction, with the construction of a perfusable resonator for EPRI. Additionally, we have conducted research on measuring oxygen production by algae in bioprinted constructs (paper in press). Measurement of continued oxygen consumption in bioprints could be a way to determine cell viability in deep tissue noninvasively. Another future direction will be the expansion of spectral-spatial-temporal (5D) EPRI measurements into other aspects of the microenvironment. Many different EPR probes are available, measuring reactive oxygen species, pH, phosphate, viscosity, and enzymatic activity. Each of these measurements have potential applications in 3D cultures. For example, mechanical properties of the cell environment, including viscosity, are important in stem cell behavior and differentiation.[32] Bioprinted tumor models should recapitulate aspects of the microenvironment seen in patients: low pH and low oxygen concentration.[33,34]

The original goal of measuring OCR in yeast in this study was to measure the ratio of aerobic to anaerobic respiration. However, yeast have complex metabolism like most other cells, and this ratio cannot be measured simply through consumption of oxygen. Additional ability to measure the consumption of nutrients such as glucose and production of $CO_2$ would be required. Glucose consumption can be readily measured using positron emission tomography (PET),[35] and $CO_2$ production could perhaps be estimated by measuring pH, though this has not been tried to the writer's knowledge. Fortunately, a PET/EPR combined imager is in the final phases of construction, and would allow for glucose, oxygen, and pH coimaging.

Moving towards *in vivo* studies, mapping kinetics using EPRI will have applications in animal studies as well. Mapping changes in signal intensity over time would enable the measurement of the length of time from EPR probe or medicine administration to arrival at the target organ. Certain treatments for disease models can be expected to change the microenvironment as well.[36] For example, Vandetanib increases tumor oxygenation not related to changes in vasculature or perfusion.[37] Mapping the rate of oxygen increase over time could give further insights to dosing and optimal timing for radiotherapy.

## 5.4 Final Thoughts

In this research, our modular RS EPRI system was upgraded with ADiC, a modular, automatic digital control of coupling, tuning, and data acquisition. ADiC is fast (<15ms), repeatable (no mistakes over 24 hour periods), and provides wide range coupling and tuning. Automation became possible through the use of an Arduino, and digital components including a PCB resonator, DTCs and DFS. ADiC should make imaging of small animals faster and more repeatable, but also enables the measurement of kinetics by combining EPR images in the time dimension. Although ADiC is fast, it can be made much faster using an FPGA or ASIC with parallel processing. Additional DTC values would allow for finer coupling resulting in better EPR SNR. Potentially the entire EPR RF bridge and resonator could be combined into one PCB, controlled by an ASIC also located on the PCB. Such future directions are also a goal of the EPR-on-a-chip project.

Next, oxygen concentration was mapped in 3D printing resins and 3D printed constructs. Oxygen is a crucial inhibitor of photopolymerization of light-based printing. Measurement and mapping of oxygen in 3D printed constructs can lead to better optimized 3D printing systems and quality control for 3D prints. Degradation of the 3D print over time due to oxidation of the photocured resin may also be studied with EPRI.

Finally, oxygen kinetics were mapped in several bioprinted constructs made with alginate bioink containing EPR oxygen sensitive probe. OCR was extracted from EPR image data. Importantly, difference in OCR between pyruvate-fed and starved yeast was evident. To better understand diffusion and oxygen consumption in 3D culture at the cellular level, a computational model of yeast diffusion and consumption was developed using the heat equation in a spherical coordinate system. The ability to map oxygen levels and OCR in 3D culture can be used to quantify improvements in methods to deliver oxygen to cells deep inside 3D tissue models such as vasculature, chemical oxygen generation, and cell-based oxygen generation. Mapping OCR with EPR can also be used in much the same way as in 2D culture, to determine the respiration of different cell types in models of disease. It would be interesting to see if oxygen consumption rates for cells is similar or the same in 2D compared to 3D culture. Future efforts can be directed towards these goals.

# References

[1] Tseytlin O, Guggilapu P, Bobko AA, AlAhmad H, Xu X, Epel B, O'Connell R, Hoblitzell EH, Eubank TD, Khramtsov VV, Driesschaert B, Kazkaz E, and Tseytlin M (2019) Modular imaging system: Rapid scan EPR at 800 MHz. Journal of Magnetic Resonance 305:94–103. ISSN 10907807. doi:10.1016/j.jmr.2019.06.003. URL https://linkinghub.elsevier.com/retrieve/pii/S1090780719301107.

[2] Mihalik NE, Steinberger KJ, Stevens AM, Bobko AA, Hoblitzell EH, Tseytlin O, Akhter H, Dziadowicz SA, Wang L, O'Connell RC, Monaghan KL, Hu G, Mo X, Khramtsov VV, Tseytlin M, Driesschaert B, Wan ECK, and Eubank TD (2023) Dose-Specific Intratumoral GM-CSF Modulates Breast Tumor Oxygenation and Antitumor Immunity. The Journal of Immunology 211(10):1589–1604. ISSN 0022-1767, 1550-6606. doi:10.4049/jimmunol.2300326. URL https://journals.aai.org/jimmunol/article/211/10/1589/265942/Dose-Specific-Intratumoral-GM-CSF-Modulates-Breast.

[3] Sarvari S, McGee D, O'Connell R, Tseytlin O, Bobko AA, and Tseytlin M (2023) Electron spin resonance probe incorporation into bioink permits longitudinal oxygen imaging of bioprinted constructs. Molecular Imaging and Biology doi:10.1007/s11307-023-01871-0.

[4] Travis C (1935) Automatic Frequency Control. Proceedings of the IRE 23(10):1125–1141. ISSN 0096-8390. doi:10.1109/JRPROC.1935.227257. URL http://ieeexplore.ieee.org/document/1685769/.

[5] Hyde JS and Gajdzinski J (1988) EPR automatic frequency control circuit with field effect transistor (FET) microwave amplification. Review of Scientific Instruments 59(8):1352–1356. ISSN 0034-6748, 1089-7623. doi:10.1063/1.1139721. URL https://pubs.aip.org/rsi/article/59/8/1352/924935/EPR-automatic-frequency-control-circuit-with-field.

[6] Hirata H, Walczak T, and Swartz HM (2000) Electronically Tunable Surface-Coil-Type Resonator for L-Band EPR Spectroscopy. Journal of Magnetic Resonance 142(1):159–167. ISSN 10907807. doi:10.1006/jmre.1999.1927. URL https://linkinghub.elsevier.com/retrieve/pii/S1090780799919274.

[7] Yokoyama H, Sato T, Ogata T, Ohya H, and Kamada H (2001) Automatic Coupling Control of a Loop–Gap Resonator by a Variable Capacitor Attached Coupling Coil for EPR Measurements at 650 MHz. Journal of Magnetic Resonance 149(1):29–35. ISSN 10907807. doi:10.1006/jmre.2000.2268. URL https://linkinghub.elsevier.com/retrieve/pii/S1090780700922687.

[8] Hirata H and Luo ZW (2001) Stability analysis and design of automatic frequency control system for in vivo EPR spectroscopy. Magnetic Resonance in Medicine 46(6):1209–1215. ISSN 0740-3194, 1522-2594. doi:10.1002/mrm.1318. URL https://onlinelibrary.wiley.com/doi/10.1002/mrm.1318.

[9] Hyde JS (2005) Trends in EPR Technology. In Eaton SR, Eaton GR, and Berliner LJ, editors, *Biomedical EPR, Part B: Methodology, Instrumentation, and Dynamics*, volume 24/B, 409–428. Kluwer Academic Publishers-Plenum Publishers, New York. ISBN 978-0-306-48532-9. doi: 10.1007/0-306-48533-8_13. URL http://link.springer.com/10.1007/0-306-48533-8_13. Series Title: Biological Magnetic Resonance.

[10] Tseitlin M, Quine RW, Rinard GA, Eaton SS, and Eaton GR (2011) Digital EPR with an arbitrary waveform generator and direct detection at the carrier frequency. Journal of Magnetic Resonance S1090780711003363. ISSN 10907807. doi:10.1016/j.jmr.2011.09.024. URL https://linkinghub.elsevier.com/retrieve/pii/S1090780711003363.

[11] Sato-Akaba H, Emoto MC, Hirata H, and Fujii HG (2017) Design and testing of a 750 MHz CW-EPR digital console for small animal imaging. Journal of Magnetic Resonance 284:48–58. ISSN 10907807. doi:10.1016/j.jmr.2017.09.008. URL https://linkinghub.elsevier.com/retrieve/pii/S109078071730232X.

[12] Chu A, Kern M, Khan K, LiPS K, and Anders J (2023) A 263GHz 32-Channel EPR-on-a-Chip Injection-Locked VCO-Array. In *2023 IEEE International Solid- State Circuits Conference (ISSCC)*, 20–22. IEEE, San Francisco, CA, USA. ISBN 978-1-66549-016-0. doi:10.1109/ISSCC42615.2023.10067623. URL https://ieeexplore.ieee.org/document/10067623/.

[13] Jeschke G (2019) Quo vadis EPR? Journal of Magnetic Resonance 306:36–41. ISSN 10907807. doi:10.1016/j.jmr.2019.07.008. URL https://linkinghub.elsevier.com/retrieve/pii/S1090780719301338.

[14] Buchbinder L, Datz H, Dayan N, Carmieli R, and Blank A (2022) The Potential of Pulsed Electron Spin Resonance for Tooth-Based Retrospective Biodosimetry. Applied Magnetic Resonance 53(7-9):989–1014. ISSN 0937-9347, 1613-7507. doi:10.1007/s00723-021-01417-z. URL https://link.springer.com/10.1007/s00723-021-01417-z.

[15] Wolfson H, Ahmad R, Twig Y, Blank A, and Kuppusamy P (2015) A hand-held EPR scanner for transcutaneous oximetry. 941706. Orlando, Florida, United States. doi:10.1117/12.2083864. URL http://proceedings.spiedigitallibrary.org/proceeding.aspx?doi=10.1117/12.2083864.

[16] Sato-Akaba H and Tseytlin M (2019) Development of an L-band rapid scan EPR digital console. Journal of Magnetic Resonance 304:42–52. ISSN 10907807. doi:10.1016/j.jmr.2019.05.003. URL https://linkinghub.elsevier.com/retrieve/pii/S1090780719300874.

[17] Fairchild GA (1972) Measurement of Respiratory Volume for Virus Retention Studies in Mice. Applied Microbiology 24(5):812–818. ISSN 0003-6919. doi:10.1128/am.24.5.812-818.1972. URL https://journals.asm.org/doi/10.1128/am.24.5.812-818.1972.

[18] Mitchell GF, Jeron A, and Koren G (1998) Measurement of heart rate and Q-T interval in the conscious mouse. American Journal of Physiology-Heart and Circulatory Physiology 274(3):H747–H751. ISSN 0363-6135, 1522-1539. doi:10.1152/ajpheart.1998.274.3.H747. URL https://www.physiology.org/doi/10.1152/ajpheart.1998.274.3.H747.

[19] Tumbleston JR, Shirvanyants D, Ermoshkin N, Januszewicz R, Johnson AR, Kelly D, Chen K, Pinschmidt R, Rolland JP, Ermoshkin A, Samulski ET, and DeSimone JM (2015) Continuous liquid interface production of 3D objects. Science 347(6228):1349–1352. ISSN 0036-8075, 1095-9203. doi:10.1126/science.aaa2397. URL https://www.science.org/doi/10.1126/science.aaa2397.

[20] Marin E, Boschetto F, Zanocco M, Doan HN, Sunthar TP, Kinashi K, Iba D, Zhu W, and Pezzotti G (2021) UV-curing and thermal ageing of methacrylated stereo-lithographic resin. Polymer Degradation and Stability 185:109503. ISSN 01413910. doi:10.1016/j.polymdegradstab.2021.109503. URL https://linkinghub.elsevier.com/retrieve/pii/S0141391021000239.

[21] Plitzko B and Loesgen S (2018) Measurement of Oxygen Consumption Rate (OCR) and Extra-cellular Acidification Rate (ECAR) in Culture Cells for Assessment of the Energy Metabolism. BIO-PROTOCOL 8(10). ISSN 2331-8325. doi:10.21769/BioProtoc.2850. URL https://bio-protocol.org/e2850.

[22] Hou J, Wright HJ, Chan N, Tran R, Razorenova OV, Potma EO, and Tromberg BJ (2016) Correlating two-photon excited fluorescence imaging of breast cancer cellular redox state with seahorse flux analysis of normalized cellular oxygen consumption. Journal of Biomedical Optics 21(6):060503. ISSN 1083-3668. doi:10.1117/1.JBO.21.6.060503. URL http://biomedicaloptics.spiedigitallibrary.org/article.aspx?doi=10.1117/1.JBO.21.6.060503.

[23] Juraski AC, Sharma S, Sparanese S, Da Silva VA, Wong J, Laksman Z, Flannigan R, Rohani L, and Willerth SM (2023) 3D bioprinting for organ and organoid models and disease modeling. Expert Opinion on Drug Discovery 18(9):1043–1059. ISSN 1746-0441, 1746-045X. doi:10.1080/17460441.2023.2234280. URL https://www.tandfonline.com/doi/full/10.1080/17460441.2023.2234280.

[24] Sekine W and Haraguchi Y (2011) Thickness limitation and cell viability of multi-layered cell sheets and overcoming the diffusion limit by a porous-membrane culture insert. Journal of Biochips & Tissue Chips 01(01). ISSN 21530777. doi:10.4172/2153-0777.S1-007. URL https://www.omicsonline.org/thickness-limitation-and-cell-viability-of-multilayered-cell-sheets-and-overcoming-the-di:php?aid=9796.

[25] Fritschen A, Acedo Mestre M, Scholpp S, and Blaeser A (2023) Influence of the physico-chemical bioink composition on the printability and cell biological properties in 3D-bioprinting of a liver tumor cell line. Frontiers in Bioengineering and Biotechnology 11:1093101. ISSN 2296-4185. doi:10.3389/fbioe.2023.1093101. URL https://www.frontiersin.org/articles/10.3389/fbioe.2023.1093101/full.

[26] Fang Y, Guo Y, Wu B, Liu Z, Ye M, Xu Y, Ji M, Chen L, Lu B, Nie K, Wang Z, Luo J, Zhang T, Sun W, and Xiong Z (2023) Expanding Embedded 3D Bioprinting Capability for Engineering Complex Organs with Freeform Vascular Networks. Advanced Materials 35(22):2205082. ISSN

0935-9648, 1521-4095. doi:10.1002/adma.202205082. URL https://onlinelibrary.wiley.com/doi/10.1002/adma.202205082.

[27] Ortega JS, Corrales-Orovio R, Ralph P, Egaña JT, and Gentile C (2023) Photosynthetic microorganisms for the oxygenation of advanced 3D bioprinted tissues. Acta Biomaterialia 165:180–196. ISSN 17427061. doi:10.1016/j.actbio.2022.05.009. URL https://linkinghub.elsevier.com/retrieve/pii/S1742706122002781.

[28] Augustine R, Gezek M, Bostanci NS, Nguyen A, and Camci-Unal G (2023) Oxygen-generating scaffolds: One step closer to the clinical translation of tissue engineered products. Chemical Engineering Journal 455:140783. ISSN 13858947. doi:10.1016/j.cej.2022.140783. URL https://linkinghub.elsevier.com/retrieve/pii/S1385894722062635.

[29] Maharjan S, Alva J, Cámara C, Rubio AG, Hernández D, Delavaux C, Correa E, Romo MD, Bonilla D, Santiago ML, Li W, Cheng F, Ying G, and Zhang YS (2021) Symbiotic Photosynthetic Oxygenation within 3D-Bioprinted Vascularized Tissues. Matter 4(1):217–240. ISSN 25902385. doi:10.1016/j.matt.2020.10.022. URL https://linkinghub.elsevier.com/retrieve/pii/S2590238520305762.

[30] Figueiredo L, Le Visage C, Weiss P, and Yang J (2020) Quantifying Oxygen Levels in 3D Bioprinted Cell-Laden Thick Constructs with Perfusable Microchannel Networks. Polymers 12(6):1260. ISSN 2073-4360. doi:10.3390/polym12061260. URL https://www.mdpi.com/2073-4360/12/6/1260.

[31] Trampe E, Koren K, Akkineni AR, Senwitz C, Krujatz F, Lode A, Gelinsky M, and Kühl M (2018) Functionalized Bioink with Optical Sensor Nanoparticles for $O_2$ Imaging in 3D-Bioprinted Constructs. Advanced Functional Materials 28(45):1804411. ISSN 1616301X. doi:10.1002/adfm.201804411. URL https://onlinelibrary.wiley.com/doi/10.1002/adfm.201804411.

[32] Vining KH and Mooney DJ (2017) Mechanical forces direct stem cell behaviour in development and regeneration. Nature Reviews Molecular Cell Biology 18(12):728–742. ISSN 1471-0072, 1471-0080. doi:10.1038/nrm.2017.108. URL https://www.nature.com/articles/nrm.2017.108.

[33] Dutta SD, Ganguly K, Hexiu J, Randhawa A, Moniruzzaman M, and Lim K (2023) A 3D

Bioprinted Nanoengineered Hydrogel with Photoactivated Drug Delivery for Tumor Apoptosis and Simultaneous Bone Regeneration via Macrophage Immunomodulation. Macromolecular Bioscience 2300096. ISSN 1616-5187, 1616-5195. doi:10.1002/mabi.202300096. URL https://onlinelibrary.wiley.com/doi/10.1002/mabi.202300096.

[34] Mao S, Pang Y, Liu T, Shao Y, He J, Yang H, Mao Y, and Sun W (2020) Bioprinting of *in vitro* tumor models for personalized cancer treatment: a review. Biofabrication 12(4):042001. ISSN 1758-5090. doi:10.1088/1758-5090/ab97c0. URL https://iopscience.iop.org/article/10.1088/1758-5090/ab97c0.

[35] Juhasz C, Chugani DC, Muzik O, Shah A, Shah J, Watson C, Canady A, and Chugani HT (2001) Relationship of flumazenil and glucose PET abnormalities to neocortical epilepsy surgery outcome. Neurology 56(12):1650–1658. ISSN 0028-3878, 1526-632X. doi:10.1212/WNL.56.12.1650. URL https://www.neurology.org/lookup/doi/10.1212/WNL.56.12.1650.

[36] Danhier P, De Saedeleer CJ, Karroum O, De Preter G, Porporato PE, Jordan BF, Gallez B, and Sonveaux P (2013) Optimization of Tumor Radiotherapy With Modulators of Cell Metabolism: Toward Clinical Applications. Seminars in Radiation Oncology 23(4):262–272. ISSN 10534296. doi:10.1016/j.semradonc.2013.05.008. URL https://linkinghub.elsevier.com/retrieve/pii/S1053429613000490.

[37] Ansiaux R, Dewever J, Grégoire V, Feron O, Jordan BF, and Gallez B (2009) Decrease in Tumor Cell Oxygen Consumption after Treatment with Vandetanib (ZACTIMA™; ZD6474) and its Effect on Response to Radiotherapy. Radiation Research 172(5):584–591. ISSN 0033-7587, 1938-5404. doi:10.1667/RR1744.1. URL http://www.bioone.org/doi/10.1667/RR1744.1.

# Chapter 6

# Appendix

NOTE: code will not run "as is", the code was changed to fit within the size of the page. Ask author for original code files.

## 6.1 EPRI GUI in MATLAB Appdesigner

```
classdef August_EPRI_2023 < matlab.apps.AppBase


    % Properties that correspond to app components
    properties (Access = public)
        UIFigure                    matlab.ui.Figure
        ButtonGroup_4               matlab.ui.container.ButtonGroup
        DButton                     matlab.ui.control.RadioButton
        OButton                     matlab.ui.control.RadioButton
        SButton                     matlab.ui.control.RadioButton
        RButton                     matlab.ui.control.RadioButton
        MButton                     matlab.ui.control.RadioButton
        AutocenterOffButton         matlab.ui.control.Button
        AutoTuningCheckBox          matlab.ui.control.CheckBox
        BG_Zero_Button              matlab.ui.control.Button
        BG_ReNew_Button             matlab.ui.control.Button
```

| | |
|---|---|
| CorrEditField | matlab.ui.control.NumericEditField |
| CorrEditFieldLabel | matlab.ui.control.Label |
| PicoScopeGppLabel | matlab.ui.control.Label |
| Pico_calibration | matlab.ui.control.NumericEditField |
| TabGroup | matlab.ui.container.TabGroup |
| UsermodeTab | matlab.ui.container.Tab |
| BackgroundsubtractionCheckBox | matlab.ui.control.CheckBox |
| Label_7 | matlab.ui.control.Label |
| Label_6 | matlab.ui.control.Label |
| Average_Plots_chk | matlab.ui.control.CheckBox |
| AverLabel | matlab.ui.control.Label |
| Plot_EPR_chk | matlab.ui.control.CheckBox |
| PlotLabel_2 | matlab.ui.control.Label |
| RSData_Show_CheckBox | matlab.ui.control.CheckBox |
| ProbeDropDown | matlab.ui.control.DropDown |
| Panel_9 | matlab.ui.container.Panel |
| Current_Field | matlab.ui.control.NumericEditField |
| RunStopButton_2 | matlab.ui.control.StateButton |
| SpectraLabel | matlab.ui.control.Label |
| Panel_8 | matlab.ui.container.Panel |
| RSFreqLabel | matlab.ui.control.Label |
| Scan_Frequency | matlab.ui.control.DropDown |
| AutoLabel | matlab.ui.control.Label |
| AutpRS_CheckBox | matlab.ui.control.CheckBox |
| PhRSLabel | matlab.ui.control.Label |
| Phase_RS | matlab.ui.control.NumericEditField |
| Phase_RF | matlab.ui.control.NumericEditField |
| PhRFLabel | matlab.ui.control.Label |
| Number_of_spectra | matlab.ui.control.NumericEditField |
| RunStopButton | matlab.ui.control.StateButton |

| | |
|---|---|
| Naver_V | matlab.ui.control.NumericEditField |
| NaverLabel | matlab.ui.control.Label |
| Panel_4 | matlab.ui.container.Panel |
| Start_Imaging_3D | matlab.ui.control.StateButton |
| Image | matlab.ui.control.Image |
| CheckBox_Kinetics | matlab.ui.control.CheckBox |
| NewImagingButton | matlab.ui.control.Button |
| Start_Imaging_4D | matlab.ui.control.StateButton |
| Number_of_Imagers | matlab.ui.control.NumericEditField |
| ImsLabel | matlab.ui.control.Label |
| GmaxGcmButtonGroup | matlab.ui.container.ButtonGroup |
| gmax_4 | matlab.ui.control.RadioButton |
| gmax_5 | matlab.ui.control.RadioButton |
| gmax_3 | matlab.ui.control.RadioButton |
| Protocol_imaging | matlab.ui.control.DropDown |
| ImagProtocolLabel | matlab.ui.control.Label |
| Panel_5 | matlab.ui.container.Panel |
| Sent_G0 | matlab.ui.control.Button |
| Send2NI | matlab.ui.control.Button |
| ShimmingButton | matlab.ui.control.Button |
| GzEditField | matlab.ui.control.NumericEditField |
| GzEditFieldLabel | matlab.ui.control.Label |
| GyEditField | matlab.ui.control.NumericEditField |
| GyEditFieldLabel | matlab.ui.control.Label |
| GxEditField | matlab.ui.control.NumericEditField |
| GxEditFieldLabel | matlab.ui.control.Label |
| Panel_3 | matlab.ui.container.Panel |
| Comments_Image_Info | matlab.ui.control.TextArea |
| ScreenButton | matlab.ui.control.Button |
| SaveDataChk | matlab.ui.control.CheckBox |

185

| | |
|---|---|
| FileName | matlab.ui.control.EditField |
| F_nameLabel | matlab.ui.control.Label |
| CommentsTextArea | matlab.ui.control.TextArea |
| FolderName | matlab.ui.control.EditField |
| FolderEditFieldLabel | matlab.ui.control.Label |
| Panel | matlab.ui.container.Panel |
| ScanGppLabel | matlab.ui.control.Label |
| RS_Amp_DropDown | matlab.ui.control.DropDown |
| Ax_1 | matlab.ui.control.UIAxes |
| Ax_2 | matlab.ui.control.UIAxes |
| Ax_3 | matlab.ui.control.UIAxes |
| TuningTab | matlab.ui.container.Tab |
| ReflThershEditField | matlab.ui.control.NumericEditField |
| ReflThershEditFieldLabel | matlab.ui.control.Label |
| FineCWControlLabel | matlab.ui.control.Label |
| CW_Freq | matlab.ui.control.NumericEditField |
| RFMHzEditFieldLabel | matlab.ui.control.Label |
| DTC2 | matlab.ui.control.NumericEditField |
| DTC2EditFieldLabel | matlab.ui.control.Label |
| DTC1 | matlab.ui.control.NumericEditField |
| DTC1EditFieldLabel | matlab.ui.control.Label |
| PlottunepatternCheckBox | matlab.ui.control.CheckBox |
| SmartTuneButton | matlab.ui.control.Button |
| SetonceButton | matlab.ui.control.Button |
| kHzButton_2 | matlab.ui.control.Button |
| kHzButton | matlab.ui.control.Button |
| TunerangeButtonGroup | matlab.ui.container.ButtonGroup |
| Button_780 | matlab.ui.control.RadioButton |
| Button_770 | matlab.ui.control.RadioButton |
| GapsButton | matlab.ui.control.RadioButton |

| | |
|---|---|
| Button_3 | matlab.ui.control.RadioButton |
| Button_2 | matlab.ui.control.RadioButton |
| Button | matlab.ui.control.RadioButton |
| testserialcombox | matlab.ui.control.EditField |
| TestserialcommunicationButton | matlab.ui.control.Button |
| DisconnectMCButton | matlab.ui.control.Button |
| ConnectMCButton | matlab.ui.control.Button |
| MCbox | matlab.ui.control.UIAxes |
| CalibrationsTab | matlab.ui.container.Tab |
| AWGGppLabel | matlab.ui.control.Label |
| AWG_calibration | matlab.ui.control.NumericEditField |
| PicoScopeTextArea | matlab.ui.control.TextArea |
| PicoScopeTextAreaLabel | matlab.ui.control.Label |
| ExpertmodeTab | matlab.ui.container.Tab |
| MaxpointformagnetfieldcorrectioniffittingissporadicCheckBox matlab.ui.control.CheckBox | |
| LWmin | matlab.ui.control.NumericEditField |
| LWminLabel | matlab.ui.control.Label |
| ScanFrequnecyHzLabel | matlab.ui.control.Label |
| Scan_Frequency_old | matlab.ui.control.DropDown |
| ButtonGroup_2 | matlab.ui.container.ButtonGroup |
| SumButton | matlab.ui.control.RadioButton |
| DownButton | matlab.ui.control.RadioButton |
| UpButton | matlab.ui.control.RadioButton |
| SettingsTab | matlab.ui.container.Tab |
| ShimminggradientEditField | matlab.ui.control.NumericEditField |
| ShimminggradientEditFieldLabel | matlab.ui.control.Label |
| Panel_7 | matlab.ui.container.Panel |
| FOVEditField | matlab.ui.control.NumericEditField |
| FOVEditFieldLabel | matlab.ui.control.Label |

| | |
|---|---|
| FindButton | matlab.ui.control.Button |
| MaxScanEditField | matlab.ui.control.NumericEditField |
| MaxScanEditFieldLabel | matlab.ui.control.Label |
| StartScanEditField | matlab.ui.control.NumericEditField |
| StartScanEditFieldLabel | matlab.ui.control.Label |
| GmaxEditField | matlab.ui.control.NumericEditField |
| GmaxEditFieldLabel | matlab.ui.control.Label |
| ButtonGroup_3 | matlab.ui.container.ButtonGroup |
| Quad_but | matlab.ui.control.RadioButton |
| Spec_but | matlab.ui.control.RadioButton |
| FT_but | matlab.ui.control.RadioButton |
| Data_but | matlab.ui.control.RadioButton |
| BGRemovalPanel | matlab.ui.container.Panel |
| HPFCorrectCheckBox | matlab.ui.control.CheckBox |
| HarmLabel | matlab.ui.control.Label |
| BGR_Harm_2 | matlab.ui.control.DropDown |
| BGR_Harm | matlab.ui.control.DropDown |
| BGR_order | matlab.ui.control.DropDown |
| OrderLabel | matlab.ui.control.Label |
| BGR_percent | matlab.ui.control.NumericEditField |
| EPRsamplesEditField | matlab.ui.control.NumericEditField |
| EPRsamplesEditFieldLabel | matlab.ui.control.Label |
| GzcalibrationEditField | matlab.ui.control.NumericEditField |
| GzcalibrationEditFieldLabel | matlab.ui.control.Label |
| GycalibrationEditField | matlab.ui.control.NumericEditField |
| GycalibrationEditFieldLabel | matlab.ui.control.Label |
| GxcalibrationEditField | matlab.ui.control.NumericEditField |
| GxcalibrationEditFieldLabel | matlab.ui.control.Label |
| SpectraaveragingEditField | matlab.ui.control.NumericEditField |
| SpectraaveragingEditFieldLabel | matlab.ui.control.Label |

```
HalfPeriodEditField          matlab.ui.control.NumericEditField

HalfPeriodEditFieldLabel     matlab.ui.control.Label

DCMagneticFieldGPanel        matlab.ui.container.Panel

DC_Max_Volt_V                matlab.ui.control.NumericEditField

ATTVoltageabslimitVLabel     matlab.ui.control.Label

DC_Coil_Calibration_V        matlab.ui.control.NumericEditField

CoilCalibrationVGEditFieldLabel  matlab.ui.control.Label

DigitizerPanel               matlab.ui.container.Panel

TextArea_3                   matlab.ui.control.TextArea

DigitizerPanel_2             matlab.ui.container.Panel

yyyy                         matlab.ui.control.NumericEditField

TBDLabel_7                   matlab.ui.control.Label

xxxx                         matlab.ui.control.NumericEditField

TBDLabel_6                   matlab.ui.control.Label

Flags                        matlab.ui.control.NumericEditField

TBDLabel_5                   matlab.ui.control.Label

VertVoltage                  matlab.ui.control.NumericEditField

VerticalVoltageLabel         matlab.ui.control.Label

Mode                         matlab.ui.control.NumericEditField

TBDLabel_4                   matlab.ui.control.Label

NumAverages                  matlab.ui.control.NumericEditField

NumaveragesLabel             matlab.ui.control.Label

NumSegments                  matlab.ui.control.NumericEditField

NumsegmentsLabel             matlab.ui.control.Label

NumSamples                   matlab.ui.control.NumericEditField

NumsamplesLabel              matlab.ui.control.Label

Timeout                      matlab.ui.control.NumericEditField

TBDLabel_3                   matlab.ui.control.Label

NumChan                      matlab.ui.control.NumericEditField

TBDLabel_2                   matlab.ui.control.Label
```

