

University of Strathclyde Glasgow

μLED arrays for optogenetic cortical surface stimulation

PhD Thesis

Ryan Greer

Neurophotonics group

Department of Physics

University of Strathclyde, Glasgow

October 6, 2026

This thesis is the result of the author's original research. It has been composed by the author and has not been previously submitted for examination which has led to the award of a degree.

The copyright of this thesis belongs to the author under the terms of the United Kingdom Copyright Acts as qualified by University of Strathclyde Regulation 3.50. Due acknowledgement must always be made of the use of any material contained in, or derived from, this thesis.

R. Greer
06/10/26

Abstract

Cortical implants are a translational neurotechnology, already transforming the lives of patients with debilitating neurodegenerative conditions. One application of these devices are as a cortical stimulator to restore lost sensation, and specifically there is the possibility to restore lost hearing through stimulation of the auditory cortex, motivating the work in this thesis. Here, optogenetic cortical implants have been developed based on custom-fabricated arrays of 100 μLEDs (micro-light emitting diodes). These devices can deliver spatiotemporal optical stimulation across the surface of the cortex. A packaging process has been developed integrating the devices into flexible systems, enabling chronic implantation and stable months-long experiments in the mouse model. Extensive laboratory characterisation and modelling has established an operational framework for the devices and validated them for long-term implantation. These devices have enabled *in-vivo* experiments with simultaneous optogenetic stimulation of the auditory cortex with electrophysiology recording, as well as behavioural experiments in freely-moving mice. These experiments have allowed a stimulation threshold and spatial resolution to be estimated for the devices, informing future experiments that integrate novel deep learning models for meaningful information transfer to the auditory cortex. Lastly, an electronic controller system was developed to drive the devices *in-vivo*, enabling experiments involving sophisticated spatio-temporal stimulation patterns and closed-loop control.

Contents

Acknowledgements **6**

1 Motivation **7**
1.1 Introduction . . . 7
1.2 Literature review . . . 8
1.3 Thesis summary . . . 9

2 Background and theory **14**
2.1 The brain, cerebral cortex and neurons . . . 14
2.2 Auditory cortex and the auditory pathway . . . 16
2.3 Cortical surface stimulation methodologies . . . 17
2.4 Optogenetics . . . 18
2.5 Light emitting diodes . . . 22
2.6 µLED array fabrication . . . 23
2.7 Device encapsulation strategies for long-term chronic implantation . . . 26
2.8 Closed-loop control . . . 27
2.9 Signal processing electronics . . . 27

3 µLED array systems for chronic *in-vivo* experiments **37**
3.1 µLED array operation and interconnection . . . 38
3.2 Integration and packaging of µLED arrays . . . 39
3.2.1 Rigid implants for acute experimentation . . . 39
3.2.2 Flexible implants for chronic implantation . . . 44
3.2.3 Wireless system prototype . . . 50
3.2.4 Scalable integration of multiple µLED arrays to flexible package . . . 52
3.3 Accelerated ageing for the evaluation of encapsulation strategies for long-term chronic implantation . . . 53
3.4 Integrated recording with flexible µECoG array . . . 59

4 Optical and thermal considerations for cortical surface implanted µLEDs 69
4.1 Optical modelling . . . 69
4.2 µLED array electrical and optical characteristics . . . 73
4.3 Design and analysis of an optical collimator for enhanced spatial resolution . 77
4.4 Comparison of Lambertian emitting µLEDs versus collimated digital light projector approach . . . 81
4.5 Thermal characterisation and modelling . . . 84
4.6 Discussion . . . 89

5 Device validation through *in-vivo* experimentation in mice 94
5.1 Acute *in-vivo* electrophysiology experiments . . . 95
5.1.1 Experimental setup . . . 96
5.1.2 Data analysis methods . . . 99
5.1.3 Stimulation threshold results . . . 100
5.1.4 Spatial resolution . . . 106
5.1.5 Multi-unit analysis . . . 109
5.2 Chronic behavioural experiments . . . 112
5.2.1 Behavioural experiments in head-fixed mice . . . 114
5.2.2 Freely-behaving experiments . . . 119
5.3 Histology . . . 122
5.4 Current and future directions . . . 124

6 Advanced systems for closed-loop cortical stimulation 128
6.1 Microcontroller-based current driver system for µLED cortical implant (v1) . . 128
6.2 FPGA-enabled system for closed-loop optogenetic control and integration of electrophysiology recording . . . 130
6.2.1 LED driver software . . . 133
6.2.2 Intan RHD interface FPGA design . . . 134
6.2.3 Closed-loop electrophysiology recording and optogenetic stimulation 140
6.3 Discussion . . . 141

7 Conclusions and future work 144
7.1 Discussion . . . 144
7.2 Future work . . . 146
7.2.1 Braincodec . . . 146
7.2.2 Novel *in-vivo* experiments . . . 148
7.2.3 Closed-loop electrophysiology . . . 148
7.2.4 Novel system geometries . . . 149

7.2.5 Multi-colour µLED arrays . . . 149
7.3 Final remark . . . 149

A Supplementary information **152**
A.1 µLED array wafer history and failure mechanisms . . . 152
A.2 Comparison of previous chronically implantable µLED/LED array cortical surface stimulators . . . 153
A.3 p values from statistical tests . . . 154
A.4 Behavioural experimental parameters . . . 155

B Hardware designs **157**
B.1 PCB schematics . . . 158
B.1.1 RHD2000 interface . . . 158
B.1.2 LED driver . . . 159
B.2 PCB board design . . . 161
B.2.1 Front layer . . . 161
B.2.2 Inner layer 1 . . . 162
B.2.3 Inner layer 2 . . . 163
B.2.4 Bottom layer . . . 164

C Code listings **165**
C.1 FPGA Verilog code listings . . . 166
C.2 Vivado IP Integrator block diagram - Rhythm AXI integrated with PYNQ base overlay . . . 179
C.3 µLED array driver code listings . . . 180
C.4 Control system Jupyter Notebook demos . . . 189

Acknowledgements

I would like to acknowledge the following people for making this exciting project possible and supporting me along the way.

Keith Mathieson for offering me the opportunity to work in the Neurophotonics group and for his guidance and support during the project and with my professional career. Paul Murray for his support as second supervisor and continued support of my professional career.

All the members of the group (Emma, Vanesa, Yunzhou, Svet and Niall) as well as the wider Strathclyde Neurotechnology Centre and Institute of Photonics. It has been a pleasure working with such a great group of people along the years.

Emma Butt, Yunzhou Cheng: fabrication of the μLED arrays and cleanroom expertise.

Niall McAlinden: laboratory expertise including training with lab equipment, experimental setups and modelling software.

Svetoslav Stoyanov: design of the wireless headstages used to demonstrate prototype wireless μLED array.

Shuzo Sakata: providing his expertise on histology.

The HearLight consortium; specifically the partnership with Antonin Verdier and Brice Bathellier who have extensively tested the devices in the laboratory and provided the data analysed in Chapter V.

Antonin Verdier: conducting electrophysiology experiments, surgeries and head-fixed behavioural experiments. Providing the data analysed in Chapter V.

Ella Callas: conducting head-fixed behavioural experiments.

Alicia Aniorte, Eya Mabrouk: preparing the freely-behaving experimental setup and conducting experiments.

Eya Mabrouk: analysing the data and producing the plots in figures 5.23-5.26, 5.28 and 7.1.

Magdalena Pereyra: carrying out histology and providing the images and data for figure 5.27.

Amélie Albon and Charles Rezaei-Mazinani for their expertise in fabricating the ECoG devices used in this project.

I would also like to acknowledge Sharon Kelly for administrative support; Ronnie Roger, Benoit Guilhabert, Ian Watson for cleanroom support; Lewis Hannah for machining work.

Chapter 1

Motivation

1.1 Introduction

Cortical implants represent a key neurotechnology, garnering significant attention in recent years due to their clinical implications. These technologies can allow paralysed patients to once again interact with the world through their motor intentions [1, 2, 3] or restore visual sensations to blind patients through stimulation of their visual cortex [4]. Recent developments in high channel-count cortical surface stimulators can facilitate information transfer to brain networks in a minimally-invasive manner [5]. Implantable devices capable of delivering patterned electrical stimulation to the cortical surface are therefore likely to represent a translational medical intervention for patients to address lost function or sensation. One such example explored in this project is an auditory cortical prosthesis, which could restore hearing by bypassing the inner ear and auditory pathway, and stimulating the auditory cortex directly. The motivations of this approach are two-fold; firstly, to restore hearing to patients unable to benefit from the state-of-the-art auditory prosthesis - the cochlear implant - as their acoustic nerve which transmits information from the inner ear to the brain may be missing or damaged. Secondly, there is the potential to improve upon the performance of the cochlear implant, enhancing the quality of the perception closer to normal hearing due to the increased area of the cortex, which offers a greater possible number of stimulation sites than the cochlea [6]. Electrical stimulation is not cell-type specific however, instead driving broad neuronal activity within the receptive field. This limits our ability to understand the circuit dynamics involved in the relevant perceptual processes. Optogenetics is a technique offering cell-type specific stimulation through optical activation of light-sensitive proteins (opsins) which are expressed in neurons after genetic transfection [7]. Therefore, the selective activation of, for example, excitatory or inhibitory pathways becomes possible, furthering our understanding of how best to stimulate neuronal circuits in next-generation cortical prostheses.

This thesis details the work undertaken to develop an implantable optogenetic cortical

surface stimulator based on an array of 100 µLEDs (micro-light emitting diodes). This device was designed specifically for chronic studies in the mouse cortex. Translation of an optogenetic cortical surface stimulator to the clinic is currently unlikely, with limited evidence of the safety and efficacy of the technique in human studies [8]. Nevertheless, cortical surface stimulation remains an exciting current direction in the field of neurotechnology, highlighting the significant role novel devices for neuroscience studies into cortical function will play, such as the one detailed here [9].

1.2 Literature review

This literature review focusses on implantable approaches utilising LEDs or µLEDs for optogenetic interrogation of the cortex. Although, many approaches have been investigated in the literature, ranging from scanning laser (single-photon [10] or two-photon [11]) or digital light projector (DLP) [6] systems. Implantable approaches utilising LED technology include penetrating optical probes or surface stimulators, such as the one detailed in this report. Implantable neural probes typically consist of a shank, similar in geometry to the recording probes ubiquitous in neuroscience laboratories such as Neuropixels [12]. Optogenetic neural probes instead incorporate light emitters, which can be based on waveguide [13] or LED/µLED technology [14, 15, 16, 17]. The latter offers fully implantable capabilities, with the potential for wireless operation in freely-behaving rodent studies, which are of particular interest in the neuroscience field [18, 19]. µLED neural probes offer very high resolution at depth but they are invasive, and spatiotemporal stimulation across the surface of the cortex is typically not a goal. The Utah optrode array (UOA) offers optical stimulation capabilities across the surface of the cortex with an array of 100 penetrating optrodes [20]. Like its electrical analog [3], the Utah array is targeted for non-human primate (NHP) studies and is too invasive for the mouse model, limiting its use to a smaller number of laboratories. The device presented here is purely a cortical surface stimulator, with the same number of sites (100) and a slightly smaller geometry (2 x 2 mm^2 versus 4 x 4 mm^2 for the UOA) suitable for the auditory cortex of the mouse model.