| | |
|---|---|
| Saml_Interv | matlab.ui.control.NumericEditField |
| SamplinterLabel | matlab.ui.control.Label |
| Re_arm | matlab.ui.control.NumericEditField |
| TBDLabel | matlab.ui.control.Label |
| AWGSettingsPanel | matlab.ui.container.Panel |
| ResetArduinoButton | matlab.ui.control.Button |
| ResetAWGAutostateButton | matlab.ui.control.Button |
| PauseButton | matlab.ui.control.StateButton |
| ResetAWGtoTuneCWstateButton | matlab.ui.control.Button |
| CalibrationPanel | matlab.ui.container.Panel |
| LRG_2 | matlab.ui.control.EditField |
| LRG | matlab.ui.control.EditField |
| TextArea_2 | matlab.ui.control.TextArea |
| AWGPanel | matlab.ui.container.Panel |
| AWG_Set | matlab.ui.control.Button |
| AWG_amplitude | matlab.ui.control.NumericEditField |
| AmplLabel | matlab.ui.control.Label |
| AWG_phase | matlab.ui.control.NumericEditField |
| PhaseLabel | matlab.ui.control.Label |
| CoilTypeButtonGroup | matlab.ui.container.ButtonGroup |
| SmallButton | matlab.ui.control.RadioButton |
| MiddleButton | matlab.ui.control.RadioButton |
| LargestButton | matlab.ui.control.RadioButton |
| SamplingTX | matlab.ui.control.EditField |
| SamplingBasePostSamSegInxEditFieldLabel | matlab.ui.control.Label |
| PicoscopeSettingsPanel | matlab.ui.container.Panel |
| RESETButton | matlab.ui.control.Button |
| TextArea | matlab.ui.control.TextArea |
| RetrieveDataTX | matlab.ui.control.EditField |
| RetrieveStart_iSeg_iRatioModeLabel | matlab.ui.control.Label |

```
        TriggerPanel                      matlab.ui.container.Panel

        AutoTr                            matlab.ui.control.NumericEditField

        AutoTrLabel                       matlab.ui.control.Label

        Direct                            matlab.ui.control.NumericEditField

        DirecLabel                        matlab.ui.control.Label

        Delay                             matlab.ui.control.NumericEditField

        DelayLabel                        matlab.ui.control.Label

        Treshold                          matlab.ui.control.NumericEditField

        TresholdmVEditFieldLabel          matlab.ui.control.Label

        Channel                           matlab.ui.control.DropDown

        ScanParsPanel                     matlab.ui.container.Panel

        FunGenVoltage_box                 matlab.ui.control.NumericEditField

        FunGenVLabel                      matlab.ui.control.Label

        FunGenPhase_box                   matlab.ui.control.NumericEditField

        FGPhaseDegrLabel                  matlab.ui.control.Label

        FunGenReset                       matlab.ui.control.Button

        UpdatePlot                        matlab.ui.control.CheckBox

    end



properties (Access = public)


    True_Gpp= 0;



    % Pico
    obj_picoScope;
    settings_picoScope;
    % Fun generator
    obj_funGen;
```

```
settings_funGen;

% 'U1084'

obj_U1084;

settings_U1084;

% Magnet

obj_magnet;

settings_magnet

% NI

obj_NI

settings_NI

%MC

MC

%AWG

awg_obj


Spectra; % info & data realted to spectra

Magnetic_Field = 0; % Description

Magnetic_Field_Guess = 0; % Description

Magnetic_Field_Fit % Description

CF

best_reflection

tunecounter

autocenter=0;

rs_bg = 0;



experiment % Description


pico_Calib_All  = 0.2% Description

CC = [0.153 0.25  0.38 0.25 0.513]; % coils constants_list  % Description
```

```matlab
    Compress_pars; % Description

    Aver_multiplier  = 1; % Description

    FOV = 22; % [mm] % Description
end


methods (Access = public)


    function send2teensy = Make_Serial_Pars(app)
        mode = 0; % '0' to set only pars


        switch app.TunerangeButtonGroup.SelectedObject.Text


            case '770-790'

                range = 77;


            case '780-800'

                range = 78;


            case '790-810'

                range = 79;
            case '800-820'

                range = 80;
            case '810-830'

                range = 81;
            case 'Gaps'

                range = 80;
        end


        RF = app.CW_Freq.Value;
        if RF >= range *10
```

```matlab
            RF_ = round((RF - range * 10) * 1000);
% OLD: RF_ = round((RF - 790) * 100);
            RF_A = floor (RF_/256);
            RF_B = RF_ - RF_A *256;
            DTC_1 = app.DTC1.Value;
            DTC_2 = app.DTC2.Value;


            Refl_THrehold = app.ReflThershEditField.Value;
            send2teensy = uint8( [mode, range, RF_A, RF_B, DTC_1,
DTC_2, Refl_THrehold ] ); % 6 bytes total ;
        else
            warndlg('Change tuning range');
            send2teensy = -2;
        end


        x = 0;
    end


    function smart_tune_noread(app)
        send2teensy = Make_Serial_Pars(app);
        send2teensy(1) = 3;
        write(app.MC,send2teensy ,'uint8');
        readback=read(app.MC,1,'uint8');
        while readback ~= 0
            warning("teensy communication error")
            pause(1);
            write(app.MC,send2teensy,'uint8');
            readback=read(app.MC,1,'uint8');
        end
    end
```

```matlab
function Run_Spectra(app)
    N_spectra = app.Number_of_spectra.Value;
    RS_Freq = str2double(app.Scan_Frequency.Value);
    List_F = zeros(1,5);


    for n = 1 : 5
        List_F(n) = str2double( app.Scan_Frequency.Items{n} );
    end
    inx = List_F == RS_Freq;
    app.Pico_calibration.Value = app.CC(inx) *app.CorrEditField.Value;



    %  parameters
    p.dt = app.settings_U1084.sampl_interval;
    dt = p.dt;
    N_samples = app.settings_U1084.nbrSamples;
    t = (1: N_samples) * p.dt;
    y = exp(3*1i*2*pi *RS_Freq*t);
    % save comp_test dt y RS_Freq
    [dt,v1,dv,v_end,IQ]=Compress_RS_data(dt,y,RS_Freq);
    p.size = length(IQ);
    p.protocol =   'Spectra';
    p.number_spectra = N_spectra ;
    p.data = zeros(N_spectra, p.size);
    p.measured   = 0;
```

```
p.rs_freq = RS_Freq;

p.v = [v1,dv,v_end];

average_A = 0;

average_B = 0;


app.Compress_pars.dt = dt;

app.Compress_pars.dv = dv;

app.Compress_pars.v1 = v1;

app.Compress_pars.v_end = v_end;

app.Compress_pars.IQsize = length(IQ);


% Scan Amplitude

% app.SetScanAmplitude_CB; % CB function

%  Bpp = app.Actual_Gpp;

Gpp = str2double(app.RS_Amp_DropDown.Value);

Bpp = set_Scan_Amplitude(app, Gpp, 0 ,1);


% Digitizer pars

app.settings_U1084.nbrAvg = app.Naver_V.Value;

value{1} = 'NumAver';

value{2} = app.Naver_V.Value;

send_dev(app.obj_U1084,'U1084',value, app.settings_U1084);


% Deconvolution pars

pars=rsDefaults; % loads default par

% pars.Nh = [1, 1];

H1 = str2double( app.BGR_Harm.Value );

H2 = str2double( app.BGR_Harm_2.Value );

pars.Nh = [H1, H2] ;

pars.polynomB = [app.BGR_percent.Value, str2double(app.BGR_order.Value)];
```

```
pars.fwhm = app.LWmin.Value;

pars.fig = 'None';

if app.AutpRS_CheckBox.Value

    pars.APC = [20, 3, 5];

else

    pars.APC = [ 1, .01, 5];

end


% Background

if app.BackgroundsubtractionCheckBox.Value

    app.Collect_Background;

end

% app.rs_bg


% ------------MAJOR LOOP >>------------------------------------------------------>>

for k = 1 : N_spectra

    if app.AutoTuningCheckBox.Value

        app.tunecounter = 0;

        app.Smart_Tune(app);

    else

        %  smart_tune_noread(app);

    end



    if app.autocenter ==1

        try

            Magnet_auto_field(app);

            pause(0.1);

        catch

            warndlg (' Magnet_auto_field(app); ');
```

```
        end
    end


    rs_data = get_dev(app.obj_U1084,'U1084','data', app.settings_U1084);

    [~,~,~,~,IQ]=Compress_RS_data(p.dt,rs_data,RS_Freq);

    p.data(k,:) = IQ;

    p.cw(k,1) = app.CW_Freq.Value; %continuous wave frequency

    p.dtc1(k,1) = app.DTC1.Value; %dtc1 value

    p.dtc2(k,1) = app.DTC2.Value; %dtc2 value

    p.reflection(k,1) = app.best_reflection;

    p.time(k,1) = datetime("now",Format="dd-MMM-uuuu HH:mm:ss");

    p.magnet(k,1) = app.Magnetic_Field;



    [t, rs_i] = decompress(app,p.v,IQ);

    try

        rs_data = rs_i - app.rs_bg;

    catch

        rs_data = rs_i;

    end



    if 0 % test

        %%

        pars.APC = [20, 3, 5];

        pars.fig = "Abs"

        rs_test = rs_data;

        t_test = t;

        pars.phases = [app.Phase_RS.Value, app.Phase_RF.Value];

        [h, SA, SB, pars]=RsEPR2Gen(t,rs_test,Bpp,RS_Freq,pars);
```

```
        plot(t,imag(rs_data),t,real(rs_data))

        plot(real(rs_data))

        plot(h,SA,h,SB);

        save raw_data_G2_2S t rs_test Bpp RS_Freq pars

        %R

end



if app.Plot_EPR_chk.Value % Plot

    pars.phases = [app.Phase_RS.Value, app.Phase_RF.Value];


    try

        [h, SA, SB, pars]=RsEPR2Gen(t,rs_data,Bpp,RS_Freq,pars);

        if app.AutpRS_CheckBox.Value

            app.Phase_RS.Value = app.Phase_RS.Value + pars.rs_ph_corr;

        end

    catch

        h = 1 : 10;

        SA = h;

        SB = h;

    end

    average_A =average_A + SA;

    average_B =average_B + SB;

    if app.Average_Plots_chk.Value

        SB = average_B;

        SA = average_A;

    end

    % fit

    SAB = SA + SB;

    options = optimoptions('lsqcurvefit','Algorithm',
```

```matlab
    'levenberg-marquardt');
                    options.StepTolerance=1e-9;

                    options.FunctionTolerance = 1e-9;

                    options.OptimalityTolerance = 1e-9;

                    SpectralModel = app.ProbeDropDown.Value;

                    A = double ( sum(SAB) * (h(2) -h(1)) );

                    if A < 0 ; A = 0; end

                    isGetFitLine  = 1; % compute Y fit line

                    Ampl_guess =    A; % no Guess

                    %  SpectralModel = app.SpectralModelDropDown.Value;

                    try

                        [pp, new_sp, ~] = Fit_1D(h, SAB, Ampl_guess, SpectralModel,
isGetFitLine, options);

                        Lw = round(pp(2) *1000);


                        app.Magnetic_Field_Fit = pp(3); % Description


                        LW_t = ['FWHM (Lor) = ' num2str(Lw, '%.2f') ';
Pos = ' num2str(pp(3), '%.3f')];

                        app.plot_Spectra(h,SA,SB,new_sp,rs_data);

                        title(app.Ax_1,LW_t);

                        pause(0.5);


                    catch

                        new_sp = 0 * h;

                        app.plot_Spectra(h,SA,SB,new_sp, rs_data);

                        disp('error of fitting')

                    end

                    %

                end
```

```matlab
            if ~app.RunStopButton.Value; break; end

        end % end for


        app.RunStopButton.Value = 0;

        app.RunStopButton.BackgroundColor = 0.5* [1, 1 , 1];

        app.RunStopButton.Text = 'Start spectra';


        if app.SaveDataChk.Value

            fn = [app.FolderName.Value '\' app.FileName.Value];

            save([fn '_sig'] ,'p');

            if app.Plot_EPR_chk.Value % Plot

                Spec.AB =  average_A + 1i * average_B;

                Spec.h = h;

                save([fn '_spe'] ,'Spec');

            end


        end


    end


    function plot_Spectra(app,h,SA,SB,S_fit, rs_data)


        if app.RSData_Show_CheckBox.Value

            rs = rs_data - mean(rs_data);

            plot(app.Ax_1, h, SA + SB);

            plot(app.Ax_3, real(rs));

            plot(app.Ax_2, imag(rs));

        else

            plot(app.Ax_3, h, SA, h, SB);

            plot(app.Ax_2, h, SA - SB);
```

```matlab
            Err_ = std(SA-SB);

            I = max(SA +SB );

            SNR = I/Err_;

            %  tx = []

            %  title(app.Ax_2, 'EPR_U_p - EPR_D_o_w_n')

            title( app.Ax_2, ['SD of noise = ' num2str(Err_) ';
SNR = ' num2str(SNR) '; I = ' num2str(I) ] );

            try

                plot(app.Ax_1, h, SA + SB,h,S_fit);

            catch

                plot(app.Ax_1, h, SA + SB);

            end


        end

        axis(app.Ax_1,'tight');

        axis(app.Ax_2,'tight');

        axis(app.Ax_3,'tight');


    end



    function Run_imaging(app, mode)



        pars=rsDefaults; % loads default par



        RS_Freq = str2double(app.Scan_Frequency.Value);

        app.experiment.RS_Freq = RS_Freq;

        H1 = str2double( app.BGR_Harm.Value );

        H2 = str2double( app.BGR_Harm_2.Value );
```

```matlab
pars.Nh = [H1, H2] ;

pars.polynomB = [app.BGR_percent.Value, str2double(app.BGR_order.Value)];

pars.fwhm = app.LWmin.Value;

pars.APC = [5 0.1 5];

pars.fig = 'None';

pars.phases = [app.Phase_RS.Value, app.Phase_RF.Value];

options = optimoptions('lsqcurvefit','Algorithm','levenberg-marquardt');

options.StepTolerance=1e-9;

options.FunctionTolerance = 1e-9;

options.OptimalityTolerance = 1e-9;


switch mode

    case '4D'

        app.Start_Imaging_3D.Value = 0; %
        MeasureBlocks_4D(app,pars,options); % 4D measurment


        app.Start_Imaging_4D.Value = 0 ;
        app.Start_Imaging_4D.BackgroundColor = 0.5* [1, 1 , 1];
        app.Start_Imaging_4D.Text = '4D';



    case '3D'

        app.Start_Imaging_4D.Value = 0; %
        MeasureBlock_3D(app);


        app.Start_Imaging_3D.Value = 0 ;
        app.Start_Imaging_3D.BackgroundColor = 0.5* [1, 1 , 1];
        app.Start_Imaging_3D.Text = '3D';
```

```
        end




end




function [Gx, Gy, Gz] = Read_Gxyz(app)


    Gx = app.GxEditField.Value;

    Gy = app.GyEditField.Value;

    Gz = app.GzEditField.Value;
end




function Actual_Gpp = set_Scan_Amplitude(app, Gpp, if_show, if_reset)
    % Gpp = Vpp*coilconstant/1000;

    coilconstant = app.AWG_calibration.Value;

    %  Gpp = str2double(app.RS_Amp_DropDown.Value) + extra;

    %   Vpp = Gpp /  coilconstant /2 * 1.5 /7 *6 *3/6.3;

    %  new_coil_const = 0.107;

    Vpp = Gpp * coilconstant;

    if Vpp > 3; Vpp = 3; end  % for now


    if if_reset

        send_dev(app.obj_funGen,'Fun_Gen_Ampl',Vpp,  ' ');

        disp('Reset Scan')

        pause(1);
```

```matlab
        else

                % disp('Skip reset Scan')

            end



            % Gpp = str2double(app.RS_Amp_DropDown.Value);
            %              app.set_Scan_Amplitude(Gpp); % set Scan Amplitude
    % based on Wanted Gpp
            signal = app.get_Scan_Amplitude; % get Voltage
            coilconstant = app.Pico_calibration.Value;


            %fit = signal.fit; % =0.5*Ampl*cos(2*pi*freq*t1+angle(z2))+dc;
% fitted theroretical line
            Vpp = signal.Ampl; %  = Ampl; %
            Actual_Gpp= Vpp/coilconstant/1000; % signal reads in [mV]
            %  Actual = app.Actual_Gpp;


            if if_show
                t = signal.t;
                plot(app.Ax_3,t,signal.chA,t,signal.fit);
                axis(app.Ax_3,'tight');
                title(app.Ax_3, ['Actual: '  num2str(Actual_Gpp)]);
            end
            % Gpp = Vpp*coilconstant/1000;


            %  app.BackgroundsubtractionCheckBox.Value = 0;
            app.rs_bg = 0;



        end
```

```
function signal = get_Scan_Amplitude(app)

    value = 'data';

    signal = get_dev(app.obj_picoScope,'picoScope', value, app.settings_picoScope);

    ps5000a = app.settings_picoScope.ps5000a;

    chA = signal.chA;

    iii= 1:length(chA) ;

    sampling_interval = (ps5000a.time_base- 2)/ 125000000; % [seconds]

    t = iii*sampling_interval;

    z=chA;


    % Fit Data

    freq = str2double(app.Scan_Frequency.Value);

    Period=1/freq;

    Ncycles=floor(t(end)/Period);

    inx=t<=Ncycles*Period;

    t1=t(inx);

    z1=z(inx);

    ref=exp(1i*2*pi*freq*t1); %reference wave

    z2= z1'*ref';

    L=length(t1);

    Ampl =4*abs(z2)/L ; %for peak-to-peak voltage

    ph =angle(z2)*180/pi;

    dc=sum(z1)/L;

    signal.fit=0.5*Ampl*cos(2*pi*freq*t+angle(z2))+dc; % fitted theroretical line

    signal.Ampl = Ampl; %

    signal.ph = ph;

    signal.t = t';

end
```

```matlab
function [t, rs_i] = decompress(app,V,data)

    IQ = data; % block.data(s,:);

    % V =  block.v;

    v = V(1) : V(2) : V(3);

    [ti,rs] = ifftM2Gen(v,IQ);

    dt = ti(2) - ti(1);

    L = length(rs);

    M = 8;

    rs_i = interpft(rs,L * M);

    t = (1 :  L*M) * dt/M;

    t = t - t(1);

    t = transpose(t);

    % Vm = V(2);

end




function MeasureBlocks_4D(app,pars,options,SpectralModel)


    [Gx_S, Gy_S, Gz_S] = Read_Gxyz(app); % shimming

    dt = app.settings_U1084.sampl_interval;

    isGetFitLine  = 1;

    first = app.experiment.current_block; %  - 1; % # block to start with

    last = length(app.experiment.plan);    % # block to end

    SpectralModel = app.ProbeDropDown.Value;

    RS_Freq = str2double(app.Scan_Frequency.Value);


    Gpp_ = 0;



    for k =  first : last
```

```matlab
            if app.experiment.SkipBlocks(k)   % if to collectgiven block

                tic

                inx = app.experiment.plan(k).block_inx;

                current_block = app.experiment.block(inx);

                %Tune if available


%% G =  0

                app.Send_shimming_2NI; % sets G = 00

                %% Auto-tune

                if 1

                    app.tunecounter = 0;

                    app.Smart_Tune(app);

                else

                    app.smart_tune_noread(app);

                end


    %% Set #averages

                NB = app.experiment.Num_blocks;

                block_mod = mod(k, NB); % 1 : 26 now

                switch   block_mod

                    case 1

                        app.Aver_multiplier = 1;

                    case (NB - 3)

                        app.Aver_multiplier = 2;

                    case 0 % 26 26 *2 ...

                        app.Aver_multiplier = 4;

                end

                value_A{1} = 'NumAver';

                value_A{2} = app.Naver_V.Value * app.Aver_multiplier;

                send_dev(app.obj_U1084,'U1084',value_A, app.settings_U1084);
```

```
%% Scan widths

            Gpp = str2double(app.RS_Amp_DropDown.Value) + current_block.SWplus;
            if Gpp ~= Gpp_
                True_Gpp_ = set_Scan_Amplitude(app,Gpp,0 ,1); % set Sacn Amplitude
                Gpp_ = Gpp ;
            else
                True_Gpp_ = set_Scan_Amplitude(app,Gpp,0, 0); % set Sacn Amplitude
            end


            current_block.SW_set = Gpp;
            current_block.SW_get =  True_Gpp_;


%% Read G = 0 data
            rs_data = get_dev(app.obj_U1084,'U1084','data', app.settings_U1084);
            rs_data  = rs_data /app.Aver_multiplier;
            [dt_,v1,dv,v_end,IQ]=Compress_RS_data(dt,rs_data,RS_Freq);
            current_block.Spectrum0  = IQ;
            current_block.v =   [v1,dv,v_end];
            current_block.raw_data_size =  length(rs_data);
            current_block.cw = app.CW_Freq.Value; %added
            current_block.dtc = [app.DTC1.Value, app.DTC2.Value]; %added
            current_block.reflection = app.best_reflection;
            current_block.Magnetic_Field = app.Magnetic_Field;


            %% Field correction
            if block_mod == 1
                t = (1: length(rs_data)) * dt;
                [h, SA, SB, pars]=RsEPR2Gen(t,rs_data, True_Gpp_,RS_Freq,pars);
```

209

```
SAB = SA + SB;

Ampl_guess = double ( sum(SAB) * (h(2) -h(1)) );

if Ampl_guess<0; Ampl_guess=0; end

[pp, new_sp, ~] = Fit_1D(h, SAB, Ampl_guess, SpectralModel,
isGetFitLine, options);

app.Magnetic_Field_Fit = pp(3);


Lw = round(pp(2) *1000);

LW_t = ['FWHM (Lor) = ' num2str(Lw, '%.2f') ';
Pos = ' num2str(pp(3), '%.3f')];

app.plot_Spectra(h,SA,SB,new_sp,rs_data);

title(app.Ax_1,LW_t);


Magnet_auto_field(app);


% Read G = 0 data again

rs_data = get_dev(app.obj_U1084,'U1084','data', app.settings_U1084);

rs_data  = rs_data /app.Aver_multiplier;

[dt_,v1,dv,v_end,IQ]=Compress_RS_data(dt,rs_data,RS_Freq);

current_block.Spectrum0  = IQ;

current_block.v =  [v1,dv,v_end];

current_block.raw_data_size =  length(rs_data);

current_block.cw = app.CW_Freq.Value; %added

current_block.dtc = [app.DTC1.Value, app.DTC2.Value]; %added

current_block.reflection = app.best_reflection;

current_block.Magnetic_Field = app.Magnetic_Field;




        end


                    210
```

```matlab
%%


G3 = current_block.G3;


% >>> LOOP over Block
for s = 1 : size(G3,1)
    g3 = G3(s,:); % gradinets x y z
    value_g = [0, Gx_S + g3(1), Gy_S + g3(2), Gz_S + g3(3)];
    send_dev(app.obj_NI,'gradients',value_g, app.settings_NI);
    if ~app.Start_Imaging_4D.Value; break; end
    pause(0.001); % replace  later by smthg userfull /datasaving ...
serail ...

        rs_data = get_dev(app.obj_U1084,'U1084','data', app.settings_U1084);
        rs_data  = rs_data /app.Aver_multiplier;
        [dt_,v1,dv,v_end,IQ]=Compress_RS_data(dt,rs_data,RS_Freq);
        current_block.data(s, :) = IQ;


end
% <<< LOOP over Block


% off resonance background collection
send_dev(app.obj_magnet,'magnet', 0 , app.settings_magnet); % Field OFF
pause(0.5);
rs_data = get_dev(app.obj_U1084,'U1084','data', app.settings_U1084);
[~,~,~,~,IQ] = Compress_RS_data(dt,rs_data,RS_Freq);
current_block.background = IQ;
send_dev(app.obj_magnet,'magnet',app.Magnetic_Field,
app.settings_magnet);
```

```matlab
                % end background collection

                app.Send_shimming_2NI; % sets G = 000
                current_block.toc = toc;
                current_block.date = datetime("now",Format="dd-MMM-uuuu HH:mm:ss");

                if ~app.Start_Imaging_4D.Value; break; end
                % save
                fn_full = [app.FolderName.Value '\' app.FileName.Value  '~block~'
num2str(k)];

                save(fn_full,'current_block');
                disp(fn_full);

                app.experiment.current_block = k;
                app.Comments_Image_Info.Value = ['Block # ' num2str(k) ' out of '
num2str(last) ' is done; time per block = ' num2str(round(toc/60,1))
' mins; Wait another ' num2str(round((last - k) * toc/60,1)) ' mins'];
                % pause(1);

            end % if  collect block

        end
    end

    function results = MeasureBlock_3D(app)
        [Gx_S, Gy_S, Gz_S] = Read_Gxyz(app); % shimming
        dt = app.settings_U1084.sampl_interval;
        app.Smart_Tune(app); % tuning
        RS_Freq = str2double(app.Scan_Frequency.Value);
```

```matlab
%% Vary averages

app.Aver_multiplier = 4;

value_A{1} = 'NumAver';

value_A{2} = app.Naver_V.Value * app.Aver_multiplier;

send_dev(app.obj_U1084,'U1084',value_A, app.settings_U1084);


%%


% set_Scan_Amplitude(app, Gpp)


Gpp = str2double(app.RS_Amp_DropDown.Value);

Gmax = str2double(app.GmaxGcmButtonGroup.SelectedObject.Text);

True_Gpp = set_Scan_Amplitude(app,Gpp + Gmax*app.FOV/10,0 ,1);
% set Scan Amplitude


current_block.SW_set = Gpp;

current_block.SW_get =  True_Gpp;


%correct magnetic field from last GO iteration
%                       if app.autocenter ==1 && k ~= first
%                           Magnet_auto_field(app);
%                       end


% GO ;  get & compress data

app.Send_shimming_2NI; % sets G = 00

rs_data = get_dev(app.obj_U1084,'U1084','data', app.settings_U1084);

rs_data  = rs_data /app.Aver_multiplier;
```

```matlab
[dt_,v1,dv,v_end,IQ]=Compress_RS_data(dt,rs_data,RS_Freq);

current_block.Spectrum0  = IQ;

current_block.v =  [v1,dv,v_end];

current_block.raw_data_size =  length(rs_data);

current_block.cw = app.CW_Freq.Value; %added

current_block.dtc = [app.DTC1.Value, app.DTC2.Value]; %added

current_block.reflection = app.best_reflection;



% >>> LOOP over Block
G3 = app.experiment.block_3D.G3;

ss = size(G3);

L = size(G3,2);

M = size(G3,3) ;

current_block.data = zeros (L,M,length(IQ));


for s = 1 : L
    for m = 1 : M


        g3 = G3(:,s,m); % gradinets x y z
        value_g = [0, Gx_S + g3(1), Gy_S + g3(2), Gz_S + g3(3)];
        send_dev(app.obj_NI,'gradients',value_g, app.settings_NI);
        % if ~app.Start_Imaging_4D.Value; break; end
        pause(0.001); % replace  later by smthg userfull /datasaving ...
serail ...


        rs_data = get_dev(app.obj_U1084,'U1084','data', app.settings_U1084);
        [dt_,v1,dv,v_end,IQ]=Compress_RS_data(dt,rs_data,RS_Freq);


        current_block.data(s,m,:) = IQ;
```

214

```matlab
    end
end
% <<< LOOP over Block


% off resonance background collection
send_dev(app.obj_magnet,'magnet', 0 , app.settings_magnet); % Field OFF
pause(0.5);
rs_data = get_dev(app.obj_U1084,'U1084','data', app.settings_U1084);
[~,~,~,~,IQ] = Compress_RS_data(dt,rs_data,RS_Freq);
current_block.background = IQ;
send_dev(app.obj_magnet,'magnet',app.Magnetic_Field,app.settings_magnet);
% end background collection


app.Send_shimming_2NI; % sets G = 000
current_block.toc = toc;
current_block.date = datetime("now",Format="dd-MMM-uuuu HH:mm:ss");


% if ~app.Start_Imaging_4D.Value; break; end
% save
fn_full = [app.FolderName.Value '\' app.FileName.Value  '~block_3D~'];


save(fn_full,'current_block');
disp(fn_full);


%  app.experiment.current_block = k;
app.Comments_Image_Info.Value = 'Block_3D is done';
% pause(1);


% end % if  collect block
```

```matlab
            %end
        end
    end


    % Callbacks that handle component events
    methods (Access = private)


        % Code that executes after component creation
        function startupFcn(app)


            %% path
            Lib_path = 'G:\My Drive\Share with MT\EPRI GUI 2023';
            % Lib_path = 'G:\My Drive\Matlab EPR Imaging\EPRI GUI 2023 March 2023';
            cd (Lib_path) %'G:\My Drive\Share with MT\EPRI GUI 2023';
            addpath(genpath(Lib_path));
            warndlg ('Teensy: Smart_Tune_test_Matrch_2023');
            % Parameters




            % Initialize devices >>>>>>>>>>>>>>>>>>>>>>>>> DEVICES
            if 1
                [app.obj_picoScope, app.settings_picoScope] = ini_devices_S('picoScope');
                [app.obj_funGen, app.settings_funGen] = ini_devices_S('Fun_Gen');
                [app.obj_U1084, app.settings_U1084] = ini_devices_S('U1084');
                [app.obj_magnet, app.settings_magnet] = ini_devices_S('magnet');
                [app.obj_NI, app.settings_NI] = ini_devices_S('NI');
                [app.MC,~] = ini_devices_S('MC');
                [app.awg_obj,~] = ini_devices_S('awg');
```

```matlab
        app.tunecounter = 0;

        app.Smart_Tune(app);


        app.TunerangeButtonGroup.SelectedObject = app.Button_3;

        app.TunerangeButtonGroupSelectionChanged(app);


        app.pico_Calib_All = app.Pico_calibration.Value;

        app.XHnage_RS_frequnecy;

    end



    app.pico_Calib_All = app.Pico_calibration.Value;

    app.Compress_pars.IQsize = 10; %


end


% Close request function: UIFigure
function UIFigureCloseRequest(app, event)
    %             obj_picoScope;
    %             settings_picoScope;
    %             % Fun generator
    %             obj_funGen;
    %             settings_funGen;
    try
        close_devices_S(app.obj_picoScope, 'picoScope');
    catch
    end


    try
```

```matlab
        close_devices_S(app.obj_funGen, 'Fun_Gen');
    catch
    end


    try
        status = AqD1_stopAcquisition(app.obj_U1084);
        if (status ~= 0)
            fprintf('Error in AqD1_stopAcquisition: %d\n', status);
        end
        % Free remaining resources
        status = Aq_closeAll();
        if (status ~= 0)
            fprintf('Error in Aq_closeAll: %d\n', status);
        end
    catch
    end


    app.Stop_Teensy;


    delete(app)
end


% Value changed function: HalfPeriodEditField
function HalfPeriodEditFieldValueChanged(app, event)
    app.tuning_adc_parameters.scan.onescan = app.HalfPeriodEditField.Value;
    app.pro = Appodization_Profile(app.tuning_adc_parameters.scan.onescan);
end


% Value changed function: RS_Amp_DropDown
function SetScanAmplitude_CB(app, event)
```

```matlab
            Gpp = str2double(app.RS_Amp_DropDown.Value);

            if_show = 1;

            Actual = set_Scan_Amplitude(app, Gpp, if_show,1);

            %

            %

            %                 app.set_Scan_Amplitude(Gpp); % set Scan Amplitude based on
    % Wanted Gpp

            %                 signal = app.get_Scan_Amplitude; % get Voltage

            %                 coilconstant = app.Pico_calibration.Value;

            %

            %                 %fit = signal.fit; % =0.5*Ampl*cos(2*pi*freq*t1+angle(z2))+dc;
% fitted theroretical line

            %                 Vpp = signal.Ampl; %   = Ampl; %

            %                 app. Actual_Gpp= Vpp/coilconstant/1000; % signal reads in [mV]

            %                 t = signal.t;

            %                 plot(app.Ax_3,t,signal.chA,t,signal.fit);

            %                 axis(app.Ax_3,'tight');

            %                 title(app.Ax_3, ['Actual: '  num2str(app.Actual_Gpp)]);

            %

            %                 % Gpp = Vpp*coilconstant/1000;

            %

            %                 %  app.BackgroundsubtractionCheckBox.Value = 0;

            app.rs_bg = 0;

        end


        % Callback function
        function Magnet_auto_field(app, event)
            app.Magnetic_Field = app.Magnetic_Field - app.Magnetic_Field_Fit;

            if app.Magnetic_Field > 32 || app.Magnetic_Field < 0.1

                app.Magnetic_Field = 1;
```

```matlab
            warndlg('Magnetic Field error','Magnet_auto_field');
            app.autocenter = 0;
        end
        send_dev(app.obj_magnet,'magnet', app.Magnetic_Field , app.settings_magnet);
        pause(1);
    end


% Button pushed function: ScreenButton
function take_screenshot(app, event)
    fn = [app.FolderName.Value '\' app.FileName.Value 'screenshot'];
    file_exist =  exist([fn '.mat'],'file');
    if  file_exist > 0
        fn = [fn num2str(app.savecounter) '.png'];
    else


        %  app.savecounter = 0;
        fn = [fn '.png']
    end
    exportapp(app.UIFigure,fn);
end


% Value changed function: RunStopButton
function Run_Stop_Spectra(app, event)
    if app.RunStopButton.Value % start
        app.RunStopButton.BackgroundColor = [0, 1 , 0];
        app.RunStopButton.Text = 'Running...';
        app.rs_bg = 0;
        Run_Spectra(app);
    else % stop
        app.RunStopButton.BackgroundColor = 0.5* [1, 1 , 1];
```

```matlab
            app.RunStopButton.Text = 'Start spectra';

        end


    end


% Value changed function: RunStopButton_2
function CF_Guess(app, event)

    %value = app.RunStopButton_2.Value;

    app.Magnetic_Field_Guess = (app.CW_Freq.Value - 750)/2.8 -  2.5   ;

    app.Magnetic_Field = app.Magnetic_Field_Guess; %+ app.Current_Field.Value;

    if app.Magnetic_Field < 0; app.Magnetic_Field = 0; end

    if app.Magnetic_Field > 30; app.Magnetic_Field = 30; end

    send_dev(app.obj_magnet,'magnet',app.Magnetic_Field , app.settings_magnet);


    app.Current_Field.Value = round( app.Magnetic_Field ,2);



end


% Value changed function: Start_Imaging_4D
function Run_Stop_Imaging(app, event)


    if app.Start_Imaging_4D.Value % start

        app.Start_Imaging_4D.BackgroundColor = [0, 1 , 0];

        app.Start_Imaging_4D.Text = 'Running...';

        if ~app.SaveDataChk.Value

            app.SaveDataChk.Value = 1;

            warndlg('Changing status to "Save Image" ... 3 secs to cancel');

            pause(3);

        end
```

```matlab
        Run_imaging(app,'4D');


        %   New_Imaging(app);
    else % stop
        app.Start_Imaging_4D.BackgroundColor = 0.5* [1, 1 , 1];
        app.Start_Imaging_4D.Text = 'Start/continue';

    end




end



% Button pushed function: NewImagingButton
function New_imaging(app, event)


    tx = ['Verify EPR Probe type; It is  ' app.ProbeDropDown.Value];
    %  load gong.mat; sound(y);
    %  pause(0.1);
    % f = warndlg(tx);
    try
        Run_Spectra(app); % to reste all pars
    catch


    end



    Gmax = str2double(app.GmaxGcmButtonGroup.SelectedObject.Text);


    block = Load_Protocol_J23('Imaging', app.Protocol_imaging.Value, Gmax,
```

```matlab
        str2double(app.Scan_Frequency.Value), app.FOV, app.Compress_pars.IQsize);
% loads protocol


            N_images = app.Number_of_Imagers.Value;
            Num_blocks = length(block);


            n = 1;              SW = zeros(1,  Num_blocks);
            for k = 1 : N_images
                for m = 1 : Num_blocks
                    plan(n).block_inx = m;
                    plan(n).image_num = k;
                    SW(m) = block(m).SWplus;
                    n = n + 1;
                end
            end


            % fatst kinetics >> skip high-G data
            %  NB = app.experiment.Num_blocks        ;



            Remove_blocks = 17; % numer of blocks to remove starting from this number
            first_image_cut = 1;
            get_blocks_inx = ones( Num_blocks,N_images);
            if app.CheckBox_Kinetics.Value && (N_images > 2)
                get_blocks_inx ( Remove_blocks : end, first_image_cut :
(N_images  - 1)  ) = 0;
                imagesc(get_blocks_inx);

            end
            app.experiment.SkipBlocks = get_blocks_inx ;
```

```matlab
%% 3D spatial imaging
block_3D = block(end);
N_Prjs = 18;


theta = (0 : 10 : 170)/180* pi;
phi= theta;
[P,T] = meshgrid(phi,theta);
G3x = sin(T).*cos(P);
G3y = sin(T).*sin(P);
G3z = cos(T);


d =  block.data;
s = size(d);
xxx = zeros(3,N_Prjs,N_Prjs);
xxx(1,:,:) = G3x;
xxx(2,:,:) = G3y;
xxx(3,:,:) = G3z;
block_3D.G3 = xxx;
block_3D.data = zeros(N_Prjs,N_Prjs, s(2));
%%


app.experiment.block_3D = block_3D;


app.experiment.Remove_blocks = Remove_blocks;
app.experiment.SWplus = SW;
app.experiment.block = block;
app.experiment.plan = plan;
app.experiment.Gmax = Gmax;
app.experiment.N_images =  N_images;
```

```matlab
app.experiment.Num_blocks =  Num_blocks;

data = block.data;

app.experiment.data_size = size(data );

app.experiment.fname = app.FileName.Value;

app.experiment.current_block = 1;

app.experiment.protocol = app.Protocol_imaging.Value;

app.experiment.Probe = app.ProbeDropDown.Value;

app.experiment.Naver = app.Naver_V.Value;

app.experiment.isKinetics = app.CheckBox_Kinetics.Value;

disp('new');

Exper  = app.experiment;

Exper.block = rmfield(Exper.block,'data');

Exper.RS_freq = str2double( app.Scan_Frequency.Value )  ;

app.experiment.Comments = app.CommentsTextArea.Value;

fn_full = [app.FolderName.Value '\' app.FileName.Value '~Exp  '];

save(fn_full,'Exper');

disp('Saved');

disp(fn_full);


app.Start_Imaging_3D.Value = 0 ;

app.Start_Imaging_3D.BackgroundColor = 0.5* [1, 1 , 1];

app.Start_Imaging_3D.Text = '3D';


app.Start_Imaging_4D.Value = 0 ;

app.Start_Imaging_4D.BackgroundColor = 0.5* [1, 1 , 1];

app.Start_Imaging_4D.Text = '4D';


app.NewImagingButton.Text =  'Done :)';

pause(1);
```

```matlab
    app.NewImagingButton.Text =  'New imaging';


end


% Callback function: GxEditField, GyEditField, GzEditField, Send2NI
function Send_shimming_2NI(app, event)
    [Gx, Gy, Gz] = Read_Gxyz(app);
    value = [0, Gx, Gy, Gz];
    send_dev(app.obj_NI,'gradients',value, app.settings_NI);
    pause(0.5);
end


% Button pushed function: ConnectMCButton
function ConnectMCButtonPushed(app, event)
    [app.MC,~] = ini_devices_S('MC');
end


% Button pushed function: DisconnectMCButton
function Stop_Teensy(app, event)
    try
        app.MC.delete;
        disp('MC disconnected');
    catch
        warning('MC cannot be stopped');
    end
end


% Selection changed function: TunerangeButtonGroup
function TunerangeButtonGroupSelectionChanged(app, event)
    selectedButton = app.TunerangeButtonGroup.SelectedObject;
```

```matlab
switch selectedButton.Text

    case '770-790'

        fprintf(app.awg_obj, 'MMEM:LOAD:STAT "Int:\Settings\770.STA"');

        write(app.MC,[3 3],'uint8');

        app.CW_Freq.Value = 780;

        readback=read(app.MC,1,'uint8');

        while readback ~= 0

            warning("teensy communication error")

            pause(1);

            write(app.MC,[3 3],'uint8');

            readback=read(app.MC,1,'uint8');

        end


    case '780-800'

        fprintf(app.awg_obj, 'MMEM:LOAD:STAT "Int:\Settings\770.STA"');

        write(app.MC,[3 3],'uint8');

        app.CW_Freq.Value = 790;

        readback=read(app.MC,1,'uint8');

        while readback ~= 0

            warning("teensy communication error")

            pause(1);

            write(app.MC,[3 3],'uint8');

            readback=read(app.MC,1,'uint8');

        end


    case '790-810'

        fprintf(app.awg_obj, 'MMEM:LOAD:STAT "Int:\Settings\790.STA"');

        write(app.MC,[3 2],'uint8');

        app.CW_Freq.Value = 800;

        readback=read(app.MC,1,'uint8');
```

```matlab
    while readback ~= 0

        warning("teensy communication error")

        pause(1);

        write(app.MC,[3 2],'uint8');

        readback=read(app.MC,1,'uint8');

    end


case '800-820'

    fprintf(app.awg_obj, 'MMEM:LOAD:STAT "Int:\Settings\800.STA"');

    write(app.MC,[3 1],'uint8');

    app.CW_Freq.Value = 810;

    readback=read(app.MC,1,'uint8');

    while readback ~= 0

        warning("teensy communication error")

        pause(1);

        write(app.MC,[3 1],'uint8');

        readback=read(app.MC,1,'uint8');

    end


case '810-830'

    fprintf(app.awg_obj, 'MMEM:LOAD:STAT "Int:\Settings\810.STA"');

    write(app.MC,[3 0],'uint8');

    app.CW_Freq.Value = 820;

    readback=read(app.MC,1,'uint8');

    while readback ~= 0

        warning("teensy communication error")

        pause(1);

        write(app.MC,[3 0],'uint8');

        readback=read(app.MC,1,'uint8');

    end
```

```matlab
        case 'Gaps'

            fprintf(app.awg_obj, 'MMEM:LOAD:STAT "Int:\Settings\GAPS.STA"');

    end

end


% Button pushed function: TestserialcommunicationButton
function Send2Teensy(app, event)
    send2teensy = Make_Serial_Pars(app);


    write(app.MC, send2teensy ,'uint8');
    readback=read(app.MC,1,'uint8');
    while readback ~= 0

        warning("teensy communication error")

        pause(1);

        write(app.MC, send2teensy,'uint8');

        readback=read(app.MC,1,'uint8');

    end


    pause(.1);

    app.testserialcombox.Value =  'x x x x x x';

    pause(0.1);

    try

        num = read(app.MC,6,"uint8");

        app.testserialcombox.Value = num2str( num );

        disp('reading from Teensy');

    catch

    end

end
```

```matlab
% Button pushed function: SetonceButton
function Set_Get_Reflection(app, event)
    send2teensy = Make_Serial_Pars(app);
    send2teensy(1) = 1;
    write(app.MC, send2teensy ,'uint8');
    readback=read(app.MC,1,'uint8');
    while readback ~= 0
        warning("teensy communication error")
        pause(1);
        write(app.MC, send2teensy,'uint8');
        readback=read(app.MC,1,'uint8');
    end
    offset = 11921;
    slope = 14.9;


    pause(.3);
    N = 301;
    try
        data = read(app.MC,N,"uint8");
        reflection = data(1 : 298);
        pause(0.1);
        [V , inx] = max(reflection);
        app.CF = (inx + offset)/slope+10*(double(send2teensy(2))-80);


        tx = ['CWF = ' num2str(app.CF) '; Reflection =  ' num2str(V)];
        frequencies = zeros(298,1);
        for i = 1:298
            frequencies(i) = (i + offset)/slope+10*(double(send2teensy(2))-80);
        end
        plot(app.MCbox,frequencies,reflection);
```

```matlab
        title(app.MCbox, tx);

        disp('reading from Teensy');

    catch


    end


    % write(app.MC, send2teensy ,'uint8');


end


% Button pushed function: SmartTuneButton
function Smart_Tune(app, event)

    send2teensy = Make_Serial_Pars(app);
    if  send2teensy(1) >= 0

        xxxx = 0;
        send2teensy(1) = 2;
        write(app.MC, send2teensy ,'uint8');
        offset = 11921;
        slope = 14.9;


        pause(.3);
        readback=read(app.MC,1,'uint8');
        while readback ~= 0

            warning("teensy communication error")
            pause(1);
            write(app.MC, send2teensy,'uint8');
            readback=read(app.MC,1,'uint8');

        end
        N = 307;
        %try
```

```
data = read(app.MC,N,"uint8");
while length(data)<307 || data(305)>32
    flush(app.MC);
    send2teensy = Make_Serial_Pars(app);
    send2teensy(1) = 2;
    write(app.MC, send2teensy ,'uint8');


    pause(.3);
    readback=read(app.MC,1,'uint8');
    while readback ~= 0
        warning("teensy communication error")
        pause(1);
        write(app.MC, send2teensy,'uint8');
        readback=read(app.MC,1,'uint8');
    end
    N = 307;
    %try
    data = read(app.MC,N,"uint8");
end
flush(app.MC);
reflection = data(1:296);
%   reflection = data(1:300);
info_back = data(302: 307);
app.best_reflection = info_back(1);
best_position = info_back(2);
app.DTC1.Value = info_back(3);
app.DTC2.Value = info_back(4);
best_segment =  info_back(5);
test = info_back(6); %  == 255
```

```matlab
%                       best_reflection = info_back(2);
%                       dtc1 = info_back(4);
%                       dtc2 = info_back(5);
%                       save('reflectiondata.mat','reflection','info_back',
'best_position','best_reflection','dtc1','dtc2');
%  best_segment = info_back(6);


%send2teensy = Make_Serial_Pars(app);
range = send2teensy(2);
app.CW_Freq.Value = round((best_position + offset)/slope + 10*
(double(range)-80),2);
% disp(info_back);
app.testserialcombox.Value = num2str(info_back);
pause(0.1);
[V , inx] = max(reflection);
app.CF = (inx + offset)/slope+10*(double(range)-80)+0.05;
if app.PlottunepatternCheckBox.Value ==1
    tx = ['CWF = ' num2str(app.CF) '; Reflection =  ' num2str(V)];
    frequencies = zeros(296,1);
    for i = 1:296
        frequencies(i) = (i + 11921)/14.9+10*(double(range)-80);
    end
    plot(app.MCbox,frequencies',reflection);
    xlim(app.MCbox,[frequencies(1), frequencies(end-5)]);
    title(app.MCbox, tx);
    disp('reading from Teensy');
    %save 'CW_Frequency' 'app.CF' ; %% temp solution
end
if app.tunecounter > 3
```

```matlab
            app.tunecounter = 0;

            return;
        end



    end
%                       switch app.TunerangeButtonGroup.SelectedObject.Text
%                           case '790-810'
%                               if app.CW_Freq.Value>808 || app.best_reflection < 120
%                                   app.tunecounter = app.tunecounter + 1;
%                                   app.TunerangeButtonGroup.SelectedObject =
app.Button_2; %800-820
%                                   app.TunerangeButtonGroupSelectionChanged(app);
%                                   pause(7);
%                                   app.Smart_Tune(app);
%                               end
%                           case '800-820'
%                               if app.CW_Freq.Value < 802
%                                   app.tunecounter = app.tunecounter + 1;
%                                   app.TunerangeButtonGroup.SelectedObject =
app.Button; %790-810
%                                   app.TunerangeButtonGroupSelectionChanged(app);
%                                   pause(7);
%                                   app.Smart_Tune(app);
%                               elseif app.CW_Freq.Value > 816
%                                   app.tunecounter = app.tunecounter + 1;
%                                   app.TunerangeButtonGroup.SelectedObject =
app.Button_3; %810-830
%                                   app.TunerangeButtonGroupSelectionChanged(app);
%                                   pause(7);
```

```matlab
%                                      app.Smart_Tune(app);

%                               end

%                          case '810-830'

%                              if app.CW_Freq.Value < 812 ||
app.best_reflection <120

%                                  app.tunecounter = app.tunecounter + 1;

%                                  app.TunerangeButtonGroup.SelectedObject =
app.Button_2; %800-820

%                                  app.TunerangeButtonGroupSelectionChanged(app);

%                                  pause(7);

%                                  app.Smart_Tune(app);

%                              end

%                      end

%catch


%end

end


% Button pushed function: kHzButton
function minus50kHz(app, event)
    app.CW_Freq.Value = app.CW_Freq.Value - 0.05;

    app.Set_Get_Reflection;
end


% Button pushed function: kHzButton_2
function Plus50kHz(app, event)
    app.CW_Freq.Value = app.CW_Freq.Value + 0.05;

    app.Set_Get_Reflection;
end
```

```matlab
% Value changed function: Current_Field
function Current_FieldValueChanged(app, event)
    app.Magnetic_Field= app.Current_Field.Value;
    send_dev(app.obj_magnet,'magnet',app.Magnetic_Field , app.settings_magnet);
end


% Button pushed function: AutocenterOffButton
function AutocenterOffButtonPushed(app, event)
    if app.autocenter == 0
        app.autocenter = 1;
        app.AutocenterOffButton.BackgroundColor = 'g';
        app.AutocenterOffButton.Text = 'Auto-center On';
    elseif app.autocenter ==1
        app.autocenter = 0;
        app.AutocenterOffButton.BackgroundColor = 'r';
        app.AutocenterOffButton.Text = 'Auto-center Off';
    end
end


% Button pushed function: ShimmingButton
function Shimming_CB(app, event)
    for  Gmax = 1 % 0.03 : -0.01 : 0.01
        N = 4;


        if 0
            G3 = shimming_protocol(Gmax, N); % list of gradients
        else
            G3_ = [0 0 0; 0 0 1; 0 1 0; 1 0 0];
            G3 = Gmax* [G3_; -G3_];
        end
```

```matlab
%%

% All Gradients
[Gx_S, Gy_S, Gz_S] = Read_Gxyz(app); % shimmin
%     G3 = current_block.G3;
RS_Freq = str2double(app.Scan_Frequency.Value);
%app.SetScanAmplitude_CB; % CB function
if_show  = 0;


Gpp = str2double(app.RS_Amp_DropDown.Value);
if_show = 1;
Bpp = set_Scan_Amplitude(app, Gpp, if_show ,1);




%       Actual = set_Scan_Amplitude(app, Gpp, if_show);


%       Bpp = app.Actual_Gpp;
pars.phases = [app.Phase_RS.Value, app.Phase_RF.Value];


%    p = Load_Spectra_Protocol(N_spectra, RS_Freq);
dt = app.settings_U1084.sampl_interval;




LW = zeros(1,size(G3,1));
Pos = zeros(1,size(G3,1));


options = optimoptions('lsqcurvefit','Algorithm','levenberg-marquardt');
```

```matlab
            options.StepTolerance=1e-9;

            options.FunctionTolerance = 1e-9;

            options.OptimalityTolerance = 1e-9;

            SpectralModel = app.ProbeDropDown.Value;

            LW_min = 1e10;


            for s = 1 : size(G3,1)

                g3 = G3(s,:); % gradinets x y z

                value = [0, Gx_S + g3(1), Gy_S + g3(2), Gz_S + g3(3)];

                send_dev(app.obj_NI,'gradients',value, app.settings_NI);

                pause(0.1); % replace  later by smthg userfull /datasaving ...
serail ...

                rs_data = get_dev(app.obj_U1084,'U1084','data', app.settings_U1084);

                t = (1: length(rs_data)) * dt;

                pars.fig = 'None';

                pars.Nh = [2, 2];

                pars.polynomB = [20,1];

                pars.fwhm = app.LWmin.Value;

                [h, SA, SB, pars]=RsEPR2Gen(t,rs_data,Bpp,RS_Freq,pars);

                SAB = SA + SB;

                A = double ( sum(SAB) * (h(2) -h(1)) );

                if A < 0 ; A = 0; end

                isGetFitLine  = 1; % compute Y fit line

                Ampl_guess =   A; % no Guess

                [pp, new_sp, ~] = Fit_1D(h, SAB, Ampl_guess, SpectralModel,
isGetFitLine, options);


                LW(s) = round(pp(2) *1000);

                Pos(s) = pp(3);

                disp(s);
```

238

```matlab
                    plot_Spectra(app,h,SA,SB,new_sp, rs_data);
                    LW_t = ['FWHM (Lor) = ' num2str(LW(s), '%.2f') '; Pos = '
    num2str(pp(3), '%.3f')];
                    title(app.Ax_1,LW_t);


                    if LW(s) < LW_min
                        LW_min = LW(s);
                        SA_min = SA;
                        SB_min = SB;
                        S_fit = new_sp;
                        si = s;
                    end


            end
            plot_Spectra(app,h,SA_min,SB_min,S_fit, rs_data);
            LW_t = ['FWHM (Lor) = ' num2str(LW_min, '%.2f') '; Pos = '
num2str(pp(3), '%.3f')];
            title(app.Ax_1,LW_t);
            [lw, in ] =  min(LW);
            app.GxEditField.Value = round(app.GxEditField.Value + G3(in,1),2);
            app.GyEditField.Value = round(app.GyEditField.Value + G3(in,2),2);
            app.GzEditField.Value = round(app.GzEditField.Value + G3(in,3),2);
            app.CommentsTextArea.Value = num2str(round(Pos,3));
            Gx_S =  app.GxEditField.Value;
            Gy_S =  app.GyEditField.Value;
            Gz_S =  app.GzEditField.Value;
            % [Gx_S, Gy_S, Gz_S] = Read_Gxyz(app); % shimming
            value = [0, Gx_S , Gy_S , Gz_S ];
            send_dev(app.obj_NI,'gradients',value, app.settings_NI);
```

```matlab
            Pos
        end
        disp('Shimmimg done');


        % <<< LOOP over Block



    end


    % Button pushed function: Sent_G0
    function SetG0_CB(app, event)


        value = [0, 0, 0, 0];
        app.GxEditField.Value = 0;
        app.GyEditField.Value = 0;
        app.GzEditField.Value = 0;
        send_dev(app.obj_NI,'gradients',value, app.settings_NI);
    end


    % Callback function
    function CollectBackgroundButtonPushed(app, event)
        %collect
        send_dev(app.obj_magnet,'magnet', 0 , app.settings_magnet); % Field OFF
        pause(0.5);
        rs_data = get_dev(app.obj_U1084,'U1084','data', app.settings_U1084);
        dt = app.settings_U1084.sampl_interval;
        RS_Freq = str2double(app.Scan_Frequency.Value);
        send_dev(app.obj_magnet,'magnet',app.Magnetic_Field,app.settings_magnet);
%return field to previous
        %compress
```

```matlab
    [dt,v1,dv,v_end,bg_] = Compress_RS_data(dt,rs_data,RS_Freq);

    %decompress

    vectorv = [v1,dv,v_end];

    v = vectorv(1) : vectorv(2) : vectorv(3);

    in = abs(v)> 7 * vectorv(2);

    bg_(in) = 0;

    [t, app.rs_bg] = decompress(app, vectorv, bg_);

end


% Value changed function: Phase_RF

function Phase_RFValueChanged(app, event)

    %  app.BackgroundsubtractionCheckBox.Value = 0;

    app.rs_bg = 0;

end


% Value changed function: Phase_RS

function Phase_RSValueChanged(app, event)

    %app.BackgroundsubtractionCheckBox.Value = 0;

    app.rs_bg = 0;

end


% Value changed function: Naver_V

function Naver_VValueChanged(app, event)

    %app.BackgroundsubtractionCheckBox.Value = 0;

    app.rs_bg = 0;

end


% Value changed function: Scan_Frequency

function XHnage_RS_frequnecy(app, event)

    value = app.Scan_Frequency.Value;
```

```matlab
%  C = round(str2double(value)/1000) ;
List_F = zeros(1,3);
for n = 1 : 5
    List_F(n) = str2double( app.Scan_Frequency.Items{n} );
end
inx = List_F == str2double( value );
% CC = [0.155 1 0.4]; % coils constants_list
app.Pico_calibration.Value = app.CC(inx);


send_dev(app.obj_funGen,'Fun_Gen_Freq',value,  ' ');


%               switch C
%
%                       case 8 % around Hz
%                               app.pico_Calib_All =
%                               app.Pico_calibration.Value; % ?
%                               /8130*value;
%
%                       case 12
%
%                       case 16
%                                app.pico_Calib_All = app.Pico_calibration.Value * 2;
% ? /8130*value;
%
%                       case 17
%                                app.pico_Calib_All = app.Pico_calibration.Value * 2;
% ? /8130*value;
%               end
```

```matlab
        app.Run_Spectra;


    end


    % Button pushed function: BG_ReNew_Button
    function Collect_Background(app, event)
        %collect
        send_dev(app.obj_magnet,'magnet', 0 , app.settings_magnet); % Field OFF
        pause(0.5);
        rs_data = get_dev(app.obj_U1084,'U1084','data', app.settings_U1084);
        dt = app.settings_U1084.sampl_interval;
        RS_Freq = str2double(app.Scan_Frequency.Value);
        send_dev(app.obj_magnet,'magnet',app.Magnetic_Field,app.settings_magnet);
%return field to previous
        %compress
        [dt,v1,dv,v_end,bg_] = Compress_RS_data(dt,rs_data,RS_Freq);
        %decompress
        vectorv = [v1,dv,v_end];
        v = vectorv(1) : vectorv(2) : vectorv(3);
        in = abs(v)> 7 * vectorv(2);


        [t, BG_all] = decompress(app, vectorv, bg_);
        bg_(in) = 0;
        [t, BG_harm] = decompress(app, vectorv, bg_);


        bg = BG_all -  BG_harm;
        L = length(bg);


        %               M = round(0.8*L);
```