Like the device presented here, minimally-invasive cortical surface stimulators have been developed to address this challenge of high-density modulation. A µLED array coupled to a single-photon avalanche diode (SPAD) array was developed allowing simultaneous optogenetic modulation and fluorescence imaging of the cortical surface [21]. This device was optmised for imaging and with relatively large µLED spacing ($\sim$1 mm), unsuitable for the density of information transfer needed for the mouse auditory cortex in our application, of a few 100 microns [6]. Chronically implantable arrays of LEDs have been demonstrated in the NHP [22, 23, 24] and rodent models [25, 26, 27, 28]. Two previous examples match the stimulation

site spacing and density of the devices presented in this thesis [29, 30], although they do not demonstrate robust neuronal activations leading to behavioural outcomes. A summary of the previously developed devices that are capable of chronic implantation like ours is given in the appendices (A.2).

1.3 Thesis summary

The key goal of this thesis was to address this challenge, engineering devices that could be readily translated to the neuroscience laboratory and demonstrating their efficacy in freely-behaving mouse experiments. As such, chronically implantable optogenetic cortical surface stimulators have been developed based on arrays of 100 µLEDs, fabricated within the Neurophotonics group (Emma Butt, Yunzhou Cheng - see Acknowledgements), and they have been characterised in the laboratory and through *in-vivo* experiments. The dashed area in Fig. 1.1 summarises the key contributions of this thesis.

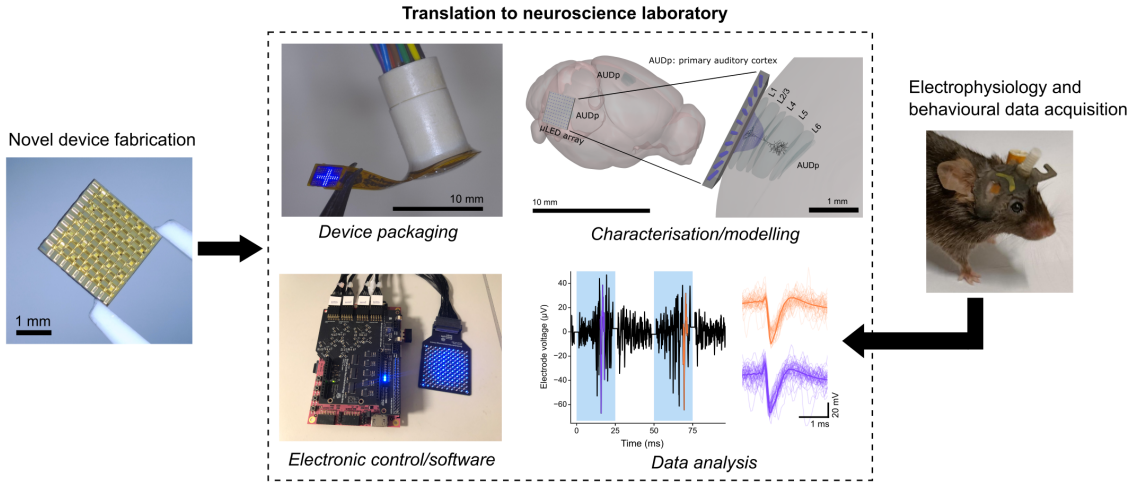


Figure 1.1: **Summary illustration of key contributions of the overall project (within dashed box).** Top-left: device packaging involved integrating custom fabricated µLED arrays to flexible PCBs, capable of chronic implantation in the mouse cortex. Top-right: laboratory characterisation and modelling was used to understand the devices' operating parameters and to validate *in-vivo* results. Bottom-left: the electronic driver system was developed to enable seamless control of devices by neuroscience collaborators during experiments. Bottom-right: the electrophysiology and behavioural data were analysed to further develop the operational framework of the devices and validate their *in-vivo* efficacy. µLED array device fabrication and the execution of the *in-vivo* experiments themselves, shown outside the dashed box, were carried out by team members and collaborators (see Acknowledgments).

Chapter 2 provides the necessary background and an understanding of the prior work carried out to design and fabricate the µLED arrays used in this project. Chapter 3 then details the integration of these devices into systems enabling *in-vivo* experimentation in the mouse

model. The design was adapted for chronic implantability with lifetime experiments being carried out demonstrating its suitability in long-term chronic studies. Chapter 4 provides optical and thermal laboratory characterisation and modelling. Chapter 5 details the *in-vivo* experimentation; actual experiment and data collection were carried out in the Bathellier laboratory (see Acknowledgements); here data analysis has been carried out to establish the operating parameters of the devices. Chapter 6 provides an overview of the electronic systems developed for this project to enable sophisticated experimentation incorporating closed-loop and deep learning-based algorithms. Finally, Chapter 7 concludes and provides discussion into further work that may be carried out based on this project.

Bibliography

1. Lorach, H. *et al.* Walking naturally after spinal cord injury using a brain–spine interface. en. *Nature* **618,** 126–133. ISSN: 1476-4687. doi:10.1038/s41586-023-06094-5. (2026) (June 2023).
2. Musk, E. & Neuralink. An Integrated Brain-Machine Interface Platform With Thousands of Channels. EN. *Journal of Medical Internet Research* **21,** e16194. doi:10.2196/16194. (2026) (Oct. 2019).
3. Maynard, E. M., Nordhausen, C. T. & Normann, R. A. The Utah intracortical Electrode Array: a recording structure for potential brain-computer interfaces. eng. *Electroencephalography and Clinical Neurophysiology* **102,** 228–239. ISSN: 0013-4694. doi:10.1016/s0013-4694(96)95176-0 (Mar. 1997).
4. Fernández, E. *et al.* Visual percepts evoked with an intracortical 96-channel microelectrode array inserted in human occipital cortex. en. *The Journal of Clinical Investigation* **131.** ISSN: 0021-9738. doi:10.1172/JCI151331. (2026) (Dec. 2021).
5. Mestais, C. S. *et al.* WIMAGINE: wireless 64-channel ECoG recording implant for long term clinical applications. eng. *IEEE transactions on neural systems and rehabilitation engineering: a publication of the IEEE Engineering in Medicine and Biology Society* **23,** 10–21. ISSN: 1558-0210. doi:10.1109/TNSRE.2014.2333541 (Jan. 2015).
6. Ceballo, S., Piwkowska, Z., Bourg, J., Daret, A. & Bathellier, B. Targeted Cortical Manipulation of Auditory Perception. eng. *Neuron* **104,** 1168–1179.e5. ISSN: 1097-4199. doi:10.1016/j.neuron.2019.09.043 (Dec. 2019).
7. Deisseroth, K. Optogenetics. eng. *Nature Methods* **8,** 26–29. ISSN: 1548-7105. doi:10.1038/nmeth.f.324 (Jan. 2011).
8. Sahel, J.-A. *et al.* Partial recovery of visual function in a blind patient after optogenetic therapy. *Nature medicine* **27,** 1223–1229. (2026) (2021).
9. Greer, R. *et al.* Chronically implantable LED arrays for optogenetic cortical surface stimulation in mice. en. *Nature Communications* **17,** 878. ISSN: 2041-1723. doi:10.1038/s41467-025-68191-5. (2026) (Jan. 2026).

10. Pinto, L. *et al.* Task-Dependent Changes in the Large-Scale Dynamics and Necessity of Cortical Regions. *Neuron* **104,** 810–824.e9. ISSN: 0896-6273. doi:10.1016/j.neuron.2019.08.025. (2026) (Nov. 2019).

11. Adesnik, H. & Abdeladim, L. Probing neural codes with two-photon holographic optogenetics. en. *Nature Neuroscience* **24,** 1356–1366. ISSN: 1546-1726. doi:10.1038/s41593-021-00902-9. (2026) (Oct. 2021).

12. Steinmetz, N. A. *et al.* Neuropixels 2.0: A miniaturized high-density probe for stable, long-term brain recordings. *Science* **372,** eabf4588. doi:10.1126/science.abf4588. (2026) (Apr. 2021).

13. Chen, F.-D. *et al.* Implantable silicon neural probes with nanophotonic phased arrays for single-lobe beam steering. en. *Communications Engineering* **3,** 182. ISSN: 2731-3395. doi:10.1038/s44172-024-00328-8. (2026) (Dec. 2024).

14. McAlinden, N., Gu, E., Dawson, M. D., Sakata, S. & Mathieson, K. Optogenetic activation of neocortical neurons in vivo with a sapphire-based micro-scale LED probe. English. *Frontiers in Neural Circuits* **9.** ISSN: 1662-5110. doi:10.3389/fncir.2015.00025. (2026) (May 2015).

15. Wu, F. *et al.* Monolithically Integrated LEDs on Silicon Neural Probes for High-Resolution Optogenetic Studies in Behaving Animals. English. *Neuron* **88,** 1136–1148. ISSN: 0896-6273. doi:10.1016/j.neuron.2015.10.032. (2026) (Dec. 2015).

16. Vöröslakos, M. *et al.* HectoSTAR LED Optoelectrodes for Large-Scale, High-Precision In Vivo Opto-Electrophysiology. en. *Advanced Science* **9,** 2105414. ISSN: 2198-3844. doi:10.1002/advs.202105414. (2026) (2022).

17. Taal, A. J. *et al.* Optogenetic stimulation probes with single-neuron resolution based on organic LEDs monolithically integrated on CMOS. en. *Nature Electronics* **6,** 669–679. ISSN: 2520-1131. doi:10.1038/s41928-023-01013-y. (2026) (Sept. 2023).

18. Park, S. I. *et al.* Soft, stretchable, fully implantable miniaturized optoelectronic systems for wireless optogenetics. eng. *Nature Biotechnology* **33,** 1280–1286. ISSN: 1546-1696. doi:10.1038/nbt.3415 (Dec. 2015).

19. Shin, H. *et al.* Transcranial optogenetic brain modulator for precise bimodal neuromodulation in multiple brain regions. en. *Nature Communications* **15,** 10423. ISSN: 2041-1723. doi:10.1038/s41467-024-54759-0. (2026) (Nov. 2024).

20. Clark, A. M. *et al.* An optrode array for spatiotemporally-precise large-scale optogenetic stimulation of deep cortical layers in non-human primates. en. *Communications Biology* **7,** 329. ISSN: 2399-3642. doi:10.1038/s42003-024-05984-2. (2026) (Mar. 2024).

21. Pollmann, E. H. *et al.* A subdural CMOS optical device for bidirectional neural interfacing. en. *Nature Electronics* **7,** 829–841. ISSN: 2520-1131. doi:10.1038/s41928-024-01209-w. (2026) (Sept. 2024).

22. Rajalingham, R. *et al.* Chronically implantable LED arrays for behavioral optogenetics in primates. en. *Nature Methods* **18,** 1112–1116. ISSN: 1548-7105. doi:10.1038/s41592-021-01238-9. (2026) (Sept. 2021).

23. Komatsu, M., Sugano, E., Tomita, H. & Fujii, N. A Chronically Implantable Bidirectional Neural Interface for Non-human Primates. English. *Frontiers in Neuroscience* **11.** ISSN: 1662-453X. doi:10.3389/fnins.2017.00514. (2026) (Sept. 2017).

24. Ohta, Y. *et al.* Micro-LED Array-Based Photo-Stimulation Devices for Optogenetics in Rat and Macaque Monkey Brains. *IEEE Access* **9,** 127937–127949. ISSN: 2169-3536. doi:10.1109/ACCESS.2021.3111666. (2026) (2021).

25. Ji, B. *et al.* Flexible polyimide-based hybrid opto-electric neural interface with 16 channels of micro-LEDs and electrodes. en. *Microsystems & Nanoengineering* **4,** 27. ISSN: 2055-7434. doi:10.1038/s41378-018-0027-0. (2026) (Oct. 2018).

26. Kwon, K. Y., Sirowatka, B., Weber, A. & Li, W. Opto-ECoG Array: A Hybrid Neural Interface With Transparent ECoG Electrode Array and Integrated LEDs for Optogenetics. *IEEE Transactions on Biomedical Circuits and Systems* **7,** 593–600. ISSN: 1940-9990. doi:10.1109/TBCAS.2013.2282318. (2026) (Oct. 2013).

27. Lee, S. H. *et al.* Optogenetic control of body movements via flexible vertical light-emitting diodes on brain surface. *Nano Energy* **44,** 447–455. ISSN: 2211-2855. doi:10.1016/j.nanoen.2017.12.011. (2026) (Feb. 2018).

28. Wu, M. *et al.* Patterned wireless transcranial optogenetics generates artificial perception. en. *Nature Neuroscience,* 1–12. ISSN: 1546-1726. doi:10.1038/s41593-025-02127-6. (2025) (Dec. 2025).

29. Sekiguchi, H. *et al.* Adhesionable flexible GaN-based microLED array film to brain surface for in vivo optogenetic stimulation. en. *Applied Physics Express* **15,** 046501. ISSN: 1882-0786. doi:10.35848/1882-0786/ac5ba3. (2026) (Mar. 2022).

30. Gu, W., Wang, L., Wang, X., Zhao, C. & Guan, S. Large-Scale, High-Density MicroLED Array-Based Optogenetic Device for Neural Stimulation and Recording. *Nano Letters* **24,** 15252–15259. ISSN: 1530-6984. doi:10.1021/acs.nanolett.4c03645. (2026) (Dec. 2024).

Chapter 2

Background and theory

This project centres around developing an optogenetic cortical implant for neuroscience studies into cortical function. Detailed descriptions of the methodologies are provided in the relevant results chapters, and the following background provides the prerequisite information in the context of relevant literature.

This chapter is structured as follows; some key concepts surrounding the brain and neuronal processes are introduced, before exploring some of the specificities of auditory processing which are relevant to this project. Then the state-of-the-art in cortical surface stimulation technologies are introduced, before explaining how optogenetics works, and the current approaches for implantable optogenetic stimulation to which the technology developed in this project is competing. Light-emitting diodes are explained, and the process that was developed in the lab, to fabricate the μLED arrays used here, is also outlined. Lastly, some background is provided into signal processing and closed-loop control which is also a key focus of this project.

2.1 The brain, cerebral cortex and neurons

The brain is our most advanced organ, responsible for receiving and processing information, and coordinating our bodies through motor and muscle control. The brain is composed of tens of billions of interconnected neurons which communicate using chemical processes and electrical signalling; these neurons are organised into different regions, each serving a specific purpose [1]. The cerebral cortex is the outermost layer and is responsible for a multitude of sensory and cognitive processes. In large mammals (humans and large non-human primates) it is wrinkled or folded to increase its surface area and hence processing capacity. Fig. 2.1a and 2.1b shows a schematic of the human brain indicating the key regions and the axis and cross-section labelling which are referred to in the neuroscience literature.

The basic structure of a neuron comprises the cell body, dendrites (which receive infor-

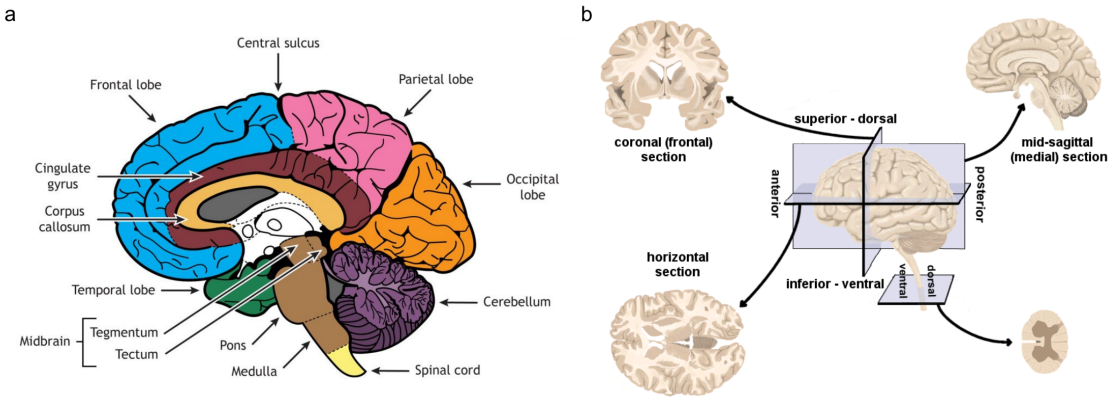


Figure 2.1: **Schematics of the human brain. (a)** Schematic illustration of human brain with key regions annotated [2]. **(b)** Standard brain axis and cross-section labelling [3].

mation) and axons (which transmit information) (Fig. 2.2a). There are also many types of neuron, such as pyramidal neurons which are primarily targeted in this work; these are of the excitatory type, but there are also inhibitory neurons. Neurons communicate with one another via electrical signalling which arise from ionic gradients which traverse the cell membrane. Action potentials then propagate along the axons to be received at the dendritic inputs of other neurons in the circuit [4]. These action potentials, or spikes, can be recorded using electrophysiology, usually with tissue-penetrating neural recording probes [5] (Fig. 2.2b) or surface recording, known as electrocorticography (ECoG) [6]. Depending on the frequency band recorded from, we can identify population activity such as local field potential (LFP) [7], multi-unit analysis (MUA) [8] or single-unit spiking activity. Identification of the latter is typically automated using clustering algorithms such as Kilosort [9]. Connections between neurons are known as synapses and are able to adapt to reduce or strengthen their connection.

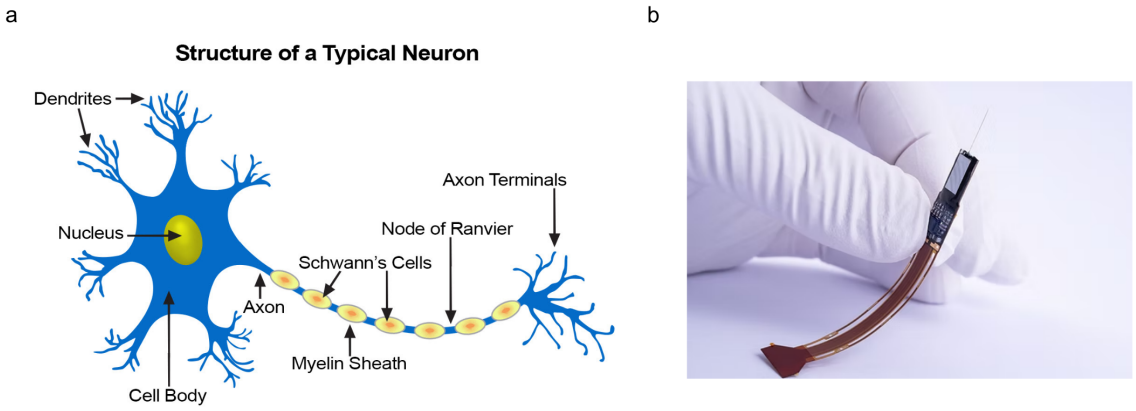


Figure 2.2: **Neurons and neuronal recording. (a)** Basic structure of a neuron annotated. [10]. **(b)** Neuropixels linear electrode array with 384 neural recording channels - these are located on the very thin shank extending from the top-right of the device. The remainder of the device constitutes the interconnection and processing electronics [5].

2.2 Auditory cortex and the auditory pathway

The auditory cortex is the target for optogenetic stimulation using the devices developed in this thesis. Although this is only a single piece of the auditory system which extends from the inner ear (cochlea) through the auditory midbrain, containing the inferior colliculus, and eventually the most downstream processing structure; the auditory cortex. Figure 2.3a shows a schematic of the auditory pathway [11]. The intention is not to describe this in detail here, rather to emphasise the complexity of this system.

The most common auditory prosthetic is the cochlear implant, which is regarded as the most successful neuroprosthesis, recently surpassing one million implantations globally [14]. A small cohort of deaf patients, who have a damaged or missing acoustic nerve, are unable to benefit from a cochlear implant and can receive an auditory brainstem implant instead [15]. However, the latter can often be ineffective, motivating the stimulation of more downstream systems; the natural choice being the auditory cortex due to its large surface area and relative ease-of-access through a craniotomy.