243

```matlab
%                N = L - M;
%
i1 = t < 0.8*t(end);
%                y = bg(i1);
order = 10;
framelen = 2*round(L/100) +1;


A = sgolayfilt(bg(i1),order,framelen);
B = sgolayfilt(bg(~i1),order/2,framelen);
AB = [ A; B];




%                PL = 40;
%                p = polyfit(t(j1)',bg(j1), PL);
%                Poly_bg = polyval(p,t(j1)');
%                bg(j1) = bg(j1) + Poly_bg;




%                app.rs_bg = BG_harm + Poly_bg;
whos AB BG_harm
app.rs_bg =AB + BG_harm;


%  plotc(t,bg_2  )


plot(app.Ax_2,t,real(BG_all),t,real(app.rs_bg),t,real(app.rs_bg -
```

```matlab
BG_all),'linewidth',2 );
        plot(app.Ax_3,t,imag(BG_all),t,imag(app.rs_bg),t,imag(app.rs_bg -
BG_all),'linewidth',2 );
        pause(1);


        %
    end


    % Button pushed function: BG_Zero_Button
    function Zero_BG(app, event)
        app.rs_bg = 0;
    end


    % Value changed function: Start_Imaging_3D
    function Run_3D_imaging(app, event)
        %   value = app.Start_Imaging_3D.Value;


        if app.Start_Imaging_3D.Value % start
            app.Start_Imaging_3D.BackgroundColor = [0, 1 , 0];
            app.Start_Imaging_3D.Text = '3D >>';
            if ~app.SaveDataChk.Value
                app.SaveDataChk.Value = 1;
                warndlg('Changing status to "Save Image" ... 3 secs to cancel');
                pause(3);
            end


            Run_imaging(app,'3D');


            %   New_Imaging(app);
        else % stop
```

```matlab
            app.Start_Imaging_3D.BackgroundColor = 0.5* [1, 1 , 1];

            app.Start_Imaging_3D.Text = 'Start 3D';

        end


    end


    % Selection changed function: ButtonGroup_4

    function User_(app, event)

        selectedButton = app.ButtonGroup_4.SelectedObject;

        P = 'C:\Data_2023\';

        switch selectedButton.Text


            case 'M'

                app.FolderName.Value = [P ,'Mark\'];

            case 'O'

                app.FolderName.Value = [P ,'Oxana\'];

            case 'R'

                app.FolderName.Value = [P ,'Ryan\'];

            case 'D'

                app.FolderName.Value = [P ,'Duncan\'];

            case 'S'

                app.FolderName.Value = [P ,'Sajad\'];


        end

        disp('user changed');


    end

end


% Component initialization
```

```matlab
methods (Access = private)


    % Create UIFigure and components
    function createComponents(app)


        % Create UIFigure and hide until all components are created
        app.UIFigure = uifigure('Visible', 'off');
        app.UIFigure.Color = [0.651 0.651 0.651];
        app.UIFigure.Position = [100 100 1372 1135];
        app.UIFigure.Name = 'MATLAB App';
        app.UIFigure.CloseRequestFcn = createCallbackFcn(app,
@UIFigureCloseRequest, true);


        % Create TabGroup
        app.TabGroup = uitabgroup(app.UIFigure);
        app.TabGroup.Position = [10 65 1321 1062];


        % Create UsermodeTab
        app.UsermodeTab = uitab(app.TabGroup);
        app.UsermodeTab.Title = 'User mode';


        % Create Ax_3
        app.Ax_3 = uiaxes(app.UsermodeTab);
        title(app.Ax_3, '1')
        xlabel(app.Ax_3, 'X')
        ylabel(app.Ax_3, 'Y')
        zlabel(app.Ax_3, 'Z')
        app.Ax_3.FontSize = 18;
        app.Ax_3.Position = [18 771 698 265];
```

```
% Create Ax_2

app.Ax_2 = uiaxes(app.UsermodeTab);

title(app.Ax_2, '2')

xlabel(app.Ax_2, 'X')

ylabel(app.Ax_2, 'Y')

zlabel(app.Ax_2, 'Z')

app.Ax_2.FontSize = 18;

app.Ax_2.Position = [30 486 686 270];


% Create Ax_1

app.Ax_1 = uiaxes(app.UsermodeTab);

title(app.Ax_1, '3')

xlabel(app.Ax_1, 'X')

ylabel(app.Ax_1, 'Y')

zlabel(app.Ax_1, 'Z')

app.Ax_1.FontSize = 20;

app.Ax_1.Position = [18 1 1284 471];


% Create Panel

app.Panel = uipanel(app.UsermodeTab);

app.Panel.BackgroundColor = [1 1 1];

app.Panel.FontSize = 22;

app.Panel.Position = [731 929 100 76];


% Create RS_Amp_DropDown

app.RS_Amp_DropDown = uidropdown(app.Panel);

app.RS_Amp_DropDown.Items = {'0.5', '1', '1.5', '2', '3',
'4', '5', '6', '7', '8', '9', '10', '11', '12', '13', '14',
'15', '16', '18', '20', '22', '24', '26'};

app.RS_Amp_DropDown.ValueChangedFcn = createCallbackFcn(app,
```

```
@SetScanAmplitude_CB, true);

        app.RS_Amp_DropDown.FontSize = 20;

        app.RS_Amp_DropDown.Position = [15 9 63 30];

        app.RS_Amp_DropDown.Value = '1';


        % Create ScanGppLabel

        app.ScanGppLabel = uilabel(app.Panel);

        app.ScanGppLabel.FontSize = 16;

        app.ScanGppLabel.Position = [9 48 86 22];

        app.ScanGppLabel.Text = 'Scan [Gpp]';


        % Create Panel_3

        app.Panel_3 = uipanel(app.UsermodeTab);

        app.Panel_3.FontSize = 16;

        app.Panel_3.Position = [736 488 566 172];


        % Create FolderEditFieldLabel

        app.FolderEditFieldLabel = uilabel(app.Panel_3);

        app.FolderEditFieldLabel.HorizontalAlignment = 'right';

        app.FolderEditFieldLabel.FontSize = 14;

        app.FolderEditFieldLabel.FontWeight = 'bold';

        app.FolderEditFieldLabel.Position = [1 106 52 22];

        app.FolderEditFieldLabel.Text = ' Folder';


        % Create FolderName

        app.FolderName = uieditfield(app.Panel_3, 'text');

        app.FolderName.FontSize = 14;

        app.FolderName.Position = [78 106 357 22];

        app.FolderName.Value = 'C:\DATA_2023\';
```

```
% Create CommentsTextArea
app.CommentsTextArea = uitextarea(app.Panel_3);
app.CommentsTextArea.FontSize = 14;
app.CommentsTextArea.Position = [9 54 539 44];
app.CommentsTextArea.Value = {'Comments:'};


% Create F_nameLabel
app.F_nameLabel = uilabel(app.Panel_3);
app.F_nameLabel.HorizontalAlignment = 'right';
app.F_nameLabel.FontSize = 14;
app.F_nameLabel.FontWeight = 'bold';
app.F_nameLabel.Position = [10 143 58 22];
app.F_nameLabel.Text = 'F_name';


% Create FileName
app.FileName = uieditfield(app.Panel_3, 'text');
app.FileName.FontSize = 14;
app.FileName.Position = [76 143 359 22];
app.FileName.Value = 'file_name';


% Create SaveDataChk
app.SaveDataChk = uicheckbox(app.Panel_3);
app.SaveDataChk.Text = 'Save? ';
app.SaveDataChk.FontSize = 20;
app.SaveDataChk.Position = [467 143 84 23];


% Create ScreenButton
app.ScreenButton = uibutton(app.Panel_3, 'push');
app.ScreenButton.ButtonPushedFcn = createCallbackFcn(app,
@take_screenshot, true);
```

```matlab
        app.ScreenButton.BackgroundColor = [0 1 0];

        app.ScreenButton.FontSize = 18;

        app.ScreenButton.FontColor = [0 0 1];

        app.ScreenButton.Position = [467 108 69 29];

        app.ScreenButton.Text = 'Screen';


        % Create Comments_Image_Info

        app.Comments_Image_Info = uitextarea(app.Panel_3);

        app.Comments_Image_Info.FontSize = 14;

        app.Comments_Image_Info.Position = [11 8 539 35];

        app.Comments_Image_Info.Value = {'Image pars info'};


        % Create Panel_5

        app.Panel_5 = uipanel(app.UsermodeTab);

        app.Panel_5.BackgroundColor = [1 1 1];

        app.Panel_5.Position = [737 792 565 41];


        % Create GxEditFieldLabel

        app.GxEditFieldLabel = uilabel(app.Panel_5);

        app.GxEditFieldLabel.HorizontalAlignment = 'right';

        app.GxEditFieldLabel.FontSize = 18;

        app.GxEditFieldLabel.Position = [6 10 29 22];

        app.GxEditFieldLabel.Text = 'Gx';


        % Create GxEditField

        app.GxEditField = uieditfield(app.Panel_5, 'numeric');

        app.GxEditField.Limits = [-5 5];

        app.GxEditField.ValueDisplayFormat = '%.2f';

        app.GxEditField.ValueChangedFcn = createCallbackFcn(app,
@Send_shimming_2NI, true);
```

```matlab
app.GxEditField.FontSize = 18;

app.GxEditField.Position = [50 9 58 23];


% Create GyEditFieldLabel

app.GyEditFieldLabel = uilabel(app.Panel_5);

app.GyEditFieldLabel.HorizontalAlignment = 'right';

app.GyEditFieldLabel.FontSize = 18;

app.GyEditFieldLabel.Position = [119 11 29 22];

app.GyEditFieldLabel.Text = 'Gy';


% Create GyEditField

app.GyEditField = uieditfield(app.Panel_5, 'numeric');

app.GyEditField.Limits = [-5 5];

app.GyEditField.ValueDisplayFormat = '%2.3g';

app.GyEditField.ValueChangedFcn = createCallbackFcn(app,
@Send_shimming_2NI, true);

app.GyEditField.FontSize = 18;

app.GyEditField.Position = [163 10 56 23];


% Create GzEditFieldLabel

app.GzEditFieldLabel = uilabel(app.Panel_5);

app.GzEditFieldLabel.HorizontalAlignment = 'right';

app.GzEditFieldLabel.FontSize = 18;

app.GzEditFieldLabel.Position = [232 11 29 22];

app.GzEditFieldLabel.Text = 'Gz';


% Create GzEditField

app.GzEditField = uieditfield(app.Panel_5, 'numeric');

app.GzEditField.Limits = [-5 5];

app.GzEditField.ValueDisplayFormat = '%2.3g';
```

```matlab
        app.GzEditField.ValueChangedFcn = createCallbackFcn(app,
@Send_shimming_2NI, true);

        app.GzEditField.FontSize = 18;

        app.GzEditField.Position = [276 10 51 23];


        % Create ShimmingButton

        app.ShimmingButton = uibutton(app.Panel_5, 'push');

        app.ShimmingButton.ButtonPushedFcn = createCallbackFcn(app,
@Shimming_CB, true);

        app.ShimmingButton.BackgroundColor = [0 1 0];

        app.ShimmingButton.FontSize = 18;

        app.ShimmingButton.FontColor = [0 0 1];

        app.ShimmingButton.Position = [450 9 101 29];

        app.ShimmingButton.Text = ' Shimming';


        % Create Send2NI

        app.Send2NI = uibutton(app.Panel_5, 'push');

        app.Send2NI.ButtonPushedFcn = createCallbackFcn(app,
@Send_shimming_2NI, true);

        app.Send2NI.BackgroundColor = [0 1 1];

        app.Send2NI.FontSize = 16;

        app.Send2NI.Position = [400 8 40 26];

        app.Send2NI.Text = 'Set';


        % Create Sent_G0

        app.Sent_G0 = uibutton(app.Panel_5, 'push');

        app.Sent_G0.ButtonPushedFcn = createCallbackFcn(app, @SetG0_CB,
true);

        app.Sent_G0.BackgroundColor = [0 1 1];

        app.Sent_G0.FontSize = 16;
```

```matlab
app.Sent_G0.Position = [343 8 46 26];

app.Sent_G0.Text = 'G=0';


% Create Panel_4

app.Panel_4 = uipanel(app.UsermodeTab);

app.Panel_4.BackgroundColor = [1 1 1];

app.Panel_4.Position = [737 676 565 98];


% Create ImagProtocolLabel

app.ImagProtocolLabel = uilabel(app.Panel_4);

app.ImagProtocolLabel.HorizontalAlignment = 'right';

app.ImagProtocolLabel.FontSize = 16;

app.ImagProtocolLabel.FontColor = [0 0 1];

app.ImagProtocolLabel.Position = [22 66 64 22];

app.ImagProtocolLabel.Text = 'Protocol';


% Create Protocol_imaging

app.Protocol_imaging = uidropdown(app.Panel_4);

app.Protocol_imaging.Items = {'FBP', 'FBPHG', 'HG_Y', ''};

app.Protocol_imaging.FontSize = 16;

app.Protocol_imaging.Position = [16 34 92 22];

app.Protocol_imaging.Value = 'FBPHG';


% Create GmaxGcmButtonGroup

app.GmaxGcmButtonGroup = uibuttongroup(app.Panel_4);

app.GmaxGcmButtonGroup.ForegroundColor = [0 0 1];

app.GmaxGcmButtonGroup.Title = 'Gmax, G/cm';

app.GmaxGcmButtonGroup.FontSize = 16;

app.GmaxGcmButtonGroup.Position = [249 32 127 54];
```

254

```
% Create gmax_3

app.gmax_3 = uiradiobutton(app.GmaxGcmButtonGroup);

app.gmax_3.Text = '3';

app.gmax_3.FontSize = 20;

app.gmax_3.Position = [1 0 58 23];

app.gmax_3.Value = true;


% Create gmax_5

app.gmax_5 = uiradiobutton(app.GmaxGcmButtonGroup);

app.gmax_5.Text = '5';

app.gmax_5.FontSize = 20;

app.gmax_5.Position = [83 1 36 23];


% Create gmax_4

app.gmax_4 = uiradiobutton(app.GmaxGcmButtonGroup);

app.gmax_4.Text = '4';

app.gmax_4.FontSize = 20;

app.gmax_4.Position = [40 1 31 23];


% Create ImsLabel

app.ImsLabel = uilabel(app.Panel_4);

app.ImsLabel.HorizontalAlignment = 'right';

app.ImsLabel.FontSize = 16;

app.ImsLabel.FontColor = [0 0 1];

app.ImsLabel.Position = [184 63 45 22];

app.ImsLabel.Text = '# Ims';


% Create Number_of_Imagers

app.Number_of_Imagers = uieditfield(app.Panel_4, 'numeric');

app.Number_of_Imagers.HorizontalAlignment = 'center';
```

```matlab
        app.Number_of_Imagers.FontSize = 20;

        app.Number_of_Imagers.Position = [187 32 40 25];

        app.Number_of_Imagers.Value = 2;


        % Create Start_Imaging_4D

        app.Start_Imaging_4D = uibutton(app.Panel_4, 'state');

        app.Start_Imaging_4D.ValueChangedFcn = createCallbackFcn(app,
@Run_Stop_Imaging, true);

        app.Start_Imaging_4D.Text = ' 4D';

        app.Start_Imaging_4D.FontSize = 16;

        app.Start_Imaging_4D.Position = [498 25 43 26];


        % Create NewImagingButton

        app.NewImagingButton = uibutton(app.Panel_4, 'push');

        app.NewImagingButton.ButtonPushedFcn = createCallbackFcn(app,
@New_imaging, true);

        app.NewImagingButton.BackgroundColor = [0.902 0.902 0.902];

        app.NewImagingButton.FontSize = 16;

        app.NewImagingButton.Position = [434 61 109 26];

        app.NewImagingButton.Text = 'New Imaging';


        % Create CheckBox_Kinetics

        app.CheckBox_Kinetics = uicheckbox(app.Panel_4);

        app.CheckBox_Kinetics.Text = '';

        app.CheckBox_Kinetics.Position = [143 29 25 22];


        % Create Image

        app.Image = uiimage(app.Panel_4);

        app.Image.Position = [138 55 29 31];

        app.Image.ImageSource = 'runner.JPG';
```

```matlab
% Create Start_Imaging_3D
app.Start_Imaging_3D = uibutton(app.Panel_4, 'state');
app.Start_Imaging_3D.ValueChangedFcn = createCallbackFcn(app,
@Run_3D_imaging, true);
app.Start_Imaging_3D.Text = ' 3D';
app.Start_Imaging_3D.FontSize = 16;
app.Start_Imaging_3D.Position = [443 23 43 26];


% Create NaverLabel
app.NaverLabel = uilabel(app.UsermodeTab);
app.NaverLabel.HorizontalAlignment = 'right';
app.NaverLabel.FontSize = 16;
app.NaverLabel.Position = [1155 939 77 22];
app.NaverLabel.Text = '# Average';


% Create Naver_V
app.Naver_V = uieditfield(app.UsermodeTab, 'numeric');
app.Naver_V.Limits = [1 Inf];
app.Naver_V.RoundFractionalValues = 'on';
app.Naver_V.ValueChangedFcn = createCallbackFcn(app,
@Naver_VValueChanged, true);
app.Naver_V.HorizontalAlignment = 'center';
app.Naver_V.FontSize = 20;
app.Naver_V.Position = [1241 936 57 25];
app.Naver_V.Value = 250;


% Create RunStopButton
app.RunStopButton = uibutton(app.UsermodeTab, 'state');
app.RunStopButton.ValueChangedFcn = createCallbackFcn(app,
```

```
@Run_Stop_Spectra, true);
        app.RunStopButton.Text = 'Start Spectra';

        app.RunStopButton.BackgroundColor = [0 0 1];

        app.RunStopButton.FontSize = 20;

        app.RunStopButton.FontColor = [1 1 1];

        app.RunStopButton.Position = [738 871 132 31];


        % Create Number_of_spectra
        app.Number_of_spectra = uieditfield(app.UsermodeTab, 'numeric');

        app.Number_of_spectra.HorizontalAlignment = 'center';

        app.Number_of_spectra.FontSize = 20;

        app.Number_of_spectra.Position = [1248 974 50 25];

        app.Number_of_spectra.Value = 20;


        % Create Panel_8
        app.Panel_8 = uipanel(app.UsermodeTab);

        app.Panel_8.BackgroundColor = [1 1 1];

        app.Panel_8.FontSize = 16;

        app.Panel_8.Position = [839 930 306 72];


        % Create PhRFLabel
        app.PhRFLabel = uilabel(app.Panel_8);

        app.PhRFLabel.HorizontalAlignment = 'right';

        app.PhRFLabel.FontSize = 16;

        app.PhRFLabel.Position = [8 37 57 22];

        app.PhRFLabel.Text = 'Ph(RF)';


        % Create Phase_RF
        app.Phase_RF = uieditfield(app.Panel_8, 'numeric');

        app.Phase_RF.ValueChangedFcn = createCallbackFcn(app,
```

```matlab
@Phase_RFValueChanged, true);

        app.Phase_RF.HorizontalAlignment = 'center';

        app.Phase_RF.FontSize = 20;

        app.Phase_RF.Position = [9 3 56 25];


        % Create Phase_RS

        app.Phase_RS = uieditfield(app.Panel_8, 'numeric');

        app.Phase_RS.ValueDisplayFormat = '%.1f';

        app.Phase_RS.ValueChangedFcn = createCallbackFcn(app,

@Phase_RSValueChanged, true);

        app.Phase_RS.HorizontalAlignment = 'center';

        app.Phase_RS.FontSize = 20;

        app.Phase_RS.Position = [75 4 90 25];

        app.Phase_RS.Value = 200;


        % Create PhRSLabel

        app.PhRSLabel = uilabel(app.Panel_8);

        app.PhRSLabel.HorizontalAlignment = 'right';

        app.PhRSLabel.FontSize = 16;

        app.PhRSLabel.Position = [89 37 58 22];

        app.PhRSLabel.Text = 'Ph(RS)';


        % Create AutpRS_CheckBox

        app.AutpRS_CheckBox = uicheckbox(app.Panel_8);

        app.AutpRS_CheckBox.Text = '';

        app.AutpRS_CheckBox.FontSize = 24;

        app.AutpRS_CheckBox.Position = [175 4 29 28];


        % Create AutoLabel

        app.AutoLabel = uilabel(app.Panel_8);
```

```matlab
            app.AutoLabel.HorizontalAlignment = 'right';

            app.AutoLabel.FontSize = 16;

            app.AutoLabel.Position = [164 36 43 22];

            app.AutoLabel.Text = 'Auto ';


            % Create Scan_Frequency

            app.Scan_Frequency = uidropdown(app.Panel_8);

            app.Scan_Frequency.Items = {'9710', '11900', '15520', '8240', '16440'};

            app.Scan_Frequency.ValueChangedFcn = createCallbackFcn(app,
@XHnage_RS_frequnecy, true);

            app.Scan_Frequency.FontSize = 20;

            app.Scan_Frequency.Position = [206 6 93 25];

            app.Scan_Frequency.Value = '9710';


            % Create RSFreqLabel

            app.RSFreqLabel = uilabel(app.Panel_8);

            app.RSFreqLabel.HorizontalAlignment = 'right';

            app.RSFreqLabel.FontSize = 16;

            app.RSFreqLabel.Position = [214 37 70 22];

            app.RSFreqLabel.Text = 'RS Freq.';


            % Create SpectraLabel

            app.SpectraLabel = uilabel(app.UsermodeTab);

            app.SpectraLabel.HorizontalAlignment = 'right';

            app.SpectraLabel.FontSize = 16;

            app.SpectraLabel.Position = [1157 977 74 22];

            app.SpectraLabel.Text = '# Spectra';


            % Create Panel_9

            app.Panel_9 = uipanel(app.UsermodeTab);
```

```matlab
        app.Panel_9.BackgroundColor = [1 1 1];

        app.Panel_9.Position = [1029 865 269 40];


        % Create RunStopButton_2

        app.RunStopButton_2 = uibutton(app.Panel_9, 'state');

        app.RunStopButton_2.ValueChangedFcn = createCallbackFcn(app,
@CF_Guess, true);

        app.RunStopButton_2.Text = 'Field ?';

        app.RunStopButton_2.BackgroundColor = [0 0 1];

        app.RunStopButton_2.FontSize = 20;

        app.RunStopButton_2.FontColor = [1 1 1];

        app.RunStopButton_2.Position = [5 5 74 31];


        % Create Current_Field

        app.Current_Field = uieditfield(app.Panel_9, 'numeric');

        app.Current_Field.Limits = [0 36];

        app.Current_Field.ValueDisplayFormat = '%2.2f';

        app.Current_Field.ValueChangedFcn = createCallbackFcn(app,
@Current_FieldValueChanged, true);

        app.Current_Field.HorizontalAlignment = 'center';

        app.Current_Field.FontSize = 20;

        app.Current_Field.Position = [131 7 72 25];

        app.Current_Field.Value = 10;


        % Create ProbeDropDown

        app.ProbeDropDown = uidropdown(app.UsermodeTab);

        app.ProbeDropDown.Items = {'Lorentz', 'Sox71', 'Ox71', 'Finland', 'HOPE'};

        app.ProbeDropDown.FontSize = 20;

        app.ProbeDropDown.FontColor = [0.9412 0.9412 0.9412];

        app.ProbeDropDown.BackgroundColor = [0.6353 0.0784 0.1843];
```

```matlab
app.ProbeDropDown.Position = [896 870 112 30];

app.ProbeDropDown.Value = 'Lorentz';


% Create RSData_Show_CheckBox

app.RSData_Show_CheckBox = uicheckbox(app.UsermodeTab);

app.RSData_Show_CheckBox.Text = ' RS Data';

app.RSData_Show_CheckBox.FontSize = 16;

app.RSData_Show_CheckBox.Position = [1212 467 87 22];


% Create PlotLabel_2

app.PlotLabel_2 = uilabel(app.UsermodeTab);

app.PlotLabel_2.HorizontalAlignment = 'right';

app.PlotLabel_2.FontSize = 16;

app.PlotLabel_2.Position = [1075 465 33 22];

app.PlotLabel_2.Text = 'Plot';


% Create Plot_EPR_chk

app.Plot_EPR_chk = uicheckbox(app.UsermodeTab);

app.Plot_EPR_chk.Text = '';

app.Plot_EPR_chk.FontSize = 20;

app.Plot_EPR_chk.Position = [1120 467 28 23];

app.Plot_EPR_chk.Value = true;


% Create AverLabel

app.AverLabel = uilabel(app.UsermodeTab);

app.AverLabel.HorizontalAlignment = 'right';

app.AverLabel.FontSize = 16;

app.AverLabel.Position = [968 465 42 22];

app.AverLabel.Text = 'Aver.';
```

```
% Create Average_Plots_chk

app.Average_Plots_chk = uicheckbox(app.UsermodeTab);

app.Average_Plots_chk.Text = '';

app.Average_Plots_chk.FontSize = 20;

app.Average_Plots_chk.Position = [1033 466 28 23];


% Create Label_6

app.Label_6 = uilabel(app.UsermodeTab);

app.Label_6.FontSize = 20;

app.Label_6.Position = [1013 464 25 24];

app.Label_6.Text = '<';


% Create Label_7

app.Label_7 = uilabel(app.UsermodeTab);

app.Label_7.FontSize = 20;

app.Label_7.Position = [1051 464 25 24];

app.Label_7.Text = '>';


% Create BackgroundsubtractionCheckBox

app.BackgroundsubtractionCheckBox = uicheckbox(app.UsermodeTab);

app.BackgroundsubtractionCheckBox.Text = 'Background subtraction';

app.BackgroundsubtractionCheckBox.FontSize = 16;

app.BackgroundsubtractionCheckBox.Position = [57 471 191 22];


% Create TuningTab

app.TuningTab = uitab(app.TabGroup);

app.TuningTab.Title = 'Tuning';


% Create MCbox

app.MCbox = uiaxes(app.TuningTab);
```

```matlab
        title(app.MCbox, 'Title')

        xlabel(app.MCbox, 'X')

        ylabel(app.MCbox, 'Y')

        zlabel(app.MCbox, 'Z')

        app.MCbox.Position = [16 421 1286 601];


        % Create ConnectMCButton

        app.ConnectMCButton = uibutton(app.TuningTab, 'push');

        app.ConnectMCButton.ButtonPushedFcn = createCallbackFcn(app,
@ConnectMCButtonPushed, true);

        app.ConnectMCButton.BackgroundColor = [0 1 0];

        app.ConnectMCButton.FontSize = 16;

        app.ConnectMCButton.Position = [58 341 104 52];

        app.ConnectMCButton.Text = 'Connect MC';


        % Create DisconnectMCButton

        app.DisconnectMCButton = uibutton(app.TuningTab, 'push');

        app.DisconnectMCButton.ButtonPushedFcn = createCallbackFcn(app,
@Stop_Teensy, true);

        app.DisconnectMCButton.BackgroundColor = [0.6353 0.0784 0.1843];

        app.DisconnectMCButton.FontSize = 16;

        app.DisconnectMCButton.FontColor = [1 1 1];

        app.DisconnectMCButton.Position = [184 341 124 52];

        app.DisconnectMCButton.Text = 'Disconnect MC';


        % Create TestserialcommunicationButton

        app.TestserialcommunicationButton = uibutton(app.TuningTab, 'push');

        app.TestserialcommunicationButton.ButtonPushedFcn =
createCallbackFcn(app, @Send2Teensy, true);

        app.TestserialcommunicationButton.FontSize = 16;
```