Stimulation of the auditory cortex is known to elicit sound perceptions in the brain [16] making this a potential candidate for an auditory cortical prosthesis. A detailed description of the cell types and structure of the auditory cortical circuit is outwith the scope of this report. Previous studies have aimed to describe the auditory cortex in detail [17]. However, like its neighbouring regions across the neocortex, the auditory cortex is organised in six distinct layers, serving different purposes in the cortical circuit such as facilitating input and output connectivity [18]. Layer 5 pyramidal neurons are thought to convey processed auditory information, and are the targets for optogenetic stimulation in this thesis and in

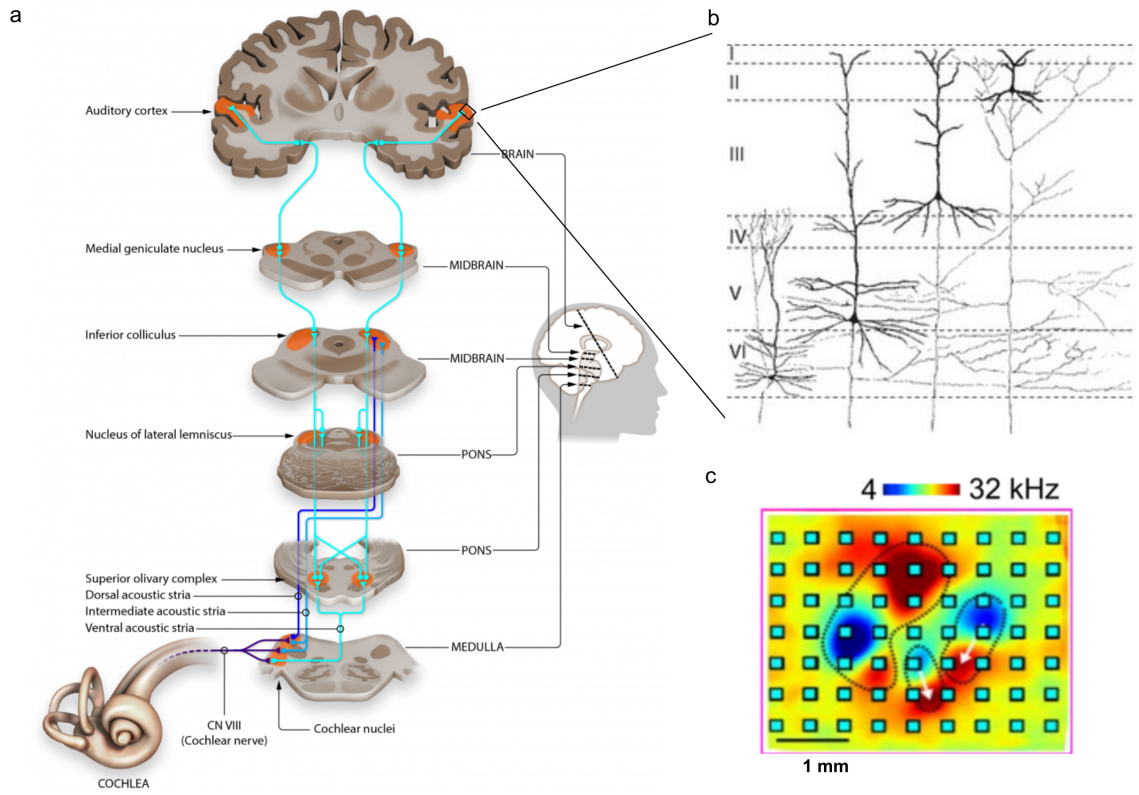


Figure 2.3: **Auditory cortex background. (a)** Schematic illustration of auditory pathway showing processing structures which extend from the inner ear (cochlea) to the auditory cortex [11]. **(b)** Golgi stain image of human cortex cross-section showing cortical layers, including layer 5 pyramidal neurons with dendrites extending to layer 2/3 [12]. **(c)** Tonotopic image of mouse auditory cortex, acquired through intrinsic imaging of blood flow during presentation of different frequencies of sound [13].

the previous work [13] (Fig. 2.3b). Their dendrites extend to layer 2/3 and also express the optogenetic construct used in this work, therefore being sensitive to blue light stimulation [19]. This is described in more detail in the upcoming section on optogenetics, and in Chapter 5. Furthermore, the auditory cortex is known to have a tonotopic arrangement [20], giving rise to the possibility of an encoding approach for a cortical prosthesis (Fig. 2.3c). Due to the complexity of the auditory system however (Fig. 2.3a), a more complex encoding approach, beyond the simple tonotopic approach like with cochlear implants, is likely to be required [21].

2.3 Cortical surface stimulation methodologies

Clinical neurotechnologies that stimulate the brain typically deliver electrical charge to modulate circuit activity or transmit information to the nervous system. For example, the deep brain stimulator uses a several centimetre-long penetrating electrode which activates the

deep-brain and can treat the symptoms of neurodegenerative diseases such as Parkinsonian tremor and epileptic seizures [22]. Cortical stimulation in clinical settings often employs the Utah electrode array (UEA); an array of microelectrodes which penetrate the brain (Fig. 2.4a) [23]. The UEA has been used for stimulation of the visual cortex in human patients [24, 25] and intra-cortical micro-stimulation (ICMS) has been used to experiment with proprioceptive feedback in patients with a motor brain-machine interface (BMI), enabling improved control of a prosthetic limb [26].

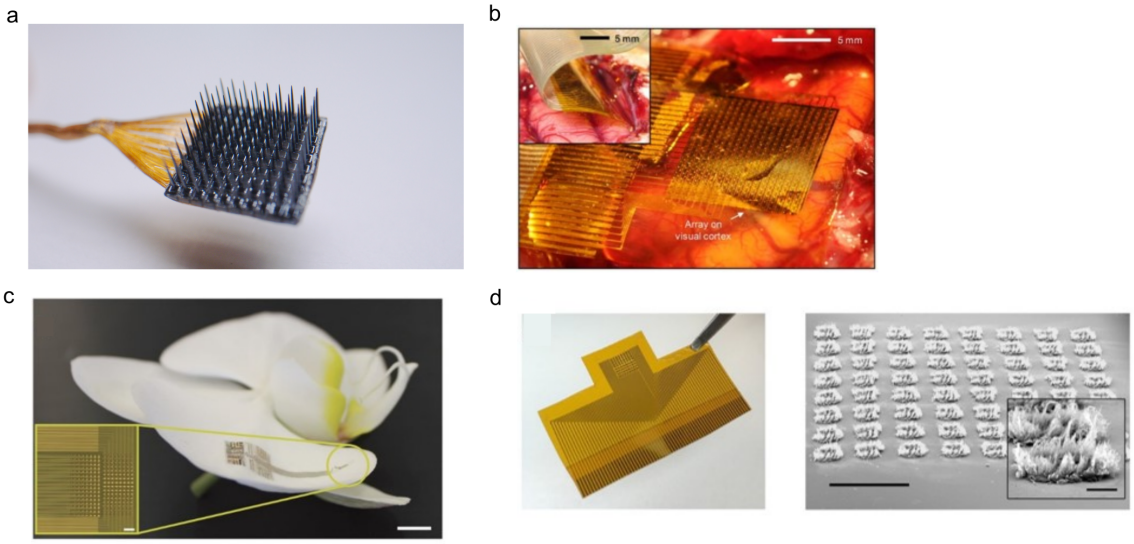


Figure 2.4: **Electrocorticography and electrical stimulation technologies. (a)** Utah electrode array [27] **(b)** Large-scale electrode surface recording; scale bars are 5 mm [6]. **(c)** Large-scale electrode surface recording; main scale bar is 5 mm, inset scale bar is 100 µm [28]. **(d)** Cortical surface electrode device optimised for stimulation; on scanning electron micrograph (SEM) image, main scale bar is 1 mm and inset scale bar is 100 µm [29].

Electrocorticography (ECoG) devices (state-of-the-art examples shown in Fig. 2.4b/2.4c), traditionally used to map cortical potentials by placing electrodes at the surface of the cortex, can be adapted for surface stimulation. Minimally-invasive cortical surface stimulation at comparable scale to that required in this project has been tested with novel device materials in the rodent model. These incorporate low impedance electrode materials and geometries to optimise charge injection capacity (CIC) (Fig. 2.4d) [30, 31, 29].

2.4 Optogenetics

Optogenetics is a relatively recently developed technique, now ubiquitous in neuroscience laboratories across the world, which enables optical activation, or inhibition, of neurons *in-vivo*, with milli-second temporal precision and cell-type specificity (fig. 2.5a) [32]. The neurons are

transfected with a light-sensitive protein known as an opsin. These opsins exist in nature, for example in the blue-light-sensitive algae (*Chlamydomonas reinhardtii*) for the channelrhodopsin 2 (ChR2) opsin; used extensively in neuroscience laboratories and in this report. Recently the opsin, ChrimsonR (red-shifted, therefore sensitive to higher wavelengths, in this case ∼590 nm) was used in a clinical trial in the retinal ganglion neurons of a blind patient, and was shown to partially recover visual function [33]. There are a raft of opsins, exhibiting excitatory or inhibitory effects on the neuron, and operating at a range of timescales (kinetics) and wavelengths (Fig. 2.5b). The molecular mechanism of all opsins is based on the same principle. They belong to the family of seven-transmembrane proteins (7TM), which contain a retinal chromophore that changes configuration upon absorption of a photon. When introduced to the membrane of the neuron (soma, dendrites or axons), the membrane potential is modified. For channelrhodopsins (like ChR2, ChrimsonR, ChRmine), these form ion (cation) channels which open upon optical stimulation, allowing charge flow across the neuronal membrane, and can cause the neuron to fire. In contrast, opsins that introduce anion channels (GtACR1/2) can be used to inhibit neuronal activity upon light activation. The mechanism can also be through active ion transport by light-driven ion pumps (halorhodopsin, archaerhodopsin). Opsins can be of the Type I (microbial), as reported in this work, or Type II (animal). Opsins are genetically introduced to the cells either through adeno-associated vector (AAV) transfection (viral injection) or through breeding of animals (transgenic). Cell-type specificity can be achieved through cell-specific promoters or transgenic Cre lines, which enable preferential expression in defined neuronal populations such as excitatory, or inhibitory, networks [34, 35].

Clinical research has not gone further with optogenetics currently, due to ongoing safety and regulatory challenges which are still to be addressed. Therefore, electrical stimulation is likely to continue to be used for neurostimulation therapies. Nevertheless, optogenetics is a very useful tool for neuroscience studies, as the transfection method can be engineered to express the opsin in specific subtypes of neurons, for example excitatory or inhibitory. This allows more specific research into neuronal circuits and pathways, potentially informing how best to stimulate the brain in future interventions even if these eventually use electrical stimulation. Transfection of the opsin can be delivered via a viral vector (injection) into the target brain region, or with transgenic animals which guarantee more uniform opsin expression. Transgenic mice expressing ChR2 in all excitatory neurons, the primary target for optogenetic stimulation of the auditory cortex, have been used throughout this work [37, 19]. Although we also explored ChRmine which is a highly light-sensitive red-shifted opsin [38]. Despite being red-shifted, it has a broad absorption spectrum, meaning it is still approximately 5-10x more sensitive to the blue light delivered with the μLED arrays described here, than ChR2.