264

```matlab
        app.TestserialcommunicationButton.Position = [73 288 201 45];

        app.TestserialcommunicationButton.Text = 'Test serial communication';


        % Create testserialcombox

        app.testserialcombox = uieditfield(app.TuningTab, 'text');

        app.testserialcombox.FontSize = 16;

        app.testserialcombox.Position = [66 244 223 23];

        app.testserialcombox.Value = 'x x x x x x';


        % Create TunerangeButtonGroup

        app.TunerangeButtonGroup = uibuttongroup(app.TuningTab);

        app.TunerangeButtonGroup.SelectionChangedFcn =
createCallbackFcn(app, @TunerangeButtonGroupSelectionChanged, true);

        app.TunerangeButtonGroup.Title = 'Tune range';

        app.TunerangeButtonGroup.BackgroundColor = [0.902 0.902 0.902];

        app.TunerangeButtonGroup.FontSize = 16;

        app.TunerangeButtonGroup.Position = [559 106 173 244];


        % Create Button

        app.Button = uiradiobutton(app.TunerangeButtonGroup);

        app.Button.Text = '790-810';

        app.Button.FontSize = 16;

        app.Button.Position = [11 194 81 22];

        app.Button.Value = true;


        % Create Button_2

        app.Button_2 = uiradiobutton(app.TunerangeButtonGroup);

        app.Button_2.Text = '800-820';

        app.Button_2.FontSize = 16;

        app.Button_2.Position = [11 172 81 22];
```

```
% Create Button_3
app.Button_3 = uiradiobutton(app.TunerangeButtonGroup);
app.Button_3.Text = '810-830';
app.Button_3.FontSize = 16;
app.Button_3.Position = [11 150 81 22];


% Create GapsButton
app.GapsButton = uiradiobutton(app.TunerangeButtonGroup);
app.GapsButton.Text = 'Gaps';
app.GapsButton.FontSize = 16;
app.GapsButton.Position = [22 28 60 22];


% Create Button_770
app.Button_770 = uiradiobutton(app.TunerangeButtonGroup);
app.Button_770.Text = '770-790';
app.Button_770.FontSize = 16;
app.Button_770.Position = [11 115 81 22];


% Create Button_780
app.Button_780 = uiradiobutton(app.TunerangeButtonGroup);
app.Button_780.Text = '780-800';
app.Button_780.FontSize = 16;
app.Button_780.Position = [11 90 81 22];


% Create kHzButton
app.kHzButton = uibutton(app.TuningTab, 'push');
app.kHzButton.ButtonPushedFcn = createCallbackFcn(app, @minus50kHz, true);
app.kHzButton.BackgroundColor = [0.651 0.651 0.651];
app.kHzButton.FontSize = 16;
```

```matlab
app.kHzButton.Position = [865 250 100 26];

app.kHzButton.Text = '-50kHz';


% Create kHzButton_2

app.kHzButton_2 = uibutton(app.TuningTab, 'push');

app.kHzButton_2.ButtonPushedFcn = createCallbackFcn(app, @Plus50kHz, true);

app.kHzButton_2.BackgroundColor = [0.651 0.651 0.651];

app.kHzButton_2.FontSize = 16;

app.kHzButton_2.Position = [1000 250 100 26];

app.kHzButton_2.Text = '+50kHz';


% Create SetonceButton

app.SetonceButton = uibutton(app.TuningTab, 'push');

app.SetonceButton.ButtonPushedFcn = createCallbackFcn(
app, @Set_Get_Reflection, true);

app.SetonceButton.BackgroundColor = [0 0.4471 0.7412];

app.SetonceButton.FontSize = 16;

app.SetonceButton.FontWeight = 'bold';

app.SetonceButton.FontColor = [1 1 1];

app.SetonceButton.Position = [492 372 100 26];

app.SetonceButton.Text = 'Set once';


% Create SmartTuneButton

app.SmartTuneButton = uibutton(app.TuningTab, 'push');

app.SmartTuneButton.ButtonPushedFcn = createCallbackFcn(app,
@Smart_Tune, true);

app.SmartTuneButton.BackgroundColor = [0.4667 0.6745 0.1882];

app.SmartTuneButton.FontSize = 16;

app.SmartTuneButton.FontWeight = 'bold';

app.SmartTuneButton.FontColor = [1 1 1];
```

```matlab
app.SmartTuneButton.Position = [644 372 102 26];

app.SmartTuneButton.Text = 'Smart Tune';


% Create PlottunepatternCheckBox

app.PlottunepatternCheckBox = uicheckbox(app.TuningTab);

app.PlottunepatternCheckBox.Text = 'Plot tune pattern';

app.PlottunepatternCheckBox.FontSize = 16;

app.PlottunepatternCheckBox.Position = [873 374 139 22];


% Create DTC1EditFieldLabel

app.DTC1EditFieldLabel = uilabel(app.TuningTab);

app.DTC1EditFieldLabel.HorizontalAlignment = 'right';

app.DTC1EditFieldLabel.FontSize = 16;

app.DTC1EditFieldLabel.Position = [74 185 47 22];

app.DTC1EditFieldLabel.Text = 'DTC1';


% Create DTC1

app.DTC1 = uieditfield(app.TuningTab, 'numeric');

app.DTC1.Limits = [0 32];

app.DTC1.RoundFractionalValues = 'on';

app.DTC1.FontSize = 16;

app.DTC1.Position = [136 185 31 22];


% Create DTC2EditFieldLabel

app.DTC2EditFieldLabel = uilabel(app.TuningTab);

app.DTC2EditFieldLabel.HorizontalAlignment = 'right';

app.DTC2EditFieldLabel.FontSize = 16;

app.DTC2EditFieldLabel.Position = [227 184 47 22];

app.DTC2EditFieldLabel.Text = 'DTC2';
```

```
% Create DTC2

app.DTC2 = uieditfield(app.TuningTab, 'numeric');

app.DTC2.Limits = [0 32];

app.DTC2.RoundFractionalValues = 'on';

app.DTC2.FontSize = 16;

app.DTC2.Position = [289 184 31 22];


% Create RFMHzEditFieldLabel

app.RFMHzEditFieldLabel = uilabel(app.TuningTab);

app.RFMHzEditFieldLabel.HorizontalAlignment = 'right';

app.RFMHzEditFieldLabel.FontSize = 16;

app.RFMHzEditFieldLabel.Position = [882 328 67 22];

app.RFMHzEditFieldLabel.Text = 'RF, MHz';


% Create CW_Freq

app.CW_Freq = uieditfield(app.TuningTab, 'numeric');

app.CW_Freq.FontSize = 16;

app.CW_Freq.Position = [964 328 100 22];

app.CW_Freq.Value = 800;


% Create FineCWControlLabel

app.FineCWControlLabel = uilabel(app.TuningTab);

app.FineCWControlLabel.FontSize = 16;

app.FineCWControlLabel.Position = [917 291 124 22];

app.FineCWControlLabel.Text = 'Fine CW Control';


% Create ReflThershEditFieldLabel

app.ReflThershEditFieldLabel = uilabel(app.TuningTab);

app.ReflThershEditFieldLabel.HorizontalAlignment = 'right';

app.ReflThershEditFieldLabel.FontSize = 18;
```

```matlab
app.ReflThershEditFieldLabel.Position = [1076 325 98 22];

app.ReflThershEditFieldLabel.Text = 'Refl Thersh';


% Create ReflThershEditField

app.ReflThershEditField = uieditfield(app.TuningTab, 'numeric');

app.ReflThershEditField.FontSize = 18;

app.ReflThershEditField.Position = [1180 327 100 23];

app.ReflThershEditField.Value = 130;


% Create CalibrationsTab

app.CalibrationsTab = uitab(app.TabGroup);

app.CalibrationsTab.Title = 'Calibrations';


% Create PicoScopeTextAreaLabel

app.PicoScopeTextAreaLabel = uilabel(app.CalibrationsTab);

app.PicoScopeTextAreaLabel.HorizontalAlignment = 'right';

app.PicoScopeTextAreaLabel.Position = [516 998 63 22];

app.PicoScopeTextAreaLabel.Text = 'PicoScope';


% Create PicoScopeTextArea

app.PicoScopeTextArea = uitextarea(app.CalibrationsTab);

app.PicoScopeTextArea.FontSize = 18;

app.PicoScopeTextArea.Position = [594 962 474 60];

app.PicoScopeTextArea.Value = {'8130 Hz : PIco : 0.265'};


% Create AWG_calibration

app.AWG_calibration = uieditfield(app.CalibrationsTab, 'numeric');

app.AWG_calibration.FontSize = 18;

app.AWG_calibration.Position = [239 701 100 23];

app.AWG_calibration.Value = 0.088;
```

```matlab
% Create AWGGppLabel
app.AWGGppLabel = uilabel(app.CalibrationsTab);

app.AWGGppLabel.FontSize = 20;

app.AWGGppLabel.Position = [40 700 136 24];

app.AWGGppLabel.Text = 'AWG  <> Gpp ';


% Create ExpertmodeTab
app.ExpertmodeTab = uitab(app.TabGroup);

app.ExpertmodeTab.Title = 'Expert mode';


% Create ButtonGroup_2
app.ButtonGroup_2 = uibuttongroup(app.ExpertmodeTab);

app.ButtonGroup_2.FontSize = 20;

app.ButtonGroup_2.Position = [495 332 123 106];


% Create UpButton
app.UpButton = uiradiobutton(app.ButtonGroup_2);

app.UpButton.Text = 'Up';

app.UpButton.FontSize = 20;

app.UpButton.Position = [11 78 48 23];


% Create DownButton
app.DownButton = uiradiobutton(app.ButtonGroup_2);

app.DownButton.Text = 'Down';

app.DownButton.FontSize = 20;

app.DownButton.Position = [8 42 73 23];


% Create SumButton
app.SumButton = uiradiobutton(app.ButtonGroup_2);
```

```matlab
app.SumButton.Text = 'Sum';

app.SumButton.FontSize = 20;

app.SumButton.Position = [12 11 65 23];

app.SumButton.Value = true;


% Create Scan_Frequency_old
app.Scan_Frequency_old = uidropdown(app.ExpertmodeTab);

app.Scan_Frequency_old.Items = {'9200', '8130', '9400', '5180'};

app.Scan_Frequency_old.FontSize = 20;

app.Scan_Frequency_old.Position = [215 980 77 25];

app.Scan_Frequency_old.Value = '8130';


% Create ScanFrequnecyHzLabel
app.ScanFrequnecyHzLabel = uilabel(app.ExpertmodeTab);

app.ScanFrequnecyHzLabel.FontSize = 20;

app.ScanFrequnecyHzLabel.Position = [18 982 185 24];

app.ScanFrequnecyHzLabel.Text = 'Scan Frequnecy, Hz';


% Create LWminLabel
app.LWminLabel = uilabel(app.ExpertmodeTab);

app.LWminLabel.HorizontalAlignment = 'right';

app.LWminLabel.FontSize = 20;

app.LWminLabel.Position = [44 938 34 24];

app.LWminLabel.Text = 'LW';


% Create LWmin
app.LWmin = uieditfield(app.ExpertmodeTab, 'numeric');

app.LWmin.HorizontalAlignment = 'center';

app.LWmin.FontSize = 20;

app.LWmin.Position = [91 937 56 25];
```

```matlab
        app.LWmin.Value = 0.01;


        % Create MaxpointformagnetfieldcorrectioniffittingissporadicCheckBox
        app.MaxpointformagnetfieldcorrectioniffittingissporadicCheckBox =
uicheckbox(app.ExpertmodeTab);
        app.MaxpointformagnetfieldcorrectioniffittingissporadicCheckBox.Text =
'Max point for magnet field correction (if fitting is sporadic)';
        app.MaxpointformagnetfieldcorrectioniffittingissporadicCheckBox.Position =
[44 900 331 22];


        % Create SettingsTab
        app.SettingsTab = uitab(app.TabGroup);
        app.SettingsTab.Title = 'Settings';


        % Create ScanParsPanel
        app.ScanParsPanel = uipanel(app.SettingsTab);
        app.ScanParsPanel.Title = 'Scan Pars';
        app.ScanParsPanel.Position = [1 870 260 166];


        % Create UpdatePlot
        app.UpdatePlot = uicheckbox(app.ScanParsPanel);
        app.UpdatePlot.Text = {'Update plot'; ''};
        app.UpdatePlot.FontSize = 16;
        app.UpdatePlot.Position = [17 113 126 26];
        app.UpdatePlot.Value = true;


        % Create FunGenReset
        app.FunGenReset = uibutton(app.ScanParsPanel, 'push');
        app.FunGenReset.FontSize = 14;
        app.FunGenReset.Position = [134 115 78 24];
```

```matlab
app.FunGenReset.Text = 'Reset';


% Create FGPhaseDegrLabel
app.FGPhaseDegrLabel = uilabel(app.ScanParsPanel);
app.FGPhaseDegrLabel.HorizontalAlignment = 'right';
app.FGPhaseDegrLabel.FontSize = 16;
app.FGPhaseDegrLabel.Position = [20 8 123 22];
app.FGPhaseDegrLabel.Text = 'FG Phase [Degr]';


% Create FunGenPhase_box
app.FunGenPhase_box = uieditfield(app.ScanParsPanel, 'numeric');
app.FunGenPhase_box.Limits = [0 360];
app.FunGenPhase_box.HorizontalAlignment = 'left';
app.FunGenPhase_box.FontSize = 16;
app.FunGenPhase_box.Position = [163 8 45 22];
app.FunGenPhase_box.Value = 130;


% Create FunGenVLabel
app.FunGenVLabel = uilabel(app.ScanParsPanel);
app.FunGenVLabel.HorizontalAlignment = 'right';
app.FunGenVLabel.FontSize = 16;
app.FunGenVLabel.Position = [43 48 94 22];
app.FunGenVLabel.Text = 'FunGen  [V].';


% Create FunGenVoltage_box
app.FunGenVoltage_box = uieditfield(app.ScanParsPanel, 'numeric');
app.FunGenVoltage_box.Limits = [0 4];
app.FunGenVoltage_box.HorizontalAlignment = 'left';
app.FunGenVoltage_box.FontSize = 16;
app.FunGenVoltage_box.Position = [149 46 35 22];
```

```matlab
app.FunGenVoltage_box.Value = 1;


% Create PicoscopeSettingsPanel
app.PicoscopeSettingsPanel = uipanel(app.SettingsTab);
app.PicoscopeSettingsPanel.Title = 'Picoscope Settings';
app.PicoscopeSettingsPanel.Position = [865 10 446 444];


% Create TriggerPanel
app.TriggerPanel = uipanel(app.PicoscopeSettingsPanel);
app.TriggerPanel.Title = 'Trigger';
app.TriggerPanel.BackgroundColor = [1 1 1];
app.TriggerPanel.Position = [210 328 230 93];


% Create Channel
app.Channel = uidropdown(app.TriggerPanel);
app.Channel.Items = {'A', 'B', 'C', 'D', 'Ex'};
app.Channel.Position = [9 43 41 22];
app.Channel.Value = 'D';


% Create TresholdmVEditFieldLabel
app.TresholdmVEditFieldLabel = uilabel(app.TriggerPanel);
app.TresholdmVEditFieldLabel.HorizontalAlignment = 'right';
app.TresholdmVEditFieldLabel.Position = [71 43 76 22];
app.TresholdmVEditFieldLabel.Text = 'Treshold, mV';


% Create Treshold
app.Treshold = uieditfield(app.TriggerPanel, 'numeric');
app.Treshold.Limits = [0 3000];
app.Treshold.RoundFractionalValues = 'on';
app.Treshold.Position = [160 43 41 22];
```

```matlab
app.Treshold.Value = 1000;


% Create DelayLabel

app.DelayLabel = uilabel(app.TriggerPanel);

app.DelayLabel.HorizontalAlignment = 'right';

app.DelayLabel.Position = [7 9 36 22];

app.DelayLabel.Text = 'Delay';


% Create Delay

app.Delay = uieditfield(app.TriggerPanel, 'numeric');

app.Delay.Position = [47 10 22 22];


% Create DirecLabel

app.DirecLabel = uilabel(app.TriggerPanel);

app.DirecLabel.HorizontalAlignment = 'right';

app.DirecLabel.Position = [78 9 34 22];

app.DirecLabel.Text = 'Direc';


% Create Direct

app.Direct = uieditfield(app.TriggerPanel, 'numeric');

app.Direct.Position = [116 10 22 22];

app.Direct.Value = 2;


% Create AutoTrLabel

app.AutoTrLabel = uilabel(app.TriggerPanel);

app.AutoTrLabel.HorizontalAlignment = 'right';

app.AutoTrLabel.Position = [143 9 41 22];

app.AutoTrLabel.Text = 'AutoTr';


% Create AutoTr
```

```matlab
        app.AutoTr = uieditfield(app.TriggerPanel, 'numeric');
        app.AutoTr.Position = [189 10 22 22];
        app.AutoTr.Value = 1;


        % Create RetrieveStart_iSeg_iRatioModeLabel
        app.RetrieveStart_iSeg_iRatioModeLabel =
uilabel(app.PicoscopeSettingsPanel);
        app.RetrieveStart_iSeg_iRatioModeLabel.HorizontalAlignment = 'right';
        app.RetrieveStart_iSeg_iRatioModeLabel.Position = [8 350 201 22];
        app.RetrieveStart_iSeg_iRatioModeLabel.Text = 'Retrieve: Start_i,
Seg_i, Ratio, Mode';


        % Create RetrieveDataTX
        app.RetrieveDataTX = uieditfield(app.PicoscopeSettingsPanel, 'text');
        app.RetrieveDataTX.Position = [16 326 162 22];
        app.RetrieveDataTX.Value = '0    0    1  0 ';


        % Create TextArea
        app.TextArea = uitextarea(app.PicoscopeSettingsPanel);
        app.TextArea.FontSize = 11;
        app.TextArea.Position = [38 37 404 279];
        app.TextArea.Value = {'%FROM: Set_Paths_Defaults_Calibration.m';
'% Trigger'; 'ps5000a.TrigChannel = 3; % 0,1,2,3,4 [A, B C D ext] ';
'ps5000a.Threshold=1000; % (mV);'; 'ps5000a.Direction=2; % Rising';
'ps5000a.Delay=0; '; 'ps5000a.AutoTrig=1; % (wait indefinitely)';
'%'; '% timebase    : 65 (default) ;   segment index: 0';
'% timebaseIndex = get(DevObj.ps5000aDeviceObj, ''timebase'');';
'% [status, timeIntervalNanoSeconds, maxSamples] =
invoke(DevObj.ps5000aDeviceObj, ''ps5000aGetTimebase2'',
timebaseIndex, 0);'; '% sampling interval = (timebase - 2)/ 125000000';
```

```
'% sampleTimePicosecond = libpointer(''int64Ptr'',zeros(1000,1));';

'ps5000a.time_base=14;'; 'ps5000a.post_samples=2*8*2048;  % post Trigger';

'ps5000a.segmentInx=0; % to Capture a block of data:';

'% Retrieve data values: '; 'ps5000a.start_index =0;';

'ps5000a.segment_index=0;'; 'ps5000a.downsampling_ratio=1;';

'ps5000a.downsampling_mode=0;'};


        % Create RESETButton

        app.RESETButton = uibutton(app.PicoscopeSettingsPanel, 'push');

        app.RESETButton.Position = [175 6 100 22];

        app.RESETButton.Text = 'RESET ';


        % Create SamplingBasePostSamSegInxEditFieldLabel

        app.SamplingBasePostSamSegInxEditFieldLabel = uilabel(app.SettingsTab);

        app.SamplingBasePostSamSegInxEditFieldLabel.HorizontalAlignment = 'right';

        app.SamplingBasePostSamSegInxEditFieldLabel.Position = [985 987 190 22];

        app.SamplingBasePostSamSegInxEditFieldLabel.Text = 'Sampling: Base,
PostSam, SegInx';


        % Create SamplingTX

        app.SamplingTX = uieditfield(app.SettingsTab, 'text');

        app.SamplingTX.Position = [992 963 162 22];

        app.SamplingTX.Value = '14   32768    0';


        % Create CalibrationPanel

        app.CalibrationPanel = uipanel(app.SettingsTab);

        app.CalibrationPanel.Title = 'Calibration';

        app.CalibrationPanel.Position = [260 870 721 166];


        % Create CoilTypeButtonGroup
```

```matlab
app.CoilTypeButtonGroup = uibuttongroup(app.CalibrationPanel);

app.CoilTypeButtonGroup.Title = 'Coil Type';

app.CoilTypeButtonGroup.Position = [10 30 123 106];


% Create LargestButton

app.LargestButton = uiradiobutton(app.CoilTypeButtonGroup);

app.LargestButton.Text = 'Largest';

app.LargestButton.Position = [11 60 62 22];

app.LargestButton.Value = true;


% Create MiddleButton

app.MiddleButton = uiradiobutton(app.CoilTypeButtonGroup);

app.MiddleButton.Text = 'Middle';

app.MiddleButton.Position = [11 38 65 22];


% Create SmallButton

app.SmallButton = uiradiobutton(app.CoilTypeButtonGroup);

app.SmallButton.Text = {'Small'; ''};

app.SmallButton.Position = [11 16 65 22];


% Create AWGPanel

app.AWGPanel = uipanel(app.CalibrationPanel);

app.AWGPanel.Title = 'AWG';

app.AWGPanel.Position = [132 30 123 106];


% Create PhaseLabel

app.PhaseLabel = uilabel(app.AWGPanel);

app.PhaseLabel.HorizontalAlignment = 'right';

app.PhaseLabel.Position = [7 59 40 22];

app.PhaseLabel.Text = 'Phase';
```

```matlab
% Create AWG_phase
app.AWG_phase = uieditfield(app.AWGPanel, 'numeric');
app.AWG_phase.Position = [55 59 46 22];


% Create AmplLabel
app.AmplLabel = uilabel(app.AWGPanel);
app.AmplLabel.HorizontalAlignment = 'right';
app.AmplLabel.Position = [9 36 36 22];
app.AmplLabel.Text = 'Ampl.';


% Create AWG_amplitude
app.AWG_amplitude = uieditfield(app.AWGPanel, 'numeric');
app.AWG_amplitude.Position = [56 34 46 22];


% Create AWG_Set
app.AWG_Set = uibutton(app.AWGPanel, 'push');
app.AWG_Set.Position = [8 5 100 22];
app.AWG_Set.Text = 'Set';


% Create TextArea_2
app.TextArea_2 = uitextarea(app.CalibrationPanel);
app.TextArea_2.Position = [324 106 353 21];
app.TextArea_2.Value = {'LARGE:  Freq_Hz    Coil Const.    Phase_Delay
Phase_AWG  '};


% Create LRG
app.LRG = uieditfield(app.CalibrationPanel, 'text');
app.LRG.Position = [276 53 142 22];
app.LRG.Value = '9400  1.26 96 135';
```