As described in the literature review in Chapter 1, there are a plethora of tools and

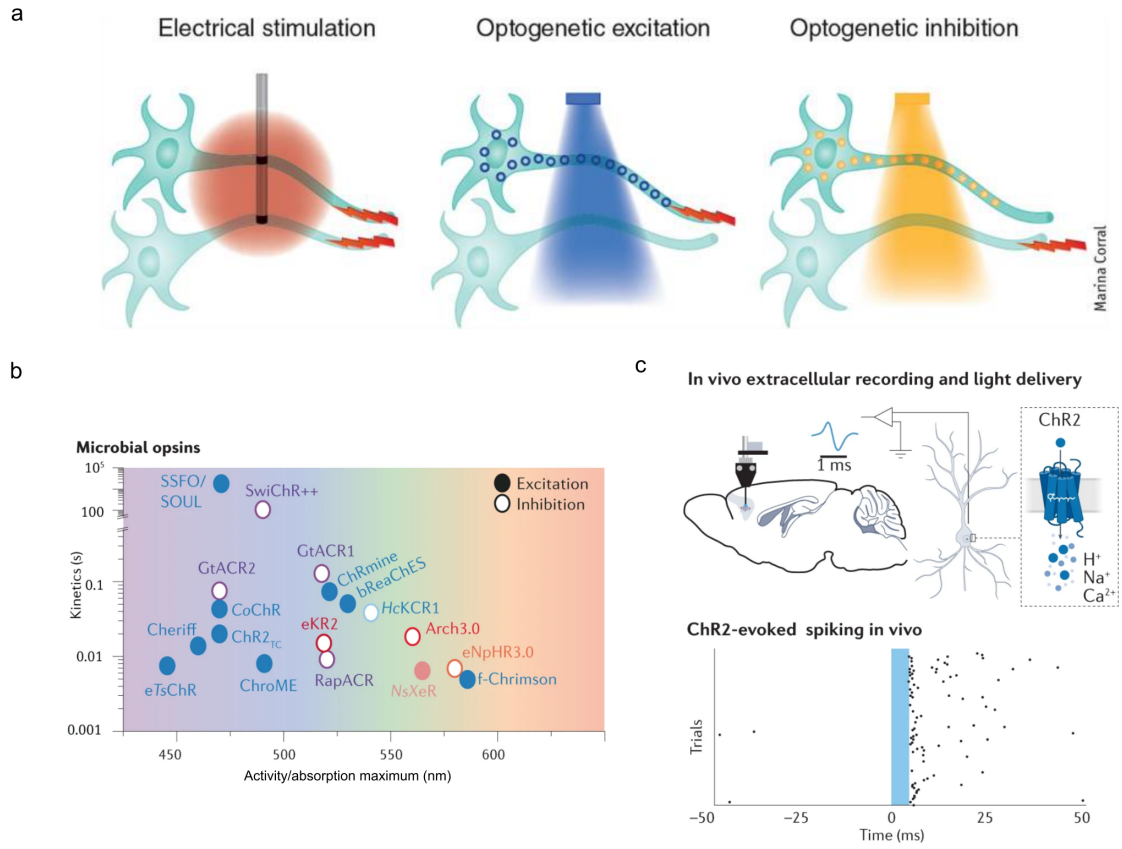


Figure 2.5: **Optogenetics overview. (a)** High-level illustration of optogenetics compared with electrical neuronal stimulation [32]. **(b)** Recent overview of microbial opsin variants, demonstrating their kinetics and wavelength sensitivity [36]. **(c)** Overview of channelrhodopsin mechanism with light delivery resulting in ion flow across a cell soma, and neuronal firing being detected with electrophysiology. The blue light is delivered via fibre-coupled optical probe. For extracellular recording, the amplifier is used to amplify small voltage before they are digitised and recorded (top). Graph of spike times (shown as dots) resulting from optogenetic stimulation with the optical fibre (blue shaded region) (bottom). [36].

technologies that enable optical interfacing with the brain *in-vivo*. Shown in Fig. 2.6 are examples of state-of-the-art implantable devices for optogenetics, with a focus on LED/µLED-based technologies; see section 2.5 for further details. These devices aim to, and already, enrich our understanding of the nervous system. They have enabled, for example, optogenetic experiments to understand the neuronal processes of the visual cortex in behaving non-human primates (NHPs) [39], and the potential for an optical cochlear implant, to overcome the limitations in spectral resolution from current cochlear implants based on electrical stimulation [40]. These devices will enable advanced neuroscience studies, and undoubtedly deliver the necessary insights in the development of next-generation clinical neurotechnologies.

Several considerations must be made when using optical technologies to activate light-

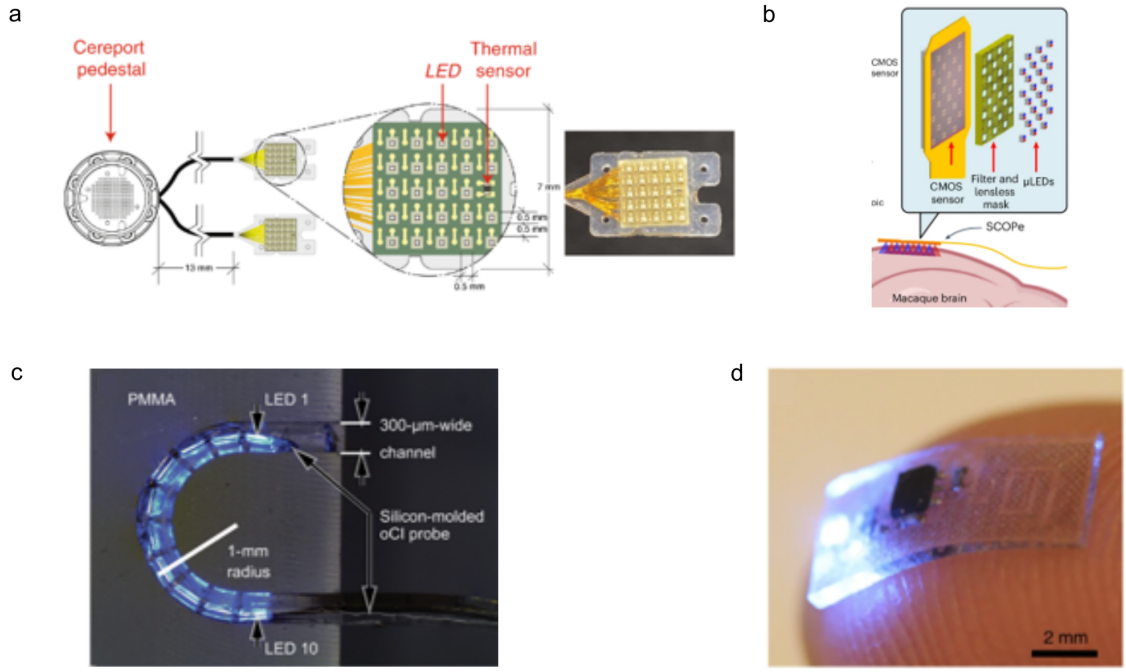


Figure 2.6: **Implantable optogenetic tecnhologies. (a)** Chronically implatable optogenetic cortical surface stimulator for non-human primate (NHP) [39]. **(b)** Dual-colour optogenetic cortical surface stimulator and flourescence imaging device [41]. **(c)** Optogenetic cochlear implant [40]. **(d)** Wireless, chronically implantable LED stimulator for optogenetics in rodent [42].

sensitive neurons with optogenetics [43]. Opsins have a threshold irradiance range, above which the neurons will fire. For ChR2 used in this work, this threshold is commonly quoted as 1 mW/mm^2 and this has been used as a benchmark here, for example in the optical models. This threshold can be significantly lower (factor of 100 times) for more sensitive opsins like ChRmine [38]. The spectrum of the opsin should be examined as the source may not be at the optimal wavelength for activation. The second consideration is the temporal dynamics of the opsin. When activated, the channel needs a certain amount of time to recover, ranging from a few milliseconds to tens or even hundreds of milliseconds for slower opsins [44]. Generally, more sensitive opsins have longer recovery times [45]. Lastly, thermal effects due to optical stimulation must be considered to ensure neuronal modulation is the result of optogenetic stimulation and not thermally-induced [46]. This issue is particularly acute when using implantable sources such as µLEDs for optogenetic stimulation due to the heat these devices generate, and their proximity to the brain tissue. This necessitates an in-depth thermal analysis depending on specific device design [47], such as the one detailed in this thesis.

2.5 Light emitting diodes

Light-emitting diode (LED) technology is the basis of the devices developed in this work. LEDs are made from doped semi-conductor materials, forming an anode (positive terminal) and cathode (negative terminal). Electrical energy is converted to light when electrons and holes undergo radiative recombination in the p-n junction (active region), as originally described in [48]. Bandgap engineering is used to control the emission wavelength, for example using certain materials like indium gallium nitride (InGaN) to generate blue light [49]. Modern, highly efficient LEDs use quantum well structures to increase the recombination probability [50]. LEDs produce a non-linear electrical characteristic, switching on at a minimum forward voltage, at which point significant current flows and light emission is produced by radiative recombination as described above. Generally, the turn-on voltage (defined as the voltage required to drive a specific current or achieve a specified light output) is related to the material bandgap energy. For commercial blue LEDs, this is typically 2-3 V; at an operating current <10 mA and often employing pulsed operation at a set duty cycle to save power and achieve arbitrary brightness. A further advantage of pulsing LEDs is reduced thermal effects from device self-heating. Electrical power is converted to light power according to the quantum efficiency (internal quantum efficiency (IQE) and external quantum efficiency (EQE)) of the LED, defined as follows:

$$IQE = \frac{N_{\text{photons generated}}}{N_{\text{electron-hole recombinations}}} \tag{2.1}$$

$$EQE = \frac{N_{\text{photons emitted}}}{N_{\text{electrons injected}}} \tag{2.2}$$

These quantities depend on materials, device fabrication and operating conditions with typical LED efficiency (that is external quantum efficiency) being as low as 10-20%. If interested further, the reader may peruse the provided references [48, 49, 50]. For this thesis, a high-level understanding of LEDs and specifically their electrical characteristics is sufficient.