280

```
% Create LRG_2
app.LRG_2 = uieditfield(app.CalibrationPanel, 'text');
app.LRG_2.Position = [427 53 142 22];
app.LRG_2.Value = '5180  0.75 96 135';


% Create AWGSettingsPanel
app.AWGSettingsPanel = uipanel(app.SettingsTab);
app.AWGSettingsPanel.Title = 'AWG Settings';
app.AWGSettingsPanel.Position = [1 701 260 170];


% Create ResetAWGtoTuneCWstateButton
app.ResetAWGtoTuneCWstateButton = uibutton(app.AWGSettingsPanel, 'push');
app.ResetAWGtoTuneCWstateButton.FontSize = 16;
app.ResetAWGtoTuneCWstateButton.FontColor = [0 0 1];
app.ResetAWGtoTuneCWstateButton.Position = [7 80 239 28];
app.ResetAWGtoTuneCWstateButton.Text = 'Reset AWG to Tune/CW  state ';


% Create PauseButton
app.PauseButton = uibutton(app.AWGSettingsPanel, 'state');
app.PauseButton.Text = 'Pause';
app.PauseButton.BackgroundColor = [1 0.702 0];
app.PauseButton.Position = [7 121 66 22];


% Create ResetAWGAutostateButton
app.ResetAWGAutostateButton = uibutton(app.AWGSettingsPanel, 'push');
app.ResetAWGAutostateButton.FontSize = 16;
app.ResetAWGAutostateButton.FontColor = [0 0 1];
app.ResetAWGAutostateButton.Position = [7 45 242 28];
app.ResetAWGAutostateButton.Text = 'Reset AWG Auto state ';
```

```matlab
% Create ResetArduinoButton
app.ResetArduinoButton = uibutton(app.AWGSettingsPanel, 'push');
app.ResetArduinoButton.FontSize = 16;
app.ResetArduinoButton.FontColor = [0 0 1];
app.ResetArduinoButton.Position = [7 11 242 28];
app.ResetArduinoButton.Text = 'Reset Arduino';


% Create DigitizerPanel
app.DigitizerPanel = uipanel(app.SettingsTab);
app.DigitizerPanel.Title = 'Digitizer';
app.DigitizerPanel.Position = [1 0 260 702];


% Create DigitizerPanel_2
app.DigitizerPanel_2 = uipanel(app.DigitizerPanel);
app.DigitizerPanel_2.ForegroundColor = [1 1 0];
app.DigitizerPanel_2.Title = 'Digitizer';
app.DigitizerPanel_2.BackgroundColor = [0.651 0.651 0.651];
app.DigitizerPanel_2.FontSize = 16;
app.DigitizerPanel_2.Position = [8 196 244 481];


% Create TBDLabel
app.TBDLabel = uilabel(app.DigitizerPanel_2);
app.TBDLabel.HorizontalAlignment = 'right';
app.TBDLabel.FontSize = 14;
app.TBDLabel.Position = [12 44 52 22];
app.TBDLabel.Text = 'TBD';


% Create Re_arm
app.Re_arm = uieditfield(app.DigitizerPanel_2, 'numeric');
```

```matlab
app.Re_arm.HorizontalAlignment = 'center';

app.Re_arm.FontSize = 14;

app.Re_arm.Position = [105 44 95 22];

app.Re_arm.Value = 5e-06;


% Create SamplinterLabel

app.SamplinterLabel = uilabel(app.DigitizerPanel_2);

app.SamplinterLabel.HorizontalAlignment = 'right';

app.SamplinterLabel.FontSize = 14;

app.SamplinterLabel.Position = [17 416 83 22];

app.SamplinterLabel.Text = 'Sampl. inter.';


% Create Saml_Interv

app.Saml_Interv = uieditfield(app.DigitizerPanel_2, 'numeric');

app.Saml_Interv.HorizontalAlignment = 'center';

app.Saml_Interv.FontSize = 14;

app.Saml_Interv.Position = [110 416 95 22];

app.Saml_Interv.Value = 1.6e-08;


% Create TBDLabel_2

app.TBDLabel_2 = uilabel(app.DigitizerPanel_2);

app.TBDLabel_2.HorizontalAlignment = 'right';

app.TBDLabel_2.FontSize = 14;

app.TBDLabel_2.Position = [59 204 33 22];

app.TBDLabel_2.Text = 'TBD';


% Create NumChan

app.NumChan = uieditfield(app.DigitizerPanel_2, 'numeric');

app.NumChan.HorizontalAlignment = 'center';

app.NumChan.FontSize = 14;
```

```matlab
app.NumChan.Position = [103 204 95 22];

app.NumChan.Value = 2;


% Create TBDLabel_3

app.TBDLabel_3 = uilabel(app.DigitizerPanel_2);

app.TBDLabel_3.HorizontalAlignment = 'right';

app.TBDLabel_3.FontSize = 14;

app.TBDLabel_3.Position = [36 10 33 22];

app.TBDLabel_3.Text = 'TBD';


% Create Timeout

app.Timeout = uieditfield(app.DigitizerPanel_2, 'numeric');

app.Timeout.HorizontalAlignment = 'center';

app.Timeout.FontSize = 14;

app.Timeout.Position = [105 10 95 22];

app.Timeout.Value = 9990;


% Create NumsamplesLabel

app.NumsamplesLabel = uilabel(app.DigitizerPanel_2);

app.NumsamplesLabel.HorizontalAlignment = 'right';

app.NumsamplesLabel.FontSize = 14;

app.NumsamplesLabel.Position = [14 353 95 22];

app.NumsamplesLabel.Text = 'Num. samples';


% Create NumSamples

app.NumSamples = uieditfield(app.DigitizerPanel_2, 'numeric');

app.NumSamples.ValueDisplayFormat = '%11.0d';

app.NumSamples.HorizontalAlignment = 'center';

app.NumSamples.FontSize = 14;

app.NumSamples.Position = [111 353 95 22];
```

```
app.NumSamples.Value = 13056;


% Create NumsegmentsLabel
app.NumsegmentsLabel = uilabel(app.DigitizerPanel_2);
app.NumsegmentsLabel.HorizontalAlignment = 'right';
app.NumsegmentsLabel.FontSize = 14;
app.NumsegmentsLabel.Position = [7 322 104 22];
app.NumsegmentsLabel.Text = 'Num. segments';


% Create NumSegments
app.NumSegments = uieditfield(app.DigitizerPanel_2, 'numeric');
app.NumSegments.HorizontalAlignment = 'center';
app.NumSegments.FontSize = 14;
app.NumSegments.Position = [112 322 95 22];
app.NumSegments.Value = 1;


% Create NumaveragesLabel
app.NumaveragesLabel = uilabel(app.DigitizerPanel_2);
app.NumaveragesLabel.HorizontalAlignment = 'right';
app.NumaveragesLabel.FontSize = 14;
app.NumaveragesLabel.Position = [7 292 100 22];
app.NumaveragesLabel.Text = 'Num. averages';


% Create NumAverages
app.NumAverages = uieditfield(app.DigitizerPanel_2, 'numeric');
app.NumAverages.HorizontalAlignment = 'center';
app.NumAverages.FontSize = 14;
app.NumAverages.Position = [112 292 95 22];
app.NumAverages.Value = 100;
```

```matlab
% Create TBDLabel_4
app.TBDLabel_4 = uilabel(app.DigitizerPanel_2);
app.TBDLabel_4.HorizontalAlignment = 'right';
app.TBDLabel_4.FontSize = 14;
app.TBDLabel_4.Position = [62 173 33 22];
app.TBDLabel_4.Text = 'TBD';


% Create Mode
app.Mode = uieditfield(app.DigitizerPanel_2, 'numeric');
app.Mode.HorizontalAlignment = 'center';
app.Mode.FontSize = 14;
app.Mode.Position = [105 173 95 22];
app.Mode.Value = 2;


% Create VerticalVoltageLabel
app.VerticalVoltageLabel = uilabel(app.DigitizerPanel_2);
app.VerticalVoltageLabel.HorizontalAlignment = 'right';
app.VerticalVoltageLabel.FontSize = 14;
app.VerticalVoltageLabel.Position = [7 384 98 22];
app.VerticalVoltageLabel.Text = 'VerticalVoltage';


% Create VertVoltage
app.VertVoltage = uieditfield(app.DigitizerPanel_2, 'numeric');
app.VertVoltage.HorizontalAlignment = 'center';
app.VertVoltage.FontSize = 14;
app.VertVoltage.Position = [112 384 95 22];
app.VertVoltage.Value = 5;


% Create TBDLabel_5
app.TBDLabel_5 = uilabel(app.DigitizerPanel_2);
```

```matlab
app.TBDLabel_5.HorizontalAlignment = 'right';

app.TBDLabel_5.FontSize = 14;

app.TBDLabel_5.Position = [64 137 33 22];

app.TBDLabel_5.Text = 'TBD';


% Create Flags
app.Flags = uieditfield(app.DigitizerPanel_2, 'numeric');

app.Flags.HorizontalAlignment = 'center';

app.Flags.FontSize = 14;

app.Flags.Position = [107 137 95 22];

app.Flags.Value = 1;


% Create TBDLabel_6
app.TBDLabel_6 = uilabel(app.DigitizerPanel_2);

app.TBDLabel_6.HorizontalAlignment = 'right';

app.TBDLabel_6.FontSize = 14;

app.TBDLabel_6.Position = [49 107 52 22];

app.TBDLabel_6.Text = 'TBD';


% Create xxxx
app.xxxx = uieditfield(app.DigitizerPanel_2, 'numeric');

app.xxxx.HorizontalAlignment = 'center';

app.xxxx.FontSize = 14;

app.xxxx.Position = [107 107 95 22];


% Create TBDLabel_7
app.TBDLabel_7 = uilabel(app.DigitizerPanel_2);

app.TBDLabel_7.HorizontalAlignment = 'right';

app.TBDLabel_7.FontSize = 14;

app.TBDLabel_7.Position = [68 76 33 22];
```

```
app.TBDLabel_7.Text = 'TBD';


% Create yyyy
app.yyyy = uieditfield(app.DigitizerPanel_2, 'numeric');
app.yyyy.HorizontalAlignment = 'center';
app.yyyy.FontSize = 14;
app.yyyy.Position = [108 76 95 22];


% Create TextArea_3
app.TextArea_3 = uitextarea(app.DigitizerPanel);
app.TextArea_3.FontSize = 10;
app.TextArea_3.Position = [65 31 137 154];
app.TextArea_3.Value = {'   Digitizer Defaults: '; '   rearm: 5.0000e-06'; '
sampl_interval: 1.6000e-08'; '    number_of_ch: 2'; '    timeout: 9990'; '
nbrSamples: 26624'; '    nbrSegments: 1'; '    nbrAvg: 1000'; '    mode: 2'; '
channelVert: [2 2]'; '    flags: 1'; ''};


% Create DCMagneticFieldGPanel
app.DCMagneticFieldGPanel = uipanel(app.SettingsTab);
app.DCMagneticFieldGPanel.ForegroundColor = [1 1 0.0667];
app.DCMagneticFieldGPanel.TitlePosition = 'centertop';
app.DCMagneticFieldGPanel.Title = 'DC Magnetic Field [G]';
app.DCMagneticFieldGPanel.BackgroundColor = [0.502 0.502 0.502];
app.DCMagneticFieldGPanel.FontSize = 16;
app.DCMagneticFieldGPanel.Position = [676 755 220 94];


% Create CoilCalibrationVGEditFieldLabel
app.CoilCalibrationVGEditFieldLabel = uilabel(app.DCMagneticFieldGPanel);
app.CoilCalibrationVGEditFieldLabel.HorizontalAlignment = 'right';
app.CoilCalibrationVGEditFieldLabel.FontSize = 14;
```

```matlab
        app.CoilCalibrationVGEditFieldLabel.FontColor = [1 1 0];

        app.CoilCalibrationVGEditFieldLabel.Position = [-1 41 136 22];

        app.CoilCalibrationVGEditFieldLabel.Text = 'Coil Calibration [V/G]';


        % Create DC_Coil_Calibration_V

        app.DC_Coil_Calibration_V = uieditfield(app.DCMagneticFieldGPanel,
'numeric');

        app.DC_Coil_Calibration_V.FontSize = 14;

        app.DC_Coil_Calibration_V.FontWeight = 'bold';

        app.DC_Coil_Calibration_V.Position = [142 41 63 22];

        app.DC_Coil_Calibration_V.Value = 0.0951;


        % Create ATTVoltageabslimitVLabel

        app.ATTVoltageabslimitVLabel = uilabel(app.DCMagneticFieldGPanel);

        app.ATTVoltageabslimitVLabel.HorizontalAlignment = 'right';

        app.ATTVoltageabslimitVLabel.FontSize = 14;

        app.ATTVoltageabslimitVLabel.FontColor = [1 1 0];

        app.ATTVoltageabslimitVLabel.Position = [11 8 169 22];

        app.ATTVoltageabslimitVLabel.Text = 'ATT Voltage abs limit [V]   ';


        % Create DC_Max_Volt_V

        app.DC_Max_Volt_V = uieditfield(app.DCMagneticFieldGPanel, 'numeric');

        app.DC_Max_Volt_V.FontWeight = 'bold';

        app.DC_Max_Volt_V.Position = [175 8 30 22];

        app.DC_Max_Volt_V.Value = 9.95;


        % Create HalfPeriodEditFieldLabel

        app.HalfPeriodEditFieldLabel = uilabel(app.SettingsTab);

        app.HalfPeriodEditFieldLabel.HorizontalAlignment = 'right';

        app.HalfPeriodEditFieldLabel.Position = [260 832 65 22];
```

```matlab
        app.HalfPeriodEditFieldLabel.Text = 'Half Period';


        % Create HalfPeriodEditField

        app.HalfPeriodEditField = uieditfield(app.SettingsTab, 'numeric');

        app.HalfPeriodEditField.ValueChangedFcn = createCallbackFcn(app,

@HalfPeriodEditFieldValueChanged, true);

        app.HalfPeriodEditField.Position = [340 832 100 22];

        app.HalfPeriodEditField.Value = 95300;


        % Create SpectraaveragingEditFieldLabel

        app.SpectraaveragingEditFieldLabel = uilabel(app.SettingsTab);

        app.SpectraaveragingEditFieldLabel.HorizontalAlignment = 'right';

        app.SpectraaveragingEditFieldLabel.Position = [270 803 103 22];

        app.SpectraaveragingEditFieldLabel.Text = 'Spectra averaging';


        % Create SpectraaveragingEditField

        app.SpectraaveragingEditField = uieditfield(app.SettingsTab, 'numeric');

        app.SpectraaveragingEditField.Limits = [0 Inf];

        app.SpectraaveragingEditField.RoundFractionalValues = 'on';

        app.SpectraaveragingEditField.Position = [388 803 100 22];

        app.SpectraaveragingEditField.Value = 1;


        % Create GxcalibrationEditFieldLabel

        app.GxcalibrationEditFieldLabel = uilabel(app.SettingsTab);

        app.GxcalibrationEditFieldLabel.HorizontalAlignment = 'right';

        app.GxcalibrationEditFieldLabel.Position = [270 771 79 22];

        app.GxcalibrationEditFieldLabel.Text = 'Gx calibration';


        % Create GxcalibrationEditField

        app.GxcalibrationEditField = uieditfield(app.SettingsTab, 'numeric');
```

```matlab
app.GxcalibrationEditField.Position = [364 771 100 22];

app.GxcalibrationEditField.Value = 0.3378;


% Create GycalibrationEditFieldLabel

app.GycalibrationEditFieldLabel = uilabel(app.SettingsTab);

app.GycalibrationEditFieldLabel.HorizontalAlignment = 'right';

app.GycalibrationEditFieldLabel.Position = [270 739 79 22];

app.GycalibrationEditFieldLabel.Text = 'Gy calibration';


% Create GycalibrationEditField

app.GycalibrationEditField = uieditfield(app.SettingsTab, 'numeric');

app.GycalibrationEditField.Position = [364 739 100 22];

app.GycalibrationEditField.Value = 0.375;


% Create GzcalibrationEditFieldLabel

app.GzcalibrationEditFieldLabel = uilabel(app.SettingsTab);

app.GzcalibrationEditFieldLabel.HorizontalAlignment = 'right';

app.GzcalibrationEditFieldLabel.Position = [270 711 79 22];

app.GzcalibrationEditFieldLabel.Text = 'Gz calibration';


% Create GzcalibrationEditField

app.GzcalibrationEditField = uieditfield(app.SettingsTab, 'numeric');

app.GzcalibrationEditField.Position = [364 711 100 22];

app.GzcalibrationEditField.Value = 0.319;


% Create EPRsamplesEditFieldLabel

app.EPRsamplesEditFieldLabel = uilabel(app.SettingsTab);

app.EPRsamplesEditFieldLabel.HorizontalAlignment = 'right';

app.EPRsamplesEditFieldLabel.Position = [273 615 78 22];

app.EPRsamplesEditFieldLabel.Text = 'EPR samples';
```

```matlab
% Create EPRsamplesEditField
app.EPRsamplesEditField = uieditfield(app.SettingsTab, 'numeric');
app.EPRsamplesEditField.Position = [366 615 100 22];
app.EPRsamplesEditField.Value = 10000;


% Create BGRemovalPanel
app.BGRemovalPanel = uipanel(app.SettingsTab);
app.BGRemovalPanel.Title = 'BG Removal';
app.BGRemovalPanel.BackgroundColor = [0.8 0.8 0.8];
app.BGRemovalPanel.FontSize = 16;
app.BGRemovalPanel.Position = [271 471 144 134];


% Create BGR_percent
app.BGR_percent = uieditfield(app.BGRemovalPanel, 'numeric');
app.BGR_percent.FontSize = 16;
app.BGR_percent.Position = [2 82 44 22];
app.BGR_percent.Value = 15;


% Create OrderLabel
app.OrderLabel = uilabel(app.BGRemovalPanel);
app.OrderLabel.Position = [50 82 36 22];
app.OrderLabel.Text = 'Order';


% Create BGR_order
app.BGR_order = uidropdown(app.BGRemovalPanel);
app.BGR_order.Items = {'0', '1', '2', '3', '4'};
app.BGR_order.FontSize = 16;
app.BGR_order.Position = [91 81 40 25];
app.BGR_order.Value = '1';
```

```matlab
% Create BGR_Harm
app.BGR_Harm = uidropdown(app.BGRemovalPanel);
app.BGR_Harm.Items = {'0', '1', '2', '3', '4'};
app.BGR_Harm.FontSize = 16;
app.BGR_Harm.Position = [7 47 40 25];
app.BGR_Harm.Value = '1';


% Create BGR_Harm_2
app.BGR_Harm_2 = uidropdown(app.BGRemovalPanel);
app.BGR_Harm_2.Items = {'0', '1', '2', '3', '4'};
app.BGR_Harm_2.FontSize = 16;
app.BGR_Harm_2.Position = [89 47 40 25];
app.BGR_Harm_2.Value = '1';


% Create HarmLabel
app.HarmLabel = uilabel(app.BGRemovalPanel);
app.HarmLabel.Position = [54 48 35 22];
app.HarmLabel.Text = 'Harm';


% Create HPFCorrectCheckBox
app.HPFCorrectCheckBox = uicheckbox(app.BGRemovalPanel);
app.HPFCorrectCheckBox.Text = ' HPF Correct';
app.HPFCorrectCheckBox.FontSize = 18;
app.HPFCorrectCheckBox.Position = [9 16 127 22];


% Create ButtonGroup_3
app.ButtonGroup_3 = uibuttongroup(app.SettingsTab);
app.ButtonGroup_3.BackgroundColor = [0.902 0.902 0.902];
app.ButtonGroup_3.FontSize = 18;
```

```matlab
app.ButtonGroup_3.Position = [274 388 152 71];


% Create Data_but

app.Data_but = uiradiobutton(app.ButtonGroup_3);

app.Data_but.Text = 'Data';

app.Data_but.FontSize = 16;

app.Data_but.Position = [6 44 60 22];


% Create FT_but

app.FT_but = uiradiobutton(app.ButtonGroup_3);

app.FT_but.Text = 'FT';

app.FT_but.FontSize = 16;

app.FT_but.Position = [68 16 37 22];


% Create Spec_but

app.Spec_but = uiradiobutton(app.ButtonGroup_3);

app.Spec_but.Text = 'Spec';

app.Spec_but.FontSize = 16;

app.Spec_but.Position = [6 14 60 27];

app.Spec_but.Value = true;


% Create Quad_but

app.Quad_but = uiradiobutton(app.ButtonGroup_3);

app.Quad_but.Text = 'Quad';

app.Quad_but.FontSize = 16;

app.Quad_but.Position = [63 44 61 22];


% Create Panel_7

app.Panel_7 = uipanel(app.SettingsTab);

app.Panel_7.BackgroundColor = [1 1 1];
```

```
app.Panel_7.Position = [271 244 167 129];


% Create GmaxEditFieldLabel

app.GmaxEditFieldLabel = uilabel(app.Panel_7);

app.GmaxEditFieldLabel.HorizontalAlignment = 'right';

app.GmaxEditFieldLabel.FontSize = 16;

app.GmaxEditFieldLabel.Position = [6 97 48 22];

app.GmaxEditFieldLabel.Text = 'Gmax';


% Create GmaxEditField

app.GmaxEditField = uieditfield(app.Panel_7, 'numeric');

app.GmaxEditField.FontSize = 16;

app.GmaxEditField.Position = [63 97 31 22];


% Create StartScanEditFieldLabel

app.StartScanEditFieldLabel = uilabel(app.Panel_7);

app.StartScanEditFieldLabel.HorizontalAlignment = 'right';

app.StartScanEditFieldLabel.FontSize = 16;

app.StartScanEditFieldLabel.Position = [4 36 80 22];

app.StartScanEditFieldLabel.Text = 'Start Scan';


% Create StartScanEditField

app.StartScanEditField = uieditfield(app.Panel_7, 'numeric');

app.StartScanEditField.FontSize = 16;

app.StartScanEditField.Position = [103 36 35 22];


% Create MaxScanEditFieldLabel

app.MaxScanEditFieldLabel = uilabel(app.Panel_7);

app.MaxScanEditFieldLabel.HorizontalAlignment = 'right';

app.MaxScanEditFieldLabel.FontSize = 16;
```

```matlab
app.MaxScanEditFieldLabel.Position = [4 8 77 22];

app.MaxScanEditFieldLabel.Text = 'Max Scan';


% Create MaxScanEditField

app.MaxScanEditField = uieditfield(app.Panel_7, 'numeric');

app.MaxScanEditField.FontSize = 16;

app.MaxScanEditField.Position = [87 8 48 22];


% Create FindButton

app.FindButton = uibutton(app.Panel_7, 'push');

app.FindButton.BackgroundColor = [0 1 0];

app.FindButton.Position = [104 97 44 22];

app.FindButton.Text = 'Find';


% Create FOVEditFieldLabel

app.FOVEditFieldLabel = uilabel(app.Panel_7);

app.FOVEditFieldLabel.HorizontalAlignment = 'right';

app.FOVEditFieldLabel.FontSize = 16;

app.FOVEditFieldLabel.Position = [6 68 38 22];

app.FOVEditFieldLabel.Text = 'FOV';


% Create FOVEditField

app.FOVEditField = uieditfield(app.Panel_7, 'numeric');

app.FOVEditField.FontSize = 16;

app.FOVEditField.Position = [50 68 48 22];

app.FOVEditField.Value = 1.5;


% Create ShimminggradientEditFieldLabel

app.ShimminggradientEditFieldLabel = uilabel(app.SettingsTab);

app.ShimminggradientEditFieldLabel.HorizontalAlignment = 'right';
```

```matlab
app.ShimminggradientEditFieldLabel.Position = [273 681 106 22];

app.ShimminggradientEditFieldLabel.Text = 'Shimming gradient';


% Create ShimminggradientEditField

app.ShimminggradientEditField = uieditfield(app.SettingsTab, 'numeric');

app.ShimminggradientEditField.Position = [394 681 100 22];

app.ShimminggradientEditField.Value = 0.1;


% Create Pico_calibration

app.Pico_calibration = uieditfield(app.UIFigure, 'numeric');

app.Pico_calibration.FontSize = 18;

app.Pico_calibration.Position = [197 20 100 23];

app.Pico_calibration.Value = 0.265;


% Create PicoScopeGppLabel

app.PicoScopeGppLabel = uilabel(app.UIFigure);

app.PicoScopeGppLabel.FontSize = 20;

app.PicoScopeGppLabel.Position = [14 19 179 24];

app.PicoScopeGppLabel.Text = 'PicoScope <> Gpp ';


% Create CorrEditFieldLabel

app.CorrEditFieldLabel = uilabel(app.UIFigure);

app.CorrEditFieldLabel.HorizontalAlignment = 'right';

app.CorrEditFieldLabel.Position = [343 20 29 22];

app.CorrEditFieldLabel.Text = 'Corr';


% Create CorrEditField

app.CorrEditField = uieditfield(app.UIFigure, 'numeric');

app.CorrEditField.Position = [387 20 100 22];

app.CorrEditField.Value = 1;
```

```
% Create BG_ReNew_Button

app.BG_ReNew_Button = uibutton(app.UIFigure, 'push');

app.BG_ReNew_Button.ButtonPushedFcn = createCallbackFcn(app,
@Collect_Background, true);

app.BG_ReNew_Button.FontSize = 20;

app.BG_ReNew_Button.Position = [516 16 59 31];

app.BG_ReNew_Button.Text = '- BG';


% Create BG_Zero_Button

app.BG_Zero_Button = uibutton(app.UIFigure, 'push');

app.BG_Zero_Button.ButtonPushedFcn = createCallbackFcn(app, @Zero_BG,
true);

app.BG_Zero_Button.FontSize = 20;

app.BG_Zero_Button.Position = [581 16 59 31];

app.BG_Zero_Button.Text = '0';


% Create AutoTuningCheckBox

app.AutoTuningCheckBox = uicheckbox(app.UIFigure);

app.AutoTuningCheckBox.Text = 'Auto Tuning';

app.AutoTuningCheckBox.FontSize = 20;

app.AutoTuningCheckBox.FontColor = [1 1 0.0667];

app.AutoTuningCheckBox.Position = [674 19 129 23];

app.AutoTuningCheckBox.Value = true;


% Create AutocenterOffButton

app.AutocenterOffButton = uibutton(app.UIFigure, 'push');

app.AutocenterOffButton.ButtonPushedFcn = createCallbackFcn(app,
@AutocenterOffButtonPushed, true);

app.AutocenterOffButton.BackgroundColor = [1 0 0];
```

```matlab
        app.AutocenterOffButton.FontSize = 18;

        app.AutocenterOffButton.Position = [820 17 137 29];

        app.AutocenterOffButton.Text = 'Auto-center Off';


        % Create ButtonGroup_4

        app.ButtonGroup_4 = uibuttongroup(app.UIFigure);

        app.ButtonGroup_4.SelectionChangedFcn = createCallbackFcn(app, @User_,

true);

        app.ButtonGroup_4.Position = [1044 16 215 30];


        % Create MButton

        app.MButton = uiradiobutton(app.ButtonGroup_4);

        app.MButton.Text = 'M';

        app.MButton.Position = [11 3 58 22];

        app.MButton.Value = true;


        % Create RButton

        app.RButton = uiradiobutton(app.ButtonGroup_4);

        app.RButton.Text = 'R';

        app.RButton.Position = [46 3 28 22];


        % Create SButton

        app.SButton = uiradiobutton(app.ButtonGroup_4);

        app.SButton.Text = 'S';

        app.SButton.Position = [82 2 40 22];


        % Create OButton

        app.OButton = uiradiobutton(app.ButtonGroup_4);

        app.OButton.Text = 'O';

        app.OButton.Position = [116 3 40 22];
```

```matlab
            % Create DButton
            app.DButton = uiradiobutton(app.ButtonGroup_4);
            app.DButton.Text = 'D';
            app.DButton.Position = [153 3 40 22];


            % Show the figure after all components are created
            app.UIFigure.Visible = 'on';
        end
    end


    % App creation and deletion
    methods (Access = public)


        % Construct app
        function app = August_EPRI_2023


            % Create UIFigure and components
            createComponents(app)


            % Register the app with App Designer
            registerApp(app, app.UIFigure)


            % Execute the startup function
            runStartupFcn(app, @startupFcn)


            if nargout == 0
                clear app
            end
        end
```

```matlab
        % Code that executes before app deletion
        function delete(app)


            % Delete UIFigure when app is deleted
            delete(app.UIFigure)
        end
    end
end
```