In the context of optogenetic devices, two distinct types of LED and fabrication processes should be distinguished; inorganic (typically denoted LED/µLED or sometimes ILED/µILED) and organic (denoted OLED). The distinction between LED and µLED refers to the size of the device, whereby the diode is less than approximately 100 µm in dimension [51]. Inorganic LEDs are made from materials such as gallium nitride (GaN) and fabricated on substrates like silicon or sapphire. These LEDs are very efficient and stable and are the type used to develop the optogenetic cortical implants presented in this thesis. OLEDs are made from organic (carbon-based) semiconductor materials, however they cannot be driven at very high current densities like their inorganic counterparts, resulting in limited brightness. They are, however,

much easier to fabricate at scale, utilising processes such as thermal vapour deposition [52]. Generally, inorganic LEDs can provide orders of magnitude higher brightness at lower power than OLEDs [53]. The monolithic fabrication of multiple colours is easier when using OLEDs and they can be fabricated directly on CMOS (complementary metal oxide semiconductor), enabling massive scalability [54, 41]. However, they may not be bright enough to reach activation threshold for the more common optogenetic opsins like channel-rhodopsin 2 (ChR2), to achieve reliable responses [54]. Encapsulation strategies for OLED-based devices are also crucial due to their inherent reduced stability. Despite the limitations of OLEDs, great strides have been made in the development of OLED-based optogenetic neural probes which were recently demonstrated successfully *in-vivo* [54]. It is possible to fabricate multiple-colour, or CMOS-based, optogenetic neural probes based on µLEDs, but the µLEDs must be transferred from a donor substrate, for example using transfer printing, which involves picking and placing µLEDs individually with a small elastomer stamp [55]. This limits production to small-batch processes, however mass transfer printing methods are being proposed and have been successfully utilised in developing µLED cortical surface stimulators with similar density and spacing of stimulation sites to our work [56, 57]. However, the stability of these devices over long timescales *in-vivo* remains unclear. Furthermore, transfer printed µLEDs, extracted as membranes from GaN-on-sapphire wafers through laser liftoff, can lose efficiency if there are defects in the GaN crystal structure, especially when the GaN layers are grown on a material that is lattice mismatched [58]. Therefore, the brightness may not be high enough to achieve robust neuronal or behavioural responses. In the context of the referenced papers using this approach, limited *in-vivo* results are presented. Thorough thermal analysis is also crucial when active devices like LEDs are positioned within or close to brain tissue.

2.6 µLED array fabrication

With minimally-invasive optogenetic cortical implants, the µLEDs reside on the surface of the brain and light must penetrate through the thin, but significant, layer of white matter (layer 1) before reaching the target neurons. In this work, the targets for optogenetic stimulation have been identified as the apical dendrites of layer 5 pyramidal neurons which extend into layer 2/3, at a depth of approximately 150 µm into the cortical tissue. To illuminate this depth with sufficient light intensity to reliably activate ChR2-expressing neurons, the InGaN-on-sapphire µLED array fabrication process was utilised [59]. These µLEDs emit blue light at a wavelength of ∼450 nm. The wafers consist of an n-doped GaN (nGaN) bottom layer, an unintentionally doped InGaN/GaN multi-quantum well region, an AlGaN electron blocking layer, and a p-doped GaN (pGaN) top layer atop a patterned sapphire substrate (PSS). The PSS helps moderate strain in the QW structures, resulting in more efficient and therefore bright

μLEDs [58]. For our fabricated μLEDs, full optical characterisations are provided in Chapter 4, however, they are capable of exceeding 3000 mW/mm^2 on the μLED surface, which we show corresponds to approximately 50 mW/mm^2 on the cortical surface (Fig. 2.8). This will be described in Chapter 4, where optical models are developed to help characterise the fabricated μLEDs, showing this difference in irradiance is due the Lambertian emission of the μLEDs and the spatial offset caused by the sapphire substrate which the light must pass through before reaching the brain.

The μLED arrays used for this project were designed and fabricated within the group (Emma Butt & Yunzhou Cheng; see Acknowledgements) in an in-house semi-conductor cleanroom facility. This thesis details the work undertaken, post-fabrication, to integrate these devices into systems capable of implantation in the mouse cortex to enable *in-vivo* experiments. This work also details the supporting characterisation and modelling of these devices, electronic driver system design and analyses of the resulting *in-vivo* experimental data. For completeness, the fabrication steps are illustrated in Fig. 2.7 and textually summarised below, where numbering corresponds to the figure.

The wafer was thinned to 150 μm and the emitting side optically polished prior to processing. A palladium layer (Pd) 100nm is deposited using an E-beam evaporator, to define the 40 μm pixel shape and also act as a current spreading layer (i). A thin photoresist layer is used to pattern oxide using an RIE tool (ii), which is used as a mask to etch (ICP) first the Pd layer, followed by the p-GaN material, forming the pixel structure (iii-v). Another oxide mask is then used to etch the n-GaN material forming the pads for interconnection and the n-type mesa, with 10 rows, each with 10 pixels per array forming the 10x10 array (vi-x). A lift-off process using a bi-layer of photoresist deposits a layer of Ti:Au (100 nm:300 nm) along the n-type mesa, the bond pads, as well as a contact on top of each p-type pixel (xi-xiii). Two insulation layers are formed over the n-metal tracks. Firstly, 800 nm of SiO2 is deposited and vias etched open using an RIE tool (xiv). A layer of SU-8 TF 6005 photoresist is spun and patterned then hard baked at a temperature of 200 °C for 2 hours (xv). A lift-off process is then repeated on top of these insulation layers, depositing Ti:Au (100 nm:300 nm) in columns forming p-contacts to the pixels as well as further depositing metal on the wire bonding and flip-chip bonding pads (xvi-xvii). A final layer of SU-8 TF 6005 is spun on top of these tracks, vias opened above the bond pads and hard baked (xviii). A protective layer of photoresist (Megaposit SPE220-4.5) is then spun on top of the wafer. The resulting wafer consists of 220 individual μLED arrays (or chips/die/devices). The devices are individualised using a diamond blade dicing saw.

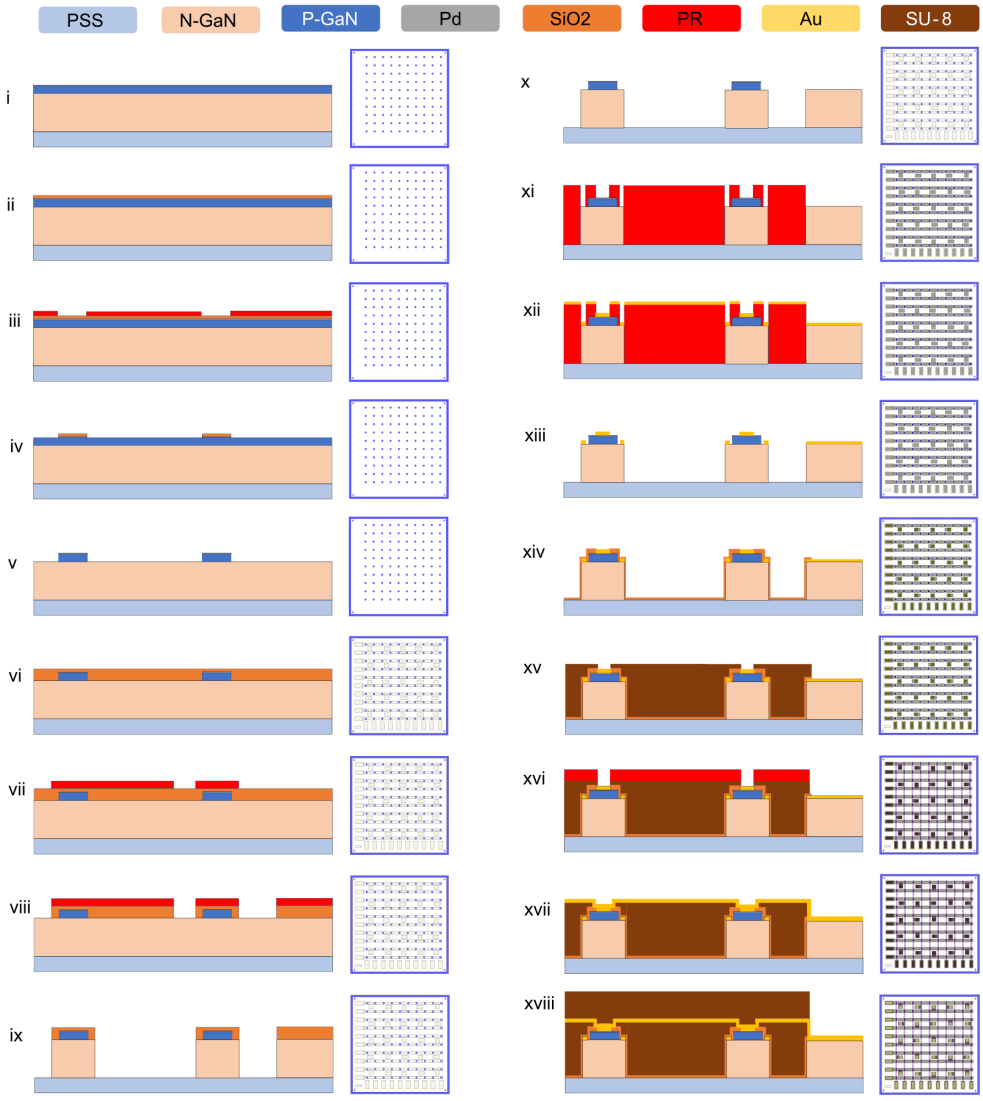


Figure 2.7: **µLED array fabrication process.** Left images show cross-section of two pixels and a wire bond pad. Right images show photolithography hard-mask relevant to each step.

The final device consists of 40 µm square pixels arranged in a 10x10 matrix-addressable array, with 200 µm pixel spacing. This is a passive-matrix device, for which a custom electronic driver system has been developed (Chapter 6). Fig. 2.8 shows representative I-V (current-voltage) and L-I (irradiance current) data. The irradiance calculations are detailed further in Chapter 4. The design and fabrication of these devices we published in [60].

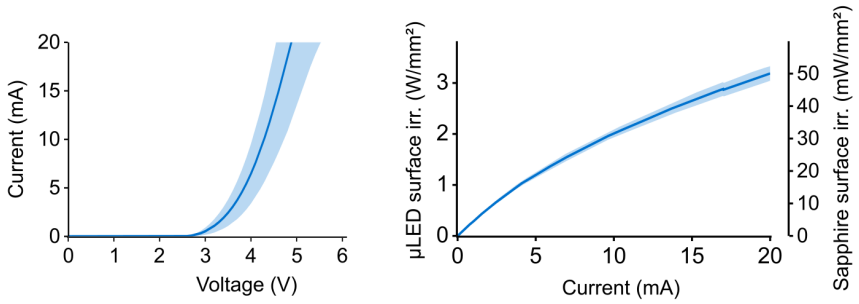


Figure 2.8: **µLED array characteristics. (a)** Current-voltage (I-V) curve. **(b)** Irradiance-current (L-I) curves at the µLED surface and sapphire surface. Both (a) and (b) show mean and standard deviation over n = 10 µLEDs.

2.7 Device encapsulation strategies for long-term chronic implantation

Biomedical implants for clinical use have expectedly delivered extensive research and development into protective encapsulation strategies [61], with standard device operating lifetimes over a decade [62].