## 6.2    ADiC Arduino program

```cpp
#include <SPI.h> //spi library
#include <sc5511a.h> //digital oscillator library


sc5511a sc5511a(30, 29);


int led = 13;
int blinks = 0, k , m;
int read_points_num; // number of points to read from A10
byte analog_read_buffer[310];
byte Matlab_data[10];
byte freqarray[5]; // array of 5 bytes to be sent to sc5511
byte mode, range, RF_A,  RF_B, DTC1 , DTC2, DTC1_i, DTC2_i; // vars to read from MATLAB
int RFplus,RFminus,current_CW,plus_CW,minus_CW; //pars to select +- CW frequency
int RF; // CW frequnecy
int rf2_frequency = 1500; // 1600/2 = 800 MHz => IF frequnecy
const byte DTC_pin_1 = 28, DTC_pin_2 = 32, switch_pin = 25, trigger_AWG_pin = 41,
reflection_pin = A10,digitizer_trig = 20;
SPISettings dtcSettings(912500, MSBFIRST, SPI_MODE0); //spi settings
```

```
// New from version 5

byte patt1[94] = {0, 0, 1, 1, 1, 2, 2, 2, 3, 3, 3, 4, 4, 4, 5, 5, 5, 6, 6, 6, 7, 7, 7,
8, 8, 8, 9, 9, 9, 10, 10, 10, 11, 11, 11, 12, 12, 12, 13, 13, 13, 14, 14, 14, 15, 15, 15,
16, 16, 16, 17, 17, 17, 18, 18, 18, 19, 19, 19, 20, 20, 20, 21, 21, 21, 22, 22, 22, 23,
23, 23, 24, 24, 24, 25, 25, 25, 26, 26, 26, 27, 27, 27, 28, 28, 28, 29, 29, 29, 30, 30,
30, 31, 31};

byte patt2[94] = {0, 1, 0, 1, 2, 1, 2, 3, 2, 3, 4, 3, 4, 5, 4, 5, 6, 5, 6, 7, 6, 7, 8, 7,
8, 9, 8, 9, 10, 9, 10, 11, 10, 11, 12, 11, 12, 13, 12, 13, 14, 13, 14, 15, 14, 15, 16,
15, 16, 17, 16, 17, 18, 17, 18, 19, 18, 19, 20, 19, 20, 21, 20, 21, 22, 21, 22, 23, 22,
23, 24, 23, 24, 25, 24, 25, 26, 25, 26, 27, 26, 27, 28, 27, 28, 29, 28, 29, 30, 29, 30,
31, 30, 31};

byte inx_total = 94, search_order[95];

byte Current_reflection, Current_position, Best_Reflection, Best_Position, Best_segment,
R_threshold = 130; // R_threshold  : reflection treshold

byte inx_current = 3; // current index in the search pattern of 94 values

boolean test_search;


void setup() {
  // put your setup code here, to run once:
  Serial.begin(9600);


  pinMode(DTC_pin_1, OUTPUT); // DTC #1
  pinMode(DTC_pin_2, OUTPUT); // DTC #2
  pinMode(switch_pin, OUTPUT);
  pinMode(trigger_AWG_pin, OUTPUT);
  pinMode(led, OUTPUT);
  pinMode(digitizer_trig,OUTPUT);
  digitalWrite(digitizer_trig,LOW);
  pinMode(reflection_pin, INPUT);
```

```
analogReadResolution(8);

analogReadAveraging(1);


sc5511a.INITIALIZE(1);

delayMicroseconds(750);

sc5511a.RF_MODE_CW();

delayMicroseconds(750);

sc5511a.RF_LEVEL(1300);

delayMicroseconds(750);

sc5511a.RF_OUT_ENABLE(1);

delayMicroseconds(750);

sc5511a.LED_ON();

delay(1000);

sc5511a.LED_OFF();

delay(1000);

sc5511a.LED_ON();

delayMicroseconds(750);

sc5511a.RF_LEVEL(1300);


sc5511a.RF2_FREQUENCY(rf2_frequency);

delay(1);

sc5511a.RF2_STANDBY(0);

delay(1);

sc5511a.RF2_FREQUENCY(rf2_frequency);

delay(1);

sc5511a.RF2_STANDBY(0);

delay(1);


SPI1.begin(); //begin spi for dtc on separate spi line from digital oscillator
}
```

```
void loop() { }


void serialEvent() {
  Tune();
}


void Tune () {
  //void serialEvent() {


  int k = 0;
  while (Serial.available() > 0) {
    Matlab_data[k] = byte( Serial.read() );
    k++;
  }
  if (k>=2){
  mode =  Matlab_data[0];
  range = Matlab_data[1];
  RF_A =  Matlab_data[2];
  RF_B =  Matlab_data[3];
  DTC1_i =  Matlab_data[4];
  DTC2_i =  Matlab_data[5];
  R_threshold = Matlab_data[6];
  Serial.write(0);
  }
  else{
    Serial.write(255);
    return;
  }
```

```
RF = int( ( float(RF_A) * 256 +  float(RF_B) ) * 1e3 + range * 10e6); // [Hz]

// RF * 10e3 + 790e6 => RF freq. in  MHz

// RF = RF * 1e4 + 790e6; // in [Hz]


//  mode = 2;

//  RF = 800e6;


//mode = 1;

switch (mode) {


  case 0:       // Write back to MATLAB to test delivery

    //  blink_test();


    for (k = 0; k < 6; k++) {

      Serial.write(Matlab_data[k]);

    }

    break;



  case 1:       // Set DTCs and receive reflecton >> MATLAB

    //  blink_test();

    //  noInterrupts();

    DTC_write(byte( DTC1_i ), DTC_pin_1);

    DTC_write(byte( DTC2_i ), DTC_pin_2);

    digitalWrite(switch_pin, 0); // 0 : tuning mode & 1 : CW mode

    read_points_num = 301;

    tuning_trigger(); // Sends trigger to AWG to start freq. sweep

    readanalog(read_points_num); //  read A10 & returns "byte analog_read_buffer"

    // interrupts();
```

```
  for (k = 0; k < read_points_num; k++) {

    Serial.write(analog_read_buffer[k]);

    // Serial.write(byte(k));

  }



  make_sc5511a_CW_array(RF); // outputs freq arrray

  setdfs();


  break;



case 2: // smart search

  digitalWrite(switch_pin, 0); // 0 = tuning mode &&   1 =  CW mode

  inx_current = Best_segment;

  make_search_order(); // outputs search_order[94]

  Best_Reflection  = 0;

  Best_Position = 0;

  // R_threshold = 135;

  read_points_num = 301;


  //  Serial.println(" New cycle >>> ");

  for (k = 1; k <= 94; k++) {


    DTC1_i = patt1[search_order[k]];  DTC_write(byte( DTC1_i ), DTC_pin_1);

    DTC2_i = patt2[search_order[k]];  DTC_write(byte( DTC2_i ), DTC_pin_2);

    tuning_trigger(); // Sends trigger to AWG to start freq. sweep

    readanalog(read_points_num); //  read A10 & returns "byte analog_read_buffer"

    max_in_array(); // outputs Current_reflection, Current_position
```

```
      // delay(200);

      if (Current_reflection > Best_Reflection ) {

        Best_Reflection = Current_reflection;

        Best_Position =  Current_position;

        Best_segment =  search_order[k];

        DTC1 = DTC1_i;

        DTC2 = DTC2_i;

      } // end if



      //        Serial.print(k);

      //        Serial.print(" ");

      //        Serial.print(DTC1_i);

      //        Serial.print(" ");

      //        Serial.print(DTC2_i);

      //        Serial.print(" ");

      //        Serial.print(Best_Reflection);

      //        Serial.print(" ");

      //        Serial.println(Current_reflection);

      blinks = k;

      if (Best_Reflection > R_threshold ) {

        break;

      }


//  digitalWrite(switch_pin, 0); // 0 : tuning mode & 1 : CW mode


    } // end for


    DTC_write(byte( DTC1 ), DTC_pin_1);

    DTC_write(byte( DTC2 ), DTC_pin_2);
```

```
// to test

test_search = 1;

if (test_search) {

  read_points_num = 301;

  tuning_trigger(); // Sends trigger to AWG to start freq. sweep

  readanalog(read_points_num); //  read A10 & returns "byte analog_read_buffer"


  // analog_read_buffer[0] = 10; // test

  analog_read_buffer[301] = Best_Reflection;

  analog_read_buffer[302] = Best_Position;

  analog_read_buffer[303] = DTC1;

  analog_read_buffer[304] = DTC2;

  analog_read_buffer[305] = Best_segment;

  analog_read_buffer[306] = blinks;


  read_points_num = 307;

  for (m = 0; m < read_points_num; m++) {

    Serial.write(analog_read_buffer[m]);

    // Serial.write(byte(k));

  }

}

RF = find_RF (Best_Position);


make_sc5511a_CW_array (RF); // outputs freq arrray

setdfs();


// read reflction

for (byte i;i<10;i++){

  current_CW = analogRead(reflection_pin);

}
```

```
  // + 35 k

  RFplus = RF + 35e3;

  make_sc5511a_CW_array(RFplus);

  setdfs();

  for (byte i;i<10;i++){

    plus_CW = analogRead(reflection_pin);

  }


  // - 35 k

  RFminus = RF - 35e3;

  make_sc5511a_CW_array(RFminus);

  setdfs();

  for(byte i; i<10;i++){

    minus_CW = analogRead(reflection_pin);

  }
  // find best
  if(plus_CW>=current_CW){

    if (plus_CW >= minus_CW){

      RF = RFplus;

    }else{

      RF = RFminus;

    }

  }else if(minus_CW>current_CW){

    RF = RFminus;

  }

  make_sc5511a_CW_array(RF);

  setdfs();

  break;
case 3: //smart tune no data back

  digitalWrite(switch_pin, 0); // 0 = tuning mode &&   1 =  CW mode
```

```
digitalWrite(digitizer_trig,HIGH);

inx_current = Best_segment;

make_search_order(); // outputs search_order[94]

Best_Reflection  = 0;

Best_Position = 0;

R_threshold = 135;

read_points_num = 301;

for (k = 1; k <= 94; k++) {

  DTC1_i = patt1[search_order[k]];  DTC_write(byte( DTC1_i ), DTC_pin_1);

  DTC2_i = patt2[search_order[k]];  DTC_write(byte( DTC2_i ), DTC_pin_2);

  tuning_trigger(); // Sends trigger to AWG to start freq. sweep

  readanalog(read_points_num); //  read A10 & returns "byte analog_read_buffer"

  max_in_array(); // outputs Current_reflection, Current_position

  if (Current_reflection > Best_Reflection ) {

    Best_Reflection = Current_reflection;

    Best_Position =  Current_position;

    Best_segment =  search_order[k];

    DTC1 = DTC1_i;

    DTC2 = DTC2_i;

  } // end if

  if (Best_Reflection > R_threshold ) {

    break;

  }

} // end for

DTC_write(byte( DTC1 ), DTC_pin_1);

DTC_write(byte( DTC2 ), DTC_pin_2);

RF = find_RF (Best_Position);

make_sc5511a_CW_array (RF); // outputs freq arrray

setdfs();
```

```
// read reflction
for (byte i;i<10;i++){
  current_CW = analogRead(reflection_pin);
}
// + 35 k
RFplus = RF + 35e3;
make_sc5511a_CW_array(RFplus);
setdfs();
for (byte i;i<10;i++){
  plus_CW = analogRead(reflection_pin);
}


// - 35 k
RFminus = RF - 35e3;
make_sc5511a_CW_array(RFminus);
setdfs();
for(byte i; i<10;i++){
  minus_CW = analogRead(reflection_pin);
}
// find best
if(plus_CW>=current_CW){
  if (plus_CW >= minus_CW){
    RF = RFplus;
  }else{
    RF = RFminus;
  }
}else if(minus_CW>current_CW){
  RF = RFminus;
}
make_sc5511a_CW_array(RF);
```

```
        setdfs();

        digitalWrite(digitizer_trig,LOW);

        break;

  } // end switch

digitalWrite(switch_pin, 1); // 0 : tuning mode & 1 : CW mode




} // end Event


// ******* Functions ************************ >>


void DTC_write(byte x, int capacitor_pin) {

  digitalWrite(capacitor_pin, HIGH);


  SPI1.beginTransaction(dtcSettings);

  SPI1.transfer(byte(x));

  SPI1.endTransaction();

  digitalWrite(capacitor_pin, LOW);

  delayMicroseconds(100);

}


void tuning_trigger() {

  digitalWrite(trigger_AWG_pin, 0); //

  digitalWrite(trigger_AWG_pin, 1);

  digitalWrite(trigger_AWG_pin, 0); //

  delayMicroseconds(5);

}


void readanalog(int Npoints) {

  for (int p = 0; p < Npoints; p++) {
```

```
      analog_read_buffer[p] = analogRead(reflection_pin);

  }

}


void blink_test() {

  int b;

  for (b = 0; b < 2; b++) {

    digitalWrite(led, HIGH);   // turn the LED on (HIGH is the voltage level)

    delay(220);                // wait for a second

    digitalWrite(led, LOW);    // turn the LED off by making the voltage LOW

    delay(220);                // wait for a second

  }


}


void make_sc5511a_CW_array (int CW_in_Hz) {

  byte s;

  for (s = 1; s < 6; s++) {

    freqarray[5 - s] = floor((CW_in_Hz % int(pow(256, float(s)))) /

    pow(256, float((s - 1))));

    //Serial.println(freqarray[5-k]); //for arduino monitor

  }

}


void setdfs() {

  sc5511a.RF_OUT_ENABLE(1);

  delay(5);

  sc5511a.RF_FREQUENCY_BY_BYTE(freqarray[0], freqarray[1], freqarray[2], freqarray[3],

  freqarray[4]);

  delay(5);
```

```
  sc5511a.RF_LEVEL(1300);

  delay(5);


  //usingenable();

  digitalWrite(switch_pin, 1);// 2 CW

}


void make_search_order() {

  byte R, L, N;

  int n;

  byte dtc;


  dtc = inx_current;

  N = dtc + 1;   // i = 0   => N = 1

  R =  94 - N ; // inx_total = 94

  L = N - 1;

  search_order[1] = dtc ; // order starts from index  1 to be as in MATLAB


  if (R > L) {


    //

    for (n = 1; n <= L; n++) {

      search_order[2 * n ] = N - n - 1;

      search_order[2 * n + 1] = N + n - 1;

    }

    //

    for (n = N + L + 1; n <= 94 ; n++) { // inx_total = 94

      search_order[n] = n - 1;

    }

    //
```

```
    } // if


    else {

      //

      for (n = 1; n <= R; n++) {

        search_order[2 * n] = N - n - 1;

        search_order[2 * n + 1 ] = N + n - 1;

      }

      //

      for (n = (2 * R + 2 ); n <= 94; n++) { // inx_total = 94

        search_order[n ] = 94 - n;

      }

      //


    } // end else


}// end function



void max_in_array () {

  float a, b, R , max_ = 0;

  int n;

  byte  pos = 0;

  byte sh = 10; //shift to exclude bad data ??? why : not sure yet

  for (n = sh; n < read_points_num - sh; n++) {


    a = float( analog_read_buffer[n - 1] ) ;

    b = float( analog_read_buffer[n + 1] ) ;

    R = 0.2 * (a + b) + 0.6 * float(analog_read_buffer[n]);
```

```
  if (R > max_) {

    max_ = R;

    pos = n;

  }

  }

  Current_reflection = int(max_);

  Current_position = pos;

}




int find_RF (int position_) {

  float  CF, offset_ = 11922, slope = 14.901;

  int result;

  CF = int( 1e6 * (position_ + offset_) / slope); // RF frequnecy in [Hz]

  result = CF  + (range-80) * 10e6;

  return result;



}

//        for (m = 0; m < read_points_num; m++) {

//          Serial.println(analog_read_buffer[m]);

//          }
```

## 6.3   Arduino digital frequency source sc5511a library h file

```
/*

  sc5511a.h library for the sc5511a SPI controlled oscillator.

  Created by Ryan Curtis O'Connell, August 25th, 2021.

  Released into the public domain.

*/
```

```cpp
#ifndef sc5511a_h
#define sc5511a_h


#include "Arduino.h"
#include "SPI.h"


class sc5511a
{
  public:
    sc5511a(int cs_B, int Serial_ready);
    void INITIALIZE(int initialize_option);
    void LED_ON();
    void LED_OFF();
    void SYNTH_MODE(int synth_mode_select);
    void RF_MODE_CW();
    void RF_MODE_SWEEP();
    void LIST_MODE_CONFIG(int list_mode_config_select);
    void LIST_START_FREQ(unsigned long start_frequency_1, unsigned long
     start_frequency_2);
    void LIST_STOP_FREQ(unsigned long stop_frequency_1,unsigned long stop_frequency_2);
    void LIST_STEP_FREQ(unsigned long step_frequency_1, unsigned long step_frequency_2);
    void LIST_DWELL_TIME(unsigned long dwell_time);
    void LIST_CYCLE_COUNT(unsigned long cycle_count);
    void LIST_BUFFER_POINTS(int buffer_points);
    void LIST_BUFFER_WRITE(unsigned long buffer_write_1, unsigned long buffer_write_2);
    void LIST_BUF_MEM_XFER(int xfer_select);
    void LIST_SOFT_TRIGGER();
    void RF_FREQUENCY(unsigned long frequency_1, unsigned long frequency_2);
    void RF_FREQUENCY_BY_BYTE(byte rf_byte_4, byte rf_byte_3,
```

```
      byte rf_byte_2, byte rf_byte_1, byte rf_byte_0);

    void RF_LEVEL(word level);

    void RF_OUT_ENABLE(byte enable_select);

    void RF_PHASE(int phase_select);

    void AUTO_LEVEL_DISABLE(int level_disable_select);

    void RF_ALC_MODE(int alc_mode_select);

    void RF_STANDBY(int rf_standby_select);

    void REFERENCE_MODE(int reference_mode_select);

    void REFERENCE_DAC_SETTING(word dac_setting);

    void ALC_DAC_VALUE(word dac_value);

    void STORE_DEFAULT_STATE();

    void RF2_STANDBY(int rf2_standby_select);

    void RF2_FREQUENCY(word rf2_frequency_select);

    void SYNTH_SELF_CAL();
  private:
    int _cs_B;

    int _Serial_ready;

    byte _byte1;

    byte _byte2;

    byte _byte3;

    byte _byte4;

    byte _byte5;

    int _bit_hold;
};


#endif
```

## 6.4    Arduino digital frequency source sc5511a library cpp file

```
/* sc5511a.cpp library for the sc5511a SPI controlled oscillator.
```

Created by Ryan Curtis O'Connell, August 25th, 2021.

```
*/


#include "Arduino.h"

#include "sc5511a.h"

#include "SPI.h"

SPISettings _mySettings(2000000,MSBFIRST,SPI_MODE0);


sc5511a::sc5511a(int cs_B, int Serial_ready)

{

  pinMode(cs_B,OUTPUT);

  pinMode(Serial_ready,INPUT);

  _cs_B = cs_B;

  _Serial_ready = Serial_ready;


}


/*

  this function sets up SPI and tells SC5511a to run through initialization.

  initialize_option: 0 = load saved settings, 1 = initialize only

*/

void sc5511a::INITIALIZE(int initialize_option)

{

  SPI.begin();

  digitalWrite(_cs_B,1);

  _byte1 = 0;

  _byte2 = 0;

  _byte3 = 0;

  _byte4 = 0;

  _byte5 = 0;
```

```cpp
  SPI.beginTransaction(_mySettings);

  digitalWrite(_cs_B,0);

  delayMicroseconds(1);

  SPI.transfer(0x01);

  delayMicroseconds(1);

  SPI.transfer(initialize_option);

  digitalWrite(_cs_B,1);

  SPI.endTransaction();

}


//this function turns on LED on front panel
void sc5511a::LED_ON()
{
  SPI.beginTransaction(_mySettings);

  digitalWrite(_cs_B,0);

  delayMicroseconds(1);

  SPI.transfer(0x02);

  delayMicroseconds(1);

  SPI.transfer(1);

  digitalWrite(_cs_B,1);

  SPI.endTransaction();

}


//this function turns off LED on front panel
void sc5511a::LED_OFF()
{
  SPI.beginTransaction(_mySettings);

  digitalWrite(_cs_B,0);

  delayMicroseconds(1);
```

```
  SPI.transfer(0x02);

  delayMicroseconds(1);

  SPI.transfer(0);

  digitalWrite(_cs_B,1);

  SPI.endTransaction();

}

/*bit 0: 0- harmonic offset mode

  1- fracN PLL offset mode

  bit 1: 0- normal loop gain for better close in phase noise

  1- low loop gain for better far out phase noise and spur suppression

  bit 2: only applies in harmonic offset mode

  0- automatically switches to fracN offset mode to avoid potentially

  large spurs due to intermodulation between the carrier and the

  harmonics of the reference clock.

  1- this disables the function, may speed up tuning speed

*/

void sc5511a::SYNTH_MODE(int synth_mode_select)

{

  SPI.beginTransaction(_mySettings);

  digitalWrite(_cs_B,0);

  delayMicroseconds(1);

  SPI.transfer(0x03);

  delayMicroseconds(1);

  SPI.transfer(synth_mode_select);

  digitalWrite(_cs_B,1);

  SPI.endTransaction();

}


//this function switches rf mode to cw

void sc5511a::RF_MODE_CW()
```

```
{
  SPI.beginTransaction(_mySettings);

  digitalWrite(_cs_B,0);

  delayMicroseconds(1);

  SPI.transfer(0x04);

  delayMicroseconds(1);

  SPI.transfer(0);

  digitalWrite(_cs_B,1);

  SPI.endTransaction();

}


//this function switches rf mode to sweep

void sc5511a::RF_MODE_SWEEP()

{
  SPI.beginTransaction(_mySettings);

  digitalWrite(_cs_B,0);

  delayMicroseconds(1);

  SPI.transfer(0x04);

  delayMicroseconds(1);

  SPI.transfer(1);

  digitalWrite(_cs_B,1);

  SPI.endTransaction();

}


/*

  bit 0: 0- list mode. 1-sweep mode.

  bit 1: 0-forward sweep 1-reverse sweep.

  bit 2: 0-sawtooth waveform 1-triangular waveform

  bit 3: 0-software trigger 1-hardware trigger

  bit 4: 0-start/stop behavior 1-step-on-trigger
```

```
  bit 5: 0-stop at end of sweep/list

  1- return to start.

  bit 6: 0- no trigger out 1- trigger out

  bit 7: 0- trigger out at each frequency change

  1- trigger out at end of each sweep/list cycle
*/
void sc5511a::LIST_MODE_CONFIG(int list_mode_config_select)

{

  SPI.beginTransaction(_mySettings);

  digitalWrite(_cs_B,0);

  delayMicroseconds(1);

  SPI.transfer(0x05);

  delayMicroseconds(1);

  SPI.transfer(list_mode_config_select);

  digitalWrite(_cs_B,1);

  SPI.endTransaction();

}


//has to be lower than the stop frequency

void sc5511a::LIST_START_FREQ(unsigned long start_frequency_1,

unsigned long start_frequency_2)

{

  for (int i=0; i<8; i++)

  {

    _bit_hold = bitRead(start_frequency_1,(i));

    bitWrite(_byte1,i,_bit_hold);


    _bit_hold = bitRead(start_frequency_2,(i));

    bitWrite(_byte5,i,_bit_hold);
```

```
  _bit_hold = bitRead(start_frequency_1,(i+8));

  bitWrite(_byte2,i,_bit_hold);


  _bit_hold = bitRead(start_frequency_1,(i+16));

  bitWrite(_byte3,i,_bit_hold);


  _bit_hold = bitRead(start_frequency_1,(i+24));

  bitWrite(_byte4,i,_bit_hold);

}


SPI.beginTransaction(_mySettings);

digitalWrite(_cs_B,0);

delayMicroseconds(1);

SPI.transfer(0x06);

delayMicroseconds(1);

SPI.transfer(_byte5);

delayMicroseconds(1);

SPI.transfer(_byte4);

delayMicroseconds(1);

SPI.transfer(_byte3);

delayMicroseconds(1);

SPI.transfer(_byte2);

delayMicroseconds(1);

SPI.transfer(_byte1);

delayMicroseconds(1);

digitalWrite(_cs_B,1);

SPI.endTransaction();


_byte1 = 0;

_byte2 = 0;
```

```cpp
  _byte3 = 0;

  _byte4 = 0;

  _byte5 = 0;

}


//should be higher than start frequency

void sc5511a::LIST_STOP_FREQ(unsigned long stop_frequency_1,

unsigned long stop_frequency_2){

  for (int i=0; i<8; i++)

  {

    _bit_hold = bitRead(stop_frequency_1,(i));

    bitWrite(_byte1,i,_bit_hold);


    _bit_hold = bitRead(stop_frequency_2,(i));

    bitWrite(_byte5,i,_bit_hold);


    _bit_hold = bitRead(stop_frequency_1,(i+8));

    bitWrite(_byte2,i,_bit_hold);


    _bit_hold = bitRead(stop_frequency_1,(i+16));

    bitWrite(_byte3,i,_bit_hold);


    _bit_hold = bitRead(stop_frequency_1,(i+24));

    bitWrite(_byte4,i,_bit_hold);

  }


  SPI.beginTransaction(_mySettings);

  digitalWrite(_cs_B,0);

  delayMicroseconds(1);

  SPI.transfer(0x07);
```

```cpp
  delayMicroseconds(1);

  SPI.transfer(_byte5);

  delayMicroseconds(1);

  SPI.transfer(_byte4);

  delayMicroseconds(1);

  SPI.transfer(_byte3);

  delayMicroseconds(1);

  SPI.transfer(_byte2);

  delayMicroseconds(1);

  SPI.transfer(_byte1);

  delayMicroseconds(1);

  digitalWrite(_cs_B,1);

  SPI.endTransaction();


  _byte1 = 0;

  _byte2 = 0;

  _byte3 = 0;

  _byte4 = 0;

  _byte5 = 0;

}


//do not exceed difference between start and stop frequencies

void sc5511a::LIST_STEP_FREQ(unsigned long step_frequency_1, unsigned long

step_frequency_2)

{

  for (int i=0; i<8; i++)

  {

    _bit_hold = bitRead(step_frequency_1,(i));

    bitWrite(_byte1,i,_bit_hold);
```

```
  _bit_hold = bitRead(step_frequency_2,(i));

  bitWrite(_byte5,i,_bit_hold);


  _bit_hold = bitRead(step_frequency_1,(i+8));

  bitWrite(_byte2,i,_bit_hold);


  _bit_hold = bitRead(step_frequency_1,(i+16));

  bitWrite(_byte3,i,_bit_hold);


  _bit_hold = bitRead(step_frequency_1,(i+24));

  bitWrite(_byte4,i,_bit_hold);

}


SPI.beginTransaction(_mySettings);

digitalWrite(_cs_B,0);

delayMicroseconds(1);

SPI.transfer(0x08);

delayMicroseconds(1);

SPI.transfer(_byte5);

delayMicroseconds(1);

SPI.transfer(_byte4);

delayMicroseconds(1);

SPI.transfer(_byte3);

delayMicroseconds(1);

SPI.transfer(_byte2);

delayMicroseconds(1);

SPI.transfer(_byte1);

delayMicroseconds(1);

digitalWrite(_cs_B,1);

SPI.endTransaction();
```

```cpp
  _byte1 = 0;

  _byte2 = 0;

  _byte3 = 0;

  _byte4 = 0;

  _byte5 = 0;

}

/*

  Set the dwell time at each step frequency. The dwell time is

  incremented in 500us increments. for example, to produce a 10ms

  dwell time the value written to this register is 20d.

*/

void sc5511a::LIST_DWELL_TIME(unsigned long dwell_time)

{

  for (int i=0; i<8; i++)

  {

    _bit_hold = bitRead(dwell_time,(i));

    bitWrite(_byte1,i,_bit_hold);


    _bit_hold = bitRead(dwell_time,(i+8));

    bitWrite(_byte2,i,_bit_hold);


    _bit_hold = bitRead(dwell_time,(i+16));

    bitWrite(_byte3,i,_bit_hold);


    _bit_hold = bitRead(dwell_time,(i+24));

    bitWrite(_byte4,i,_bit_hold);

  }


  SPI.beginTransaction(_mySettings);
```

```cpp
  digitalWrite(_cs_B,0);

  delayMicroseconds(1);

  SPI.transfer(0x09);

  delayMicroseconds(1);

  SPI.transfer(_byte4);

  delayMicroseconds(1);

  SPI.transfer(_byte3);

  delayMicroseconds(1);

  SPI.transfer(_byte2);

  delayMicroseconds(1);

  SPI.transfer(_byte1);

  delayMicroseconds(1);

  digitalWrite(_cs_B,1);

  SPI.endTransaction();


  _byte1 = 0;

  _byte2 = 0;

  _byte3 = 0;

  _byte4 = 0;

  _byte5 = 0;

}


/*

  0= cycle forever. not 0 will set the number of cycles the

  device will sweep or step through the list then stop.

  This applies for both start/stop and step trigger modes.

*/

void sc5511a::LIST_CYCLE_COUNT(unsigned long cycle_count)

{

  for (int i=0; i<8; i++)
```

```
{
    _bit_hold = bitRead(cycle_count,(i));
    bitWrite(_byte1,i,_bit_hold);


    _bit_hold = bitRead(cycle_count,(i+8));
    bitWrite(_byte2,i,_bit_hold);


    _bit_hold = bitRead(cycle_count,(i+16));
    bitWrite(_byte3,i,_bit_hold);


    _bit_hold = bitRead(cycle_count,(i+24));
    bitWrite(_byte4,i,_bit_hold);
}


SPI.beginTransaction(_mySettings);
digitalWrite(_cs_B,0);
delayMicroseconds(1);
SPI.transfer(0x0A);
delayMicroseconds(1);
SPI.transfer(_byte4);
delayMicroseconds(1);
SPI.transfer(_byte3);
delayMicroseconds(1);
SPI.transfer(_byte2);
delayMicroseconds(1);
SPI.transfer(_byte1);
delayMicroseconds(1);
digitalWrite(_cs_B,1);
SPI.endTransaction();
```

```
    _byte1 = 0;

    _byte2 = 0;

    _byte3 = 0;

    _byte4 = 0;

    _byte5 = 0;

}


/*

  sets the number of frequency points to step through

  in the buffer list. the number must be equal to or less

  than the buffer length. This number will overwrite the

  count determined from the list_buffer_write register.

*/

void sc5511a::LIST_BUFFER_POINTS(int buffer_points)

{

  for (int i=0; i<8; i++)

  {

    _bit_hold = bitRead(buffer_points,(i));

    bitWrite(_byte1,i,_bit_hold);


    _bit_hold = bitRead(buffer_points,(i+8));

    bitWrite(_byte2,i,_bit_hold);


    _bit_hold = bitRead(buffer_points,(i+16));

    bitWrite(_byte3,i,_bit_hold);


    _bit_hold = bitRead(buffer_points,(i+24));

    bitWrite(_byte4,i,_bit_hold);

  }
```

```
SPI.beginTransaction(_mySettings);

digitalWrite(_cs_B,0);

delayMicroseconds(1);

SPI.transfer(0x0C);

delayMicroseconds(1);

SPI.transfer(_byte4);

delayMicroseconds(1);

SPI.transfer(_byte3);

delayMicroseconds(1);

SPI.transfer(_byte2);

delayMicroseconds(1);

SPI.transfer(_byte1);

delayMicroseconds(1);

digitalWrite(_cs_B,1);

SPI.endTransaction();


  _byte1 = 0;

  _byte2 = 0;

  _byte3 = 0;

  _byte4 = 0;

  _byte5 = 0;

}


/*

  Writing this register stores the frequency point into the list

  buffer held in RAM. Writing 0x0000000000 to this buffer resets

  the pointer to buffer location 0 and flags the device to store

  data written to this register. Consecutive non-zero writes

  to this register will increase the buffer counter up to 2047.

  further writes beyond this point are not recognized. Writing
```

```
      0xFFFFFFFFFF to this register at any time will terminate

      the write process and stops the pointer increment.

      The value at which the pointer stops is the new count

      of list frequency points unless it is overwritten by

      register list_buffer_points.

*/

void sc5511a::LIST_BUFFER_WRITE(unsigned long buffer_write_1, unsigned long

buffer_write_2)

{

  for (int i=0; i<8; i++)

  {

    _bit_hold = bitRead(buffer_write_1,(i));

    bitWrite(_byte1,i,_bit_hold);


    _bit_hold = bitRead(buffer_write_2,(i));

    bitWrite(_byte5,i,_bit_hold);


    _bit_hold = bitRead(buffer_write_1,(i+8));

    bitWrite(_byte2,i,_bit_hold);


    _bit_hold = bitRead(buffer_write_1,(i+16));

    bitWrite(_byte3,i,_bit_hold);


    _bit_hold = bitRead(buffer_write_1,(i+24));

    bitWrite(_byte4,i,_bit_hold);

  }


  SPI.beginTransaction(_mySettings);

  digitalWrite(_cs_B,0);

  delayMicroseconds(1);
```

```
  SPI.transfer(0x0D);

  delayMicroseconds(1);

  SPI.transfer(_byte5);

  delayMicroseconds(1);

  SPI.transfer(_byte4);

  delayMicroseconds(1);

  SPI.transfer(_byte3);

  delayMicroseconds(1);

  SPI.transfer(_byte2);

  delayMicroseconds(1);

  SPI.transfer(_byte1);

  delayMicroseconds(1);

  digitalWrite(_cs_B,1);

  SPI.endTransaction();


  _byte1 = 0;

  _byte2 = 0;

  _byte3 = 0;

  _byte4 = 0;

  _byte5 = 0;

}


/*bit 0: 0- transfers the content of the list buffer into EEPROM

  1- transfers the contents from EEPROM into list buffer

*/

void sc5511a::LIST_BUF_MEM_XFER(int xfer_select)

{

  SPI.beginTransaction(_mySettings);

  digitalWrite(_cs_B,0);

  delayMicroseconds(1);
```

```cpp
    SPI.transfer(0x0E);

    delayMicroseconds(1);

    SPI.transfer(xfer_select);

    digitalWrite(_cs_B,1);

    SPI.endTransaction();

}


//calling this register provides a soft trigger to device

void sc5511a::LIST_SOFT_TRIGGER()

{

    SPI.beginTransaction(_mySettings);

    digitalWrite(_cs_B,0);

    delayMicroseconds(1);

    SPI.transfer(0x0F);

    delayMicroseconds(1);

    SPI.transfer(0);

    digitalWrite(_cs_B,1);

    SPI.endTransaction();

}


/*

  this function sets the RF frequency of channel 1 in Hz

  example: rf frequency = 800MHz RF_FREQUENCY(800000000,0)

  example: rf frequency = 20GHz RF_FREQUENCY(4000000000,16)

*/

void sc5511a::RF_FREQUENCY(unsigned long frequency_1,unsigned long frequency_2)

{

  for (int i=0; i<8; i++)

  {

    _bit_hold = bitRead(frequency_1,(i));
```

```
  bitWrite(_byte1,i,_bit_hold);


  _bit_hold = bitRead(frequency_2,(i));
  bitWrite(_byte5,i,_bit_hold);


  _bit_hold = bitRead(frequency_1,(i+8));
  bitWrite(_byte2,i,_bit_hold);


  _bit_hold = bitRead(frequency_1,(i+16));
  bitWrite(_byte3,i,_bit_hold);


  _bit_hold = bitRead(frequency_1,(i+24));
  bitWrite(_byte4,i,_bit_hold);
}


SPI.beginTransaction(_mySettings);
digitalWrite(_cs_B,0);
delayMicroseconds(1);
SPI.transfer(0x10);
delayMicroseconds(1);
SPI.transfer(_byte5);
delayMicroseconds(1);
SPI.transfer(_byte4);
delayMicroseconds(1);
SPI.transfer(_byte3);
delayMicroseconds(1);
SPI.transfer(_byte2);
delayMicroseconds(1);
SPI.transfer(_byte1);
delayMicroseconds(1);
```

```
  digitalWrite(_cs_B,1);

  SPI.endTransaction();


  _byte1 = 0;

  _byte2 = 0;

  _byte3 = 0;

  _byte4 = 0;

  _byte5 = 0;

}


void sc5511a::RF_FREQUENCY_BY_BYTE(byte rf_byte_1, byte rf_byte_2,

byte rf_byte_3, byte rf_byte_4, byte rf_byte_5)

{

  SPI.beginTransaction(_mySettings);

  digitalWrite(_cs_B,0);

  delayMicroseconds(3);

  SPI.transfer(0x10);

  delayMicroseconds(3);

  SPI.transfer(rf_byte_1);

  delayMicroseconds(3);

  SPI.transfer(rf_byte_2);

  delayMicroseconds(3);

  SPI.transfer(rf_byte_3);

  delayMicroseconds(3);

  SPI.transfer(rf_byte_4);

  delayMicroseconds(3);

  SPI.transfer(rf_byte_5);

  delayMicroseconds(3);

  digitalWrite(_cs_B,1);

  SPI.endTransaction();
```

```
  delayMicroseconds(500);

}


/* sets the RF power level in hundredths (13db = 1300).

  to set negative power level bitWrite() the first bit of

  the word to 1.

*/

void sc5511a::RF_LEVEL(word level)

{

  SPI.beginTransaction(_mySettings);

  digitalWrite(_cs_B,0);

  delayMicroseconds(3);

  SPI.transfer(0x11);

  delayMicroseconds(3);

  SPI.transfer(highByte(level));

  delayMicroseconds(3);

  SPI.transfer(lowByte(level));

  delayMicroseconds(3);

  digitalWrite(_cs_B,1);

  SPI.endTransaction();

  delayMicroseconds(500);

}


//enable/disable RF channel 1

void sc5511a::RF_OUT_ENABLE(byte enable_select)

{

  SPI.beginTransaction(_mySettings);

  digitalWrite(_cs_B,0);

  delayMicroseconds(3);

  SPI.transfer(0x12);
```

```
  delayMicroseconds(3);

  SPI.transfer(byte(enable_select));

  delayMicroseconds(3);

  digitalWrite(_cs_B,1);

  SPI.endTransaction();

  delayMicroseconds(500);

}

/*This function sets the phase of the oscillator.

  In tenths of a degree. Use first bit as negative

  phase.

*/

void sc5511a::RF_PHASE(int phase_select)

{

  SPI.beginTransaction(_mySettings);

  digitalWrite(_cs_B,0);

  delayMicroseconds(1);

  SPI.transfer(0x13);

  delayMicroseconds(1);

  SPI.transfer(phase_select);

  delayMicroseconds(1);

  digitalWrite(_cs_B,1);

  SPI.endTransaction();

}

/*0=power is leveled on frequency change

  1=power is not leveled on frequency change but by

    explicitly calling 0x11

*/

void sc5511a::AUTO_LEVEL_DISABLE(int level_disable_select)

{

  SPI.beginTransaction(_mySettings);
```

```cpp
  digitalWrite(_cs_B,0);

  delayMicroseconds(1);

  SPI.transfer(0x14);

  delayMicroseconds(1);

  SPI.transfer(level_disable_select);

  delayMicroseconds(1);

  digitalWrite(_cs_B,1);

  SPI.endTransaction();

}
/* 0=amplitude is corrected using ALC closed loop

   1=amplitude is corrected w/ open loop

*/
void sc5511a::RF_ALC_MODE(int alc_mode_select)

{

  SPI.beginTransaction(_mySettings);

  digitalWrite(_cs_B,0);

  delayMicroseconds(1);

  SPI.transfer(0x15);

  delayMicroseconds(1);

  SPI.transfer(alc_mode_select);

  delayMicroseconds(1);

  digitalWrite(_cs_B,1);

  SPI.endTransaction();

}


//sets channel 1 to standby (low power mode)

void sc5511a::RF_STANDBY(int rf_standby_select)

{

  SPI.beginTransaction(_mySettings);

  digitalWrite(_cs_B,0);
```

```cpp
  delayMicroseconds(1);

  SPI.transfer(0x16);

  delayMicroseconds(1);

  SPI.transfer(rf_standby_select);

  delayMicroseconds(1);

  digitalWrite(_cs_B,1);

  SPI.endTransaction();

}

/* first bit: 1=instructs the device to lock to external source

   second bit: 0=outputs 10MHz, 1=outputs 100MHz

*/

void sc5511a::REFERENCE_MODE(int reference_mode_select)

{

  SPI.beginTransaction(_mySettings);

  digitalWrite(_cs_B,0);

  delayMicroseconds(1);

  SPI.transfer(0x17);

  delayMicroseconds(1);

  SPI.transfer(reference_mode_select);

  delayMicroseconds(1);

  digitalWrite(_cs_B,1);

  SPI.endTransaction();

}

//14 bit word to set/adjust the internal 10MHz TCXO frequency

void sc5511a::REFERENCE_DAC_SETTING(word dac_setting)

{

  SPI.beginTransaction(_mySettings);

  digitalWrite(_cs_B,0);

  delayMicroseconds(1);

  SPI.transfer(0x18);
```

```
  delayMicroseconds(1);

  SPI.transfer(highByte(dac_setting));

  delayMicroseconds(1);

  SPI.transfer(lowByte(dac_setting));

  delayMicroseconds(1);

  digitalWrite(_cs_B,1);

  SPI.endTransaction();

}

/* 14 bit word to set/adjust the ALC DAC value. This is

  useful to make output adjustment. The current ALC DAC

  value can be read back via register 0x25.

*/

void sc5511a::ALC_DAC_VALUE(word dac_value)

{

  SPI.beginTransaction(_mySettings);

  digitalWrite(_cs_B,0);

  delayMicroseconds(1);

  SPI.transfer(0x19);

  delayMicroseconds(1);

  SPI.transfer(highByte(dac_value));

  delayMicroseconds(1);

  SPI.transfer(lowByte(dac_value));

  delayMicroseconds(1);

  digitalWrite(_cs_B,1);

  SPI.endTransaction();

}

/*calling this register will store the current configuration

  into memory. On reset or power-up these values are read

  from memory (intialize value 1) and set as the default

  values. Some of these values are: RF1 parameters,
```

```
  list mode configuration, RF mode, List mode parameters,

  RF2 parameters.

*/

void sc5511a::STORE_DEFAULT_STATE()

{

  SPI.beginTransaction(_mySettings);

  digitalWrite(_cs_B,0);

  delayMicroseconds(1);

  SPI.transfer(0x1B);

  delayMicroseconds(1);

  SPI.transfer(0);

  delayMicroseconds(1);

  digitalWrite(_cs_B,1);

  SPI.endTransaction();

}

//sets channel 2 to standby (low power mode). 1 standby 0 off

void sc5511a::RF2_STANDBY(int rf2_standby_select)

{

  SPI.beginTransaction(_mySettings);

  digitalWrite(_cs_B,0);

  delayMicroseconds(3);

  SPI.transfer(0x1E);

  delayMicroseconds(3);

  SPI.transfer(rf2_standby_select);

  delayMicroseconds(3);

  digitalWrite(_cs_B,1);

  SPI.endTransaction();

  delayMicroseconds(500);

}
```

```cpp
void sc5511a::RF2_FREQUENCY(word rf2_frequency_select)
{
  SPI.beginTransaction(_mySettings);
  digitalWrite(_cs_B,0);
  delayMicroseconds(3);
  SPI.transfer(0x1F);
  delayMicroseconds(3);
  SPI.transfer(highByte(rf2_frequency_select));
  delayMicroseconds(3);
  SPI.transfer(lowByte(rf2_frequency_select));
  delayMicroseconds(3);
  digitalWrite(_cs_B,1);
  SPI.endTransaction();
  delayMicroseconds(500);
}
//no description
void sc5511a::SYNTH_SELF_CAL()
{
  SPI.beginTransaction(_mySettings);
  digitalWrite(_cs_B,0);
  delayMicroseconds(1);
  SPI.transfer(0x47);
  delayMicroseconds(1);
  SPI.transfer(0);
  delayMicroseconds(1);
  digitalWrite(_cs_B,1);
  SPI.endTransaction();
}
```