Parylene-C is a well-known biocompatible polymer, which can be deposited in a thin, flexible film through a vapour deposition process, providing a conformal coating over devices [63]. Parylene-C is optically transparent to blue light, making it an ideal encapsulant for the µLED arrays, and is therefore a critical component in the encapsulation stack employed in our device packaging process. However, defects in the underlying substrate can result in cracks and voids in the encapsulation which could allow conductive solution ingress. Degradation can selectively build-up around defect areas potentially resulting in further failure mechanisms [64]. Subjecting parylene-C to constant flexing could further exacerbate these failure mechanisms. Silicone, specifically medical-grade silicone, provides excellent encapsulation and is used extensively in the field of medical implants [65]. It has been shown to improve device lifetime *in-vivo* beyond that of parylene-C alone in this work. Medical-grade silicone can be conformally applied to a surface through dip or spray coating methods. These dispersions contain silicone elastomers dispersed in a solvent, whereby the concentration of solvent and coating strategy influences the resulting coating thickness. Thus, we also investigated how best to apply a medical-grade silicone encapsulation to our devices in Chapter 3, and evaluate this through accelerated ageing tests. Although not investigated in this study, fluoropolymers can be applied in conformal coatings are generally considered a better encapsulant than parylene-C [40]. Atomic layer depositions (ALDs) can also be used to provide an excellent, defect-free barrier to solution ingress in implantable devices [66].

2.8 Closed-loop control

There has been a recent move for more autonomy in neural interfacing systems [67], namely in deep-brain stimulation systems which are most commonly operated in an open-loop manner, or by the user. Closed-loop control allows detection of the neural signalling associated with the onset of an epileptic seizure or Parkinsonian tremor and adapt stimulation autonomously [68]. Another example includes closed-loop stimulation for incontinence control, currently undergoing clinical trials [69]. These ideas could be applied to direct interfacing with the cortex, and an example of particular interest is improving the capabilities of brain-machine interfaces for "locked-in" patients who are unable to control their limbs due to paralysis or spinal cord damage. These brain-machine interfaces (BMIs) record motor intentions to control a personal computer device [70], or drive a prosthetic wheelchair [71] or limb [72]. In the latter case, introducing real-time feedback, whereby closed-loop stimulation of the somatosensory cortex is delivered with sufficiently low latency to represent the state of the prosthetic limb, can improve the control of these devices [73]. This work has been extended to optogenetic studies in mice to improve our understanding of the best closed-loop stimulation paradigms [74]. Therefore, there is a clear motivation for developing a system with electrophysiology recording and optogenetic stimulation capabilities, such as the one detailed in Chapter 6.

2.9 Signal processing electronics

Signals are a representation of information that varies in time. In digital systems, signals are sampled in both time (sample rate) and amplitude (a specific number of bits per sample) and can be processed in different ways. Some examples of signal processing include expression of the signal in the frequency domain through a discrete Fourier transform (DFT), or processed through deep neural networks which are the core of modern machine learning and artificial intelligence algorithms [75]. Signal processing is naturally therefore of core importance in neuroscience research (large-scale data analysis) and in the development of advanced neurotechnologies [76, 77, 78].

In the context of this work, and as described in the previous section, cortical stimulation devices will require real-time, closed-loop signal processing capabilities [79]. These could record neuronal signals, or record sensory information, for example from a camera or microphone, and transmit meaningful information to the cortex. Digital signal processors (DSPs) can already be found in cochlear implants, performing frequency decomposition and assigning power to different stimulation channels accordingly. However, as neural stimulation moves to the cortex, this opens physically larger regions of the nervous system necessitating more stimulation channels and increased signal processing capability. Devices such as

ASICs (application-specific integrated circuits) or FPGAs (field-programmable gate arrays) can achieve very high throughput with the potential for parallelism in data processing, useful for deep neural network models. Although tradeoffs in power consumption, design flexiblity and cost/time-to-market must be considered [80]. In this work, system-on-chip (SoC) technology has been employed, incorporating the flexibility of an ARM core with the signal processing capabilities of an FPGA (Zynq) [81]. This is ideal for neuroscience studies, offering sophisticated experimental control and ease-of-use, but power consumption remains an issue for wireless or battery-powered implementations. Dedicated microcontrollers incorporating neural network processing units (NPUs) are entering the market and could address these challenges [82]. These devices operate with power consumption per neural network inference as low as 100 µJ, which, for example, would correspond to several days of battery life assuming an inference every 100 ms and the use of a 100 mAh battery, suitable for a biomedical implant. More capable devices demand higher power consumption and thus careful engineering should be employed to address the challenge of capability versus battery-life/system mass to ultimately drive a positive patient experience.

Bibliography

1. Purves, D. *et al. Neuroscience* 2nd. ISBN: 978-0-87893-742-4 (Sinauer Associates, 2001).
2. Hedges, V. Brain Anatomy. en. Book Title: Introduction to Neuroscience. (2026) (Dec. 2022).
3. Thambusamy, V. & Mohan, M. Efficiency of Fuzzy C Means algorithm for Brain Tumor segmentation in MR Brain Images. *International Journal of Engineering and Technology* **8,** 2979–2989. doi:10.21817/ijet/2016/v8i6/160806261 (Dec. 2016).
4. Kandel, E. R., Koester, J. D., Mack, S. H. & Siegelbaum, S. A. *Principles of Neural Science* 6th. (2026) (McGraw Hill Medical).
5. Steinmetz, N. A. *et al.* Neuropixels 2.0: A miniaturized high-density probe for stable, long-term brain recordings. *Science* **372,** eabf4588. doi:10.1126/science.abf4588. (2026) (Apr. 2021).
6. Viventi, J. *et al.* Flexible, foldable, actively multiplexed, high-density electrode array for mapping brain activity in vivo. en. *Nature Neuroscience* **14,** 1599–1605. ISSN: 1546-1726. doi:10.1038/nn.2973. (2026) (Dec. 2011).
7. Burns, S. P., Xing, D. & Shapley, R. M. Comparisons of the Dynamics of Local Field Potential and Multiunit Activity Signals in Macaque Visual Cortex. en. *Journal of Neuroscience* **30,** 13739–13749. ISSN: 0270-6474, 1529-2401. doi:10.1523/JNEUROSCI.0743-10.2010. (2025) (Oct. 2010).
8. Choi, Y.-S., Koenig, M. A., Jia, X. & Thakor, N. V. Quantifying Time-Varying Multiunit Neural Activity Using Entropy-Based Measures. *IEEE transactions on bio-medical engineering* **57,** 10.1109/TBME.2010.2049266. ISSN: 0018-9294. doi:10.1109/TBME.2010.2049266. (2025) (Nov. 2010).
9. Pachitariu, M., Sridhar, S., Pennington, J. & Stringer, C. Spike sorting with Kilosort4. en. *Nature Methods* **21,** 914–921. ISSN: 1548-7105. doi:10.1038/s41592-024-02232-7. (2026) (May 2024).
10. *Nerve Tissue | SEER Training* Accessed on 08/03/2026. https://training.seer.cancer.gov/anatomy/nervous/tissue.html.

11. Peelle, J. E. *Human Auditory Pathway* Apr. 2025. osf.io/u2gxc.

12. Jones, E. G. in *Dynamic Aspects of Neocortical Function* (eds Edelman, G. M., Gall, W. E. & Cowan, W. M.) (John Wiley & Sons, New York, 1984).

13. Ceballo, S., Piwkowska, Z., Bourg, J., Daret, A. & Bathellier, B. Targeted Cortical Manipulation of Auditory Perception. eng. *Neuron* **104,** 1168–1179.e5. ISSN: 1097-4199. doi:10.1016/j.neuron.2019.09.043 (Dec. 2019).

14. Zeng, F.-G. Celebrating the one millionth cochlear implant. eng. *JASA express letters* **2,** 077201. ISSN: 2691-1191. doi:10.1121/10.0012825 (July 2022).

15. Trouillet, A. *et al.* High-resolution prosthetic hearing with a soft auditory brainstem implant in macaques. en. *Nature Biomedical Engineering* **9,** 1403–1417. ISSN: 2157-846X. doi:10.1038/s41551-025-01378-9. (2026) (Sept. 2025).

16. Penfield, W. & Perot, P. THE BRAIN'S RECORD OF AUDITORY AND VISUAL EXPERIENCE. A FINAL SUMMARY AND DISCUSSION. eng. *Brain: A Journal of Neurology* **86,** 595–696. ISSN: 0006-8950. doi:10.1093/brain/86.4.595 (Dec. 1963).

17. Harris, K. D. *et al.* How do neurons work together? Lessons from auditory cortex. *Hearing Research. Auditory Cortex: Current Concepts in Human and Animal Research* **271,** 37–53. ISSN: 0378-5955. doi:10.1016/j.heares.2010.06.006. (2026) (Jan. 2011).

18. Miyashita, Y. Operating principles of the cerebral cortex as a six-layered network in primates: beyond the classic canonical circuit model. *Proceedings of the Japan Academy, Series B* **98,** 93–111. doi:10.2183/pjab.98.007 (2022).

19. Madisen, L. *et al.* A toolbox of Cre-dependent optogenetic transgenic mice for light-induced activation and silencing. en. *Nature Neuroscience* **15,** 793–802. ISSN: 1546-1726. doi:10.1038/nn.3078. (2026) (May 2012).

20. Humphries, C., Liebenthal, E. & Binder, J. R. Tonotopic Organization of Human Auditory Cortex. *NeuroImage* **50,** 1202–1211. doi:10.1016/j.neuroimage.2010.01.046 (2010).

21. Lamothe, C., Bagur, S., Gosselin, E. & Bathellier, B. Sound Offset Responses Become Highly Informative in the Auditory Cortex. *The Journal of Physiology*. doi:10.1113/JP288416 (2025).

22. Lozano, A. M. *et al.* Deep brain stimulation: current challenges and future directions. en. *Nature Reviews Neurology* **15,** 148–160. ISSN: 1759-4766. doi:10.1038/s41582-018-0128-2. (2026) (Mar. 2019).

23. Maynard, E. M., Nordhausen, C. T. & Normann, R. A. The Utah intracortical Electrode Array: a recording structure for potential brain-computer interfaces. eng. *Electroencephalography and Clinical Neurophysiology* **102,** 228–239. ISSN: 0013-4694. doi:10.1016/s0013-4694(96)95176-0 (Mar. 1997).

24. Lozano, A. *et al.* Large-scale mapping of artificial perceptions for neuroprostheses using spontaneous neuronal activity in macaque and human visual cortex. *Brain Stimulation* **19,** 103019. ISSN: 1935-861X. doi:10.1016/j.brs.2025.103019. (2026) (Jan. 2026).

25. Fernández, E. *et al.* Visual percepts evoked with an intracortical 96-channel microelectrode array inserted in human occipital cortex. en. *The Journal of Clinical Investigation* **131.** ISSN: 0021-9738. doi:10.1172/JCI151331. (2026) (Dec. 2021).

26. Greenspon, C. M. *et al.* Evoking stable and precise tactile sensations via multi-electrode intracortical microstimulation of the somatosensory cortex. en. *Nature Biomedical Engineering* **9,** 935–951. ISSN: 2157-846X. doi:10.1038/s41551-024-01299-z. (2026) (June 2025).