## 6.5    Bioprint Supplemental Figures

| Trial | Cell concentration (cells/mL) |
|---|---|
| 1% Pyruvate trial 1 | 31,800,000 |
| 1% Pyruvate trial 2 | 29,200,000 |
| 1% Pyruvate trial 3 | 51,400,000 |
| 1% Pyruvate trial 4 | 33,600,000 |
| 0.4% Pyruvate trial 1 | 22,400,000 |
| 0.4% Pyruvate trial 2 | 42,800,000 |
| 0.4% Pyruvate trial 3 | 57,600,000 |
| 0.4% Pyruvate trial 4 | 22,000,000 |
| 0.2% Pyruvate trial 1 | 16,200,000 |
| 0.2% Pyruvate trial 2 | 54,000,000 |
| 0.2% Pyruvate trial 3 | 54,200,000 |
| 0.2% Pyruvate trial 4 | 49,000,000 |
| 0.04% Pyruvate trial 1 | 36,800,000 |
| 0.04% Pyruvate trial 2 | 54,000,000 |
| 0.04% Pyruvate trial 3 | 52,500,000 |
| 0.04% Pyruvate trial 4 | 56,100,000 |

Table 6.1: Cell concentrations (cells/mL) for stirring experiment (see figure 4.7). Cells were counted using a hemocytometer.
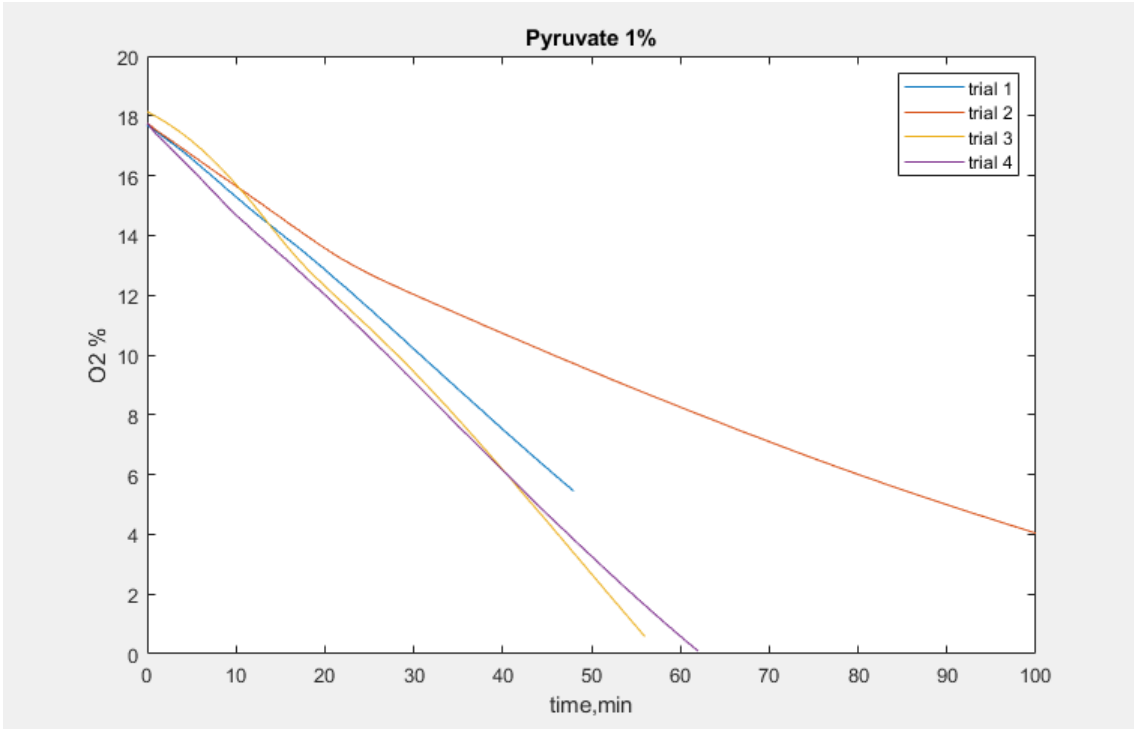
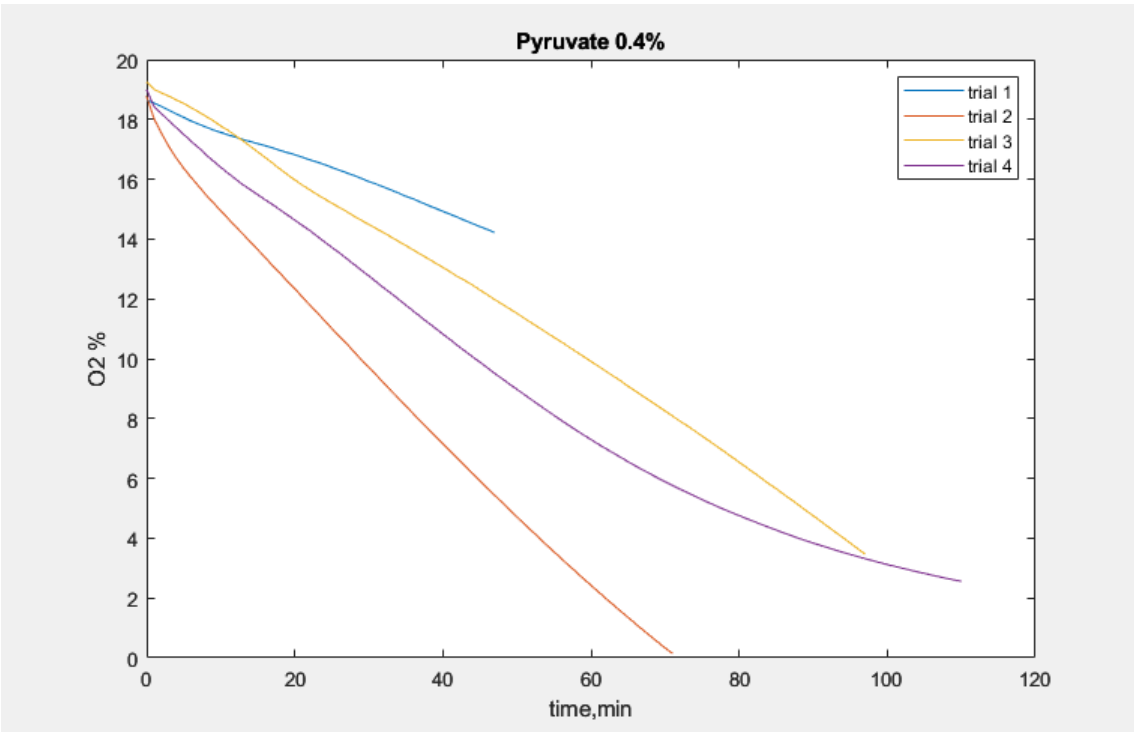Figure 6.1: Yeast oxygen measurement with 1% pyruvate



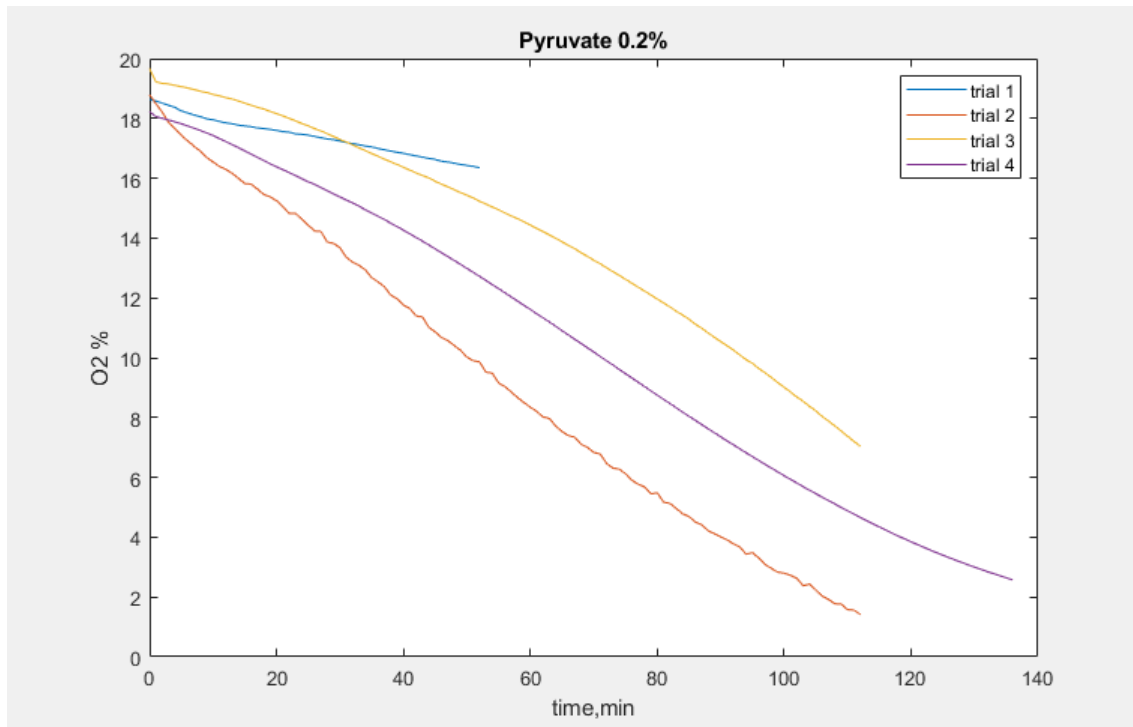Figure 6.2: Yeast oxygen measurement with 0.4% pyruvate

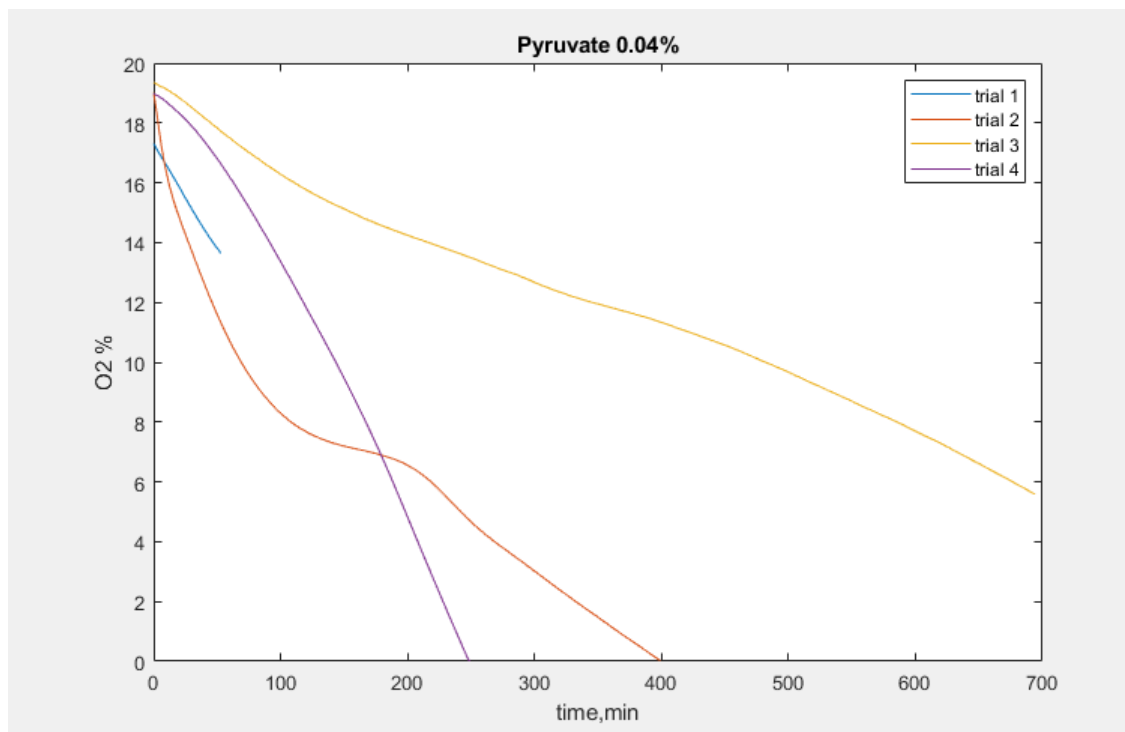Figure 6.3: Yeast oxygen measurement with 0.2% pyruvate



Figure 6.4: Yeast oxygen measurement with 0.04% pyruvate

## 6.6 Simulation code

```
clc;
clear all;


%%
NN = 400;


C = 1 *1e6;
F =       1;
Rates = [ 5,    0.5, 0.02] *1e-4;
[r, n, f] = meshgrid(Rates, NN, F);
r = r(:);
n = n(:);
f = f(:);



%%


for k = 1 : length(r)
    tic
    % N=10000; % > 1e6
    C = 1e6;
    [t, Yy] = Diffus(k,n(k), f(k), r(k));
    Res(k).Y = Yy;
    Res(k).t = t;
    Res(k).N = n(k);
    Res(k).C = C;
    Res(k).rate = r(k);
    toc
```

```
end

save 'Results_run_1'

%%



function [t, Yy] = Diffus(num,N, F, rate)


% constants

% C = 1e6; % cells per ml

C = 1e6;

K = 3*60;

r = 4; % cell diameter [um]

D=2e-5; %Diffusion

Y0 = 0.2e-3; %concentration O2 in water

rate0 =  rate; %


Cl = C^(1/3); % cells per cm, linear

Clu = Cl*1e-4; %cells per um

% C3u = Clu^3; %cells in um^3

d = 1/Clu; %inter-cell distamce in um

cm = 1e4;

D_um = D*cm^2;


%Yy = zeros(K,N);

x=linspace(r,d/2,N);

dx = x(2)-x(1); %layer thickness

dt = dx.^2/D_um;


M = round( 1/dt );
```

```matlab
Yy = zeros(K,N);

Y=x*0+Y0; %initial O2 distribution


Dx = x*0 + D_um;

Dx(1) = F*2.9e-6*cm^2;

Dx(2) = Dx(1);

Dx(3) = Dx(1);

%


x_ = x;

dx_ = gradient(x_);

t = linspace(0, M*K*dt, K);


for j= 1 : K
    tic
     for k= 1 : M
         Y1 = (x_.^2).*gradient(Y)./dx_;
         dY_ = gradient(Y1)./(x_.^2)./dx_;
         dY = dY_.*Dx ; %% > D = vector
         Y = Y + dY *dt;
         Y(1) = Y(2) - rate0 *dt; % inside cells
         if Y(1) < 0; Y(1) = 0; end
         Y(N) = Y(N-1);
     end
    toc


    Yy(j,:) = Y;
    %plot(Y);
 %   imagesc(x,t,Y);  colorbar
```

```matlab
    try

        fn = ['run_1' num2str(j)];

        save( fn );

        disp('saved')

        disp(num);

    catch

        disp(' ?')

        disp('saved ??')

    end

    pause(.1);

end



end
```

## 6.7   Gcode-generating MATLAB code

```matlab
clear all;

clc


fname = 'smallcylinder_Z20.gcode';

%% Parameters

% x  is recommend to make the longest


x0y0z0 = [100, 100, 20] ;  % starting coordinates

dz = 0.1;                  % inter-layer distances

Ex_rate = 0.01;            % Extrusion rate

speed = 800;               % Print Speed

dr = 0.25;                 % distance between spiral lines in xy plane

H = 8;                     % Verticle height
```

```matlab
R = 7.5;                        % outer radius
r = 3;                          % inner radius




%% 1
% x0y0z0 = [100, 100, 3] ;  % starting coordinates original z 1.5
% H = 8;
%r = 3.5;
% R = 7;
[Gcode1,time1,extr1] = Function_Make_cylinder (x0y0z0, H, R, r, dr, dz, Ex_rate, speed);


%% 2
%x0y0z0 = [112, 100, 1.5] ;  % starting coordinates
%H = 13;
%r = 1.5;
%R = 3.5;
%[Gcode2,time2,extr2] = Function_Make_cylinder (x0y0z0, H, R, r, dr, dz, Ex_rate, speed);


% %% 3
% x0y0z0 = [100, 110, 1] ;  % starting coordinates
% H = 14;
% r = 1.5;
% R = 3;
% [Gcode3,time3,extr3] = Function_Make_cylinder (x0y0z0, H, R, r, dr, dz, Ex_rate, speed);
%
% %% 4
% x0y0z0 = [90, 100, 1] ;  % starting coordinates
% H = 14;
% r = 1.5;
```

```matlab
% R = 3;
% [Gcode4,time4,extr4] = Function_Make_cylinder (x0y0z0, H, R, r, dr, dz, Ex_rate, speed);
%
% %% 5
% x0y0z0 = [100, 90, 1] ;  % starting coordinates
% H = 14;
% r = 1.5;
% R = 3;
% [Gcode5,time5,extr5] = Function_Make_cylinder (x0y0z0, H, R, r, dr, dz, Ex_rate, speed);


%% all together
Gcode = [Gcode1]; % Gcode3 Gcode4 Gcode5];
%Gcode = Gcode1;


writeGcode(Gcode,fname)
time = time1;% + time3 + time4 + time5;
%%fprintf('\nPart 1 Print Time Estimate(Seconds) is : %f',time1)
fprintf('\nTotal Print Time Estimate(Seconds) is : %f',time)
fprintf('\nPrint Time Estimate(Minutes) is : %.2f',time/60)
extr = extr1;% + extr2; % + extr3 + extr4 + extr5;
%%fprintf('\nPart 1 Bioink Usage Estimate(mL) is : %f',extr1)
fprintf('\nTotal Bioink Usage Estimate(mL) is : %f',extr)


function [Gcode,est_time,est_extr] = Function_Make_cylinder
(x0y0z0, H, R, r, dr, dz, Ex_rate, speed)
%% Pars
% x  is recommend to make the longest


% x0y0z0 = [100, 100, 1] ;  % starting coordinates
% dz = 0.1;                 % inter-layer distances
```

353

```matlab
% Ex_rate = 0.01;             % Extrusion rate

% dr = 0.25; % distance between spiral lines in xy plane

% dz = 0.1;

% H = 10;   % Verticle height


x0 = x0y0z0(1);

y0 = x0y0z0(2);

z0 = x0y0z0(3);



%%

%R = 8; % outer radius

%r = 3; % inner radius

d_ang = 5; % [degrees] in angle increments

%

z = 0 : dz : H; % verticle

N_rings = (R-r)/dr; % Number  of rings in xy plane

phi = 0: d_ang : 360 * N_rings;

ro = r +  phi/360*dr;

x = ro.* cos(phi/180*pi);

y = ro.* sin(phi/180*pi);

%plot(x,y)

%%

X =  [];

Y  = [];

Z =  [];


for k = 1 : length(z)

    s = (-1)^k; % sign
```

```matlab
    if s > 0

        X = [X x];
        Y = [Y y];
    else

        X = [X fliplr(x)];
        Y = [Y fliplr(y)];

    end

    Z = [Z y*0 + z(k)];
%     if k < 4
%         plot3(X,Y,Z);
%     end
end


whos X Y x y




%%
EXTR = 0;


est_time = 0;
est_extr = 0;


Gcode(1).tx = 'G92 E0' ;
cnt = 2;
```

```matlab
Gcode(cnt).tx =  G0_(x0 + X(2),y0 + Y(2),z0); % starting position

cnt = cnt + 1;


Gcode(cnt).tx =  G0_(-1,-1,z0); % starting position

cnt = cnt + 1;


N = length(X);


xi = x0;

yi = y0;

zi = z0;


for i = 2:N


    dx = x0 + X(i) - xi;

    dy = y0 + Y(i) - yi;

    dz = z0 + Z(i) - zi;


    xi = x0 + X(i);

    yi = y0 + Y(i);

    zi = z0 + Z(i);


    if dz == 0

        Ex = Ex_rate * sqrt(dx^2 + dy^2);

        EXTR = EXTR + Ex;

        Gcode(cnt).tx = G1_(xi,yi,-1,EXTR,speed);

        cnt = cnt + 1;

        est_time = est_time + sqrt(dx^2 + dy^2)/(speed/60);

    else

        Gcode(cnt).tx =  G1_(-1, -1 ,zi,-1,speed); % moving up  position
```

```matlab
        cnt = cnt + 1;

        est_time = est_time + dz*3/(speed/60);
    end
end


%%
clc
for k = 1 :  length(Gcode)
    disp(Gcode(k).tx);
end


est_extr = (EXTR/320)*5.5/1.32;
fprintf('Estimated print time is: %f',est_time)
fprintf('\nEstimated bioink needed is: %f',est_extr)


function writeGcode(Gcode,fname)


load Before_and_After Before After


%% Write header


fid = fopen(fname,'w');


for k = 1 :  length(Before)
    %disp(Before(k).tx);
    fprintf(fid,'%s\n',Before(k).tx);
end
```

```matlab
for k = 1 :  length(Gcode)

    %disp(Gcode(k).tx);

    fprintf(fid,'%s\n',Gcode(k).tx);

end


for k = 1 :  length(After)

    %disp(After(k).tx);

    fprintf(fid,'%s\n',After(k).tx);

end


fclose(fid);
```