27. *Blackrock Neurotech | Empowered by Thought* en. Accessed on 16/02/2026. https://blackrockneurotech.com/.

28. Khodagholy, D. *et al.* NeuroGrid: recording action potentials from the surface of the brain. en. *Nature Neuroscience* **18,** 310–315. ISSN: 1546-1726. doi:10.1038/nn.3905. (2026) (Feb. 2015).

29. Lu, Y., Lyu, H., Richardson, A. G., Lucas, T. H. & Kuzum, D. Flexible Neural Electrode Array Based-on Porous Graphene for Cortical Microstimulation and Sensing. en. *Scientific Reports* **6,** 33526. ISSN: 2045-2322. doi:10.1038/srep33526. (2026) (Sept. 2016).

30. Bianco, M. G. *et al.* Neural Modulation of the Primary Auditory Cortex by Intracortical Microstimulation with a Bio-Inspired Electronic System. eng. *Bioengineering* **7,** 23. ISSN: 2306-5354. doi:10.3390/bioengineering7010023 (Mar. 2020).

31. Merrill, D. R., Bikson, M. & Jefferys, J. G. R. Electrical stimulation of excitable tissue: design of efficacious and safe protocols. eng. *Journal of Neuroscience Methods* **141,** 171–198. ISSN: 0165-0270. doi:10.1016/j.jneumeth.2004.10.020 (Feb. 2005).

32. Deisseroth, K. Optogenetics. eng. *Nature Methods* **8,** 26–29. ISSN: 1548-7105. doi:10.1038/nmeth.f.324 (Jan. 2011).

33. Sahel, J.-A. *et al.* Partial recovery of visual function in a blind patient after optogenetic therapy. *Nature medicine* **27,** 1223–1229. (2026) (2021).

34. Zhang, F. *et al.* The Microbial Opsin Family of Optogenetic Tools. *Cell* **147,** 1446–1457. doi:10.1016/j.cell.2011.12.004 (2011).

35. Yizhar, O., Fenno, L. E., Davidson, T. J., Mogri, M. & Deisseroth, K. Optogenetics in Neural Systems. *Neuron* **71,** 9–34. doi:10.1016/j.neuron.2011.06.004 (2011).

36. Emiliani, V. *et al.* Optogenetics for Light Control of Biological Systems. *Nature Reviews Methods Primers* **2,** 55. doi:10.1038/s43586-022-00136-4 (2022).

37. Gorski, J. A. *et al.* Cortical Excitatory Neurons and Glia, But Not GABAergic Neurons, Are Produced in the Emx1-Expressing Lineage. en. *Journal of Neuroscience* **22,** 6309–6314. ISSN: 0270-6474, 1529-2401. doi:10.1523/JNEUROSCI.22-15-06309.2002. (2026) (Aug. 2002).

38. Marshel, J. H. *et al.* Cortical layer–specific critical dynamics triggering perception. *Science* **365,** eaaw5202. doi:10.1126/science.aaw5202. (2026) (Aug. 2019).

39. Rajalingham, R. *et al.* Chronically implantable LED arrays for behavioral optogenetics in primates. en. *Nature Methods* **18,** 1112–1116. ISSN: 1548-7105. doi:10.1038/s41592-021-01238-9. (2026) (Sept. 2021).

40. Keppeler, D. *et al.* Multichannel optogenetic stimulation of the auditory pathway using microfabricated LED cochlear implants in rodents. eng. *Science Translational Medicine* **12,** eabb8086. ISSN: 1946-6242. doi:10.1126/scitranslmed.abb8086 (July 2020).

41. Pollmann, E. H. *et al.* A subdural CMOS optical device for bidirectional neural interfacing. en. *Nature Electronics* **7,** 829–841. ISSN: 2520-1131. doi:10.1038/s41928-024-01209-w. (2026) (Sept. 2024).

42. Park, S. I. *et al.* Soft, stretchable, fully implantable miniaturized optoelectronic systems for wireless optogenetics. eng. *Nature Biotechnology* **33,** 1280–1286. ISSN: 1546-1696. doi:10.1038/nbt.3415 (Dec. 2015).

43. Jiang, S., Wu, X., Rommelfanger, N. J., Ou, Z. & Hong, G. Shedding light on neurons: optical approaches for neuromodulation. en. *National Science Review* **9.** ISSN: 2095-5138. doi:10.1093/nsr/nwac007. (2026) (Nov. 2022).

44. Lin, J. Y. A user's guide to channelrhodopsin variants: features, limitations and future developments. en. *Experimental Physiology* **96,** 19–25. ISSN: 1469-445X. doi:10.1113/expphysiol.2009.051961. (2026) (2011).

45. Gong, X. *et al.* An Ultra-Sensitive Step-Function Opsin for Minimally Invasive Optogenetic Stimulation in Mice and Macaques. *Neuron* **107,** 38–51.e8. ISSN: 0896-6273. doi:10.1016/j.neuron.2020.03.032. (2026) (July 2020).

46. Owen, S. F., Liu, M. H. & Kreitzer, A. C. Thermal constraints on in vivo optogenetic manipulations. en. *Nature Neuroscience* **22,** 1061–1065. ISSN: 1546-1726. doi:10.1038/s41593-019-0422-3. (2026) (July 2019).

47. McAlinden, N. *et al.* In vivo optogenetics using a Utah Optrode Array with enhanced light output and spatial selectivity. en. *Journal of Neural Engineering* **21,** 046051. ISSN: 1741-2552. doi:10.1088/1741-2552/ad69c3. (2026) (Aug. 2024).

48. Holonyak Jr., N. & Bevacqua, S. F. COHERENT (VISIBLE) LIGHT EMISSION FROM Ga(As1xPx) JUNCTIONS. *Applied Physics Letters* **1,** 82–83. ISSN: 0003-6951. doi:10.1063/1.1753706. (2026) (Dec. 1962).

49. Feezell, D. & Nakamura, S. Invention, development, and status of the blue light-emitting diode, the enabler of solid-state lighting. *Comptes Rendus Physique. LEDs: The new revolution in lighting / Les LED : la nouvelle révolution de l'éclairage* **19,** 113–133. ISSN: 1631-0705. doi:10.1016/j.crhy.2017.12.001. (2026) (Mar. 2018).

50. Schubert, E. F., Cho, J. & Kim, J. K. *Light-Emitting Diodes* en. ISBN: 978-0-471-23896-6. doi:10.1002/0471238961.1209070811091908.a01.pub3. (2026) (John Wiley & Sons, Ltd, 2015).

51. Lin, J. Y. & Jiang, H. X. Development of microLED. *Applied Physics Letters* **116,** 100502. ISSN: 0003-6951. doi:10.1063/1.5145201 (Mar. 2020).

52. Guo, F. *et al.* The fabrication of color-tunable organic light-emitting diode displays via solution processing. en. *Light: Science & Applications* **6,** e17094–e17094. ISSN: 2047-7538. doi:10.1038/lsa.2017.94. (2025) (Nov. 2017).

53. Kumar, V. & Kymissis, I. MicroLED/LED electro-optical integration techniques for non-display applications. *APL Photonics* **8,** 041101. doi:10.1063/5.0125103 (2023).

54. Taal, A. J. *et al.* Optogenetic stimulation probes with single-neuron resolution based on organic LEDs monolithically integrated on CMOS. en. *Nature Electronics* **6,** 669–679. ISSN: 2520-1131. doi:10.1038/s41928-023-01013-y. (2026) (Sept. 2023).

55. Rae, K. *et al.* Transfer-printed micro-LED and polymer-based transceiver for visible light communications. EN. *Optics Express* **26,** 31474–31483. ISSN: 1094-4087. doi:10.1364/OE.26.031474. (2025) (Nov. 2018).

56. Sekiguchi, H. *et al.* Adhesionable flexible GaN-based microLED array film to brain surface for in vivo optogenetic stimulation. en. *Applied Physics Express* **15,** 046501. ISSN: 1882-0786. doi:10.35848/1882-0786/ac5ba3. (2026) (Mar. 2022).

57. Gu, W., Wang, L., Wang, X., Zhao, C. & Guan, S. Large-Scale, High-Density MicroLED Array-Based Optogenetic Device for Neural Stimulation and Recording. *Nano Letters* **24,** 15252–15259. ISSN: 1530-6984. doi:10.1021/acs.nanolett.4c03645. (2026) (Dec. 2024).

58. McAlinden, N., Gu, E., Dawson, M. D., Sakata, S. & Mathieson, K. Optogenetic activation of neocortical neurons in vivo with a sapphire-based micro-scale LED probe. English. *Frontiers in Neural Circuits* **9.** ISSN: 1662-5110. doi:10.3389/fncir.2015.00025. (2026) (May 2015).

59. Choi, H., Jeon, C., Liu, C. & Dawson, M. *GaN micro-LED arrays with integrated sapphire microlenses* in *The 16th Annual Meeting of the IEEE Lasers and Electro-Optics Society, 2003. LEOS 2003.* **2** (Oct. 2003), 654–655 vol.2. doi:10.1109/LEOS.2003.1252970. (2025).

60. Greer, R. *et al.* Chronically implantable LED arrays for optogenetic cortical surface stimulation in mice. en. *Nature Communications* **17,** 878. ISSN: 2041-1723. doi:10.1038/s41467-025-68191-5. (2026) (Jan. 2026).

61. Ahn, S.-H., Jeong, J. & Kim, S. J. Emerging Encapsulation Technologies for Long-Term Reliability of Microfabricated Implantable Devices. *Micromachines* **10,** 508. ISSN: 2072-666X. doi:10.3390/mi10080508. (2025) (July 2019).

62. Seo, G., Park, S. & Lee, M. How to calculate the life cycle of high-risk medical devices for patient safety. *Frontiers in Public Health* **10,** 989320. ISSN: 2296-2565. doi:10.3389/fpubh.2022.989320. (2025) (Sept. 2022).

63. Golda-Cepa, M., Engvall, K., Hakkarainen, M. & Kotarba, A. Recent progress on parylene C polymer for biomedical applications: A review. *Progress in Organic Coatings* **140,** 105493. ISSN: 0300-9440. doi:10.1016/j.porgcoat.2019.105493. (2025) (Mar. 2020).

64. Caldwell, R. *et al.* Characterization of Parylene-C degradation mechanisms: *In vitro* reactive accelerated aging model compared to multiyear *in vivo* implantation. *Biomaterials* **232,** 119731. ISSN: 0142-9612. doi:10.1016/j.biomaterials.2019.119731. (2025) (Feb. 2020).

65. Zare, M., Ghomi, E. R., Venkatraman, P. D. & Ramakrishna, S. Silicone-based biomaterials for biomedical applications: Antimicrobial strategies and 3D printing technologies. en. *Journal of Applied Polymer Science* **138,** 50969. ISSN: 1097-4628. doi:10.1002/app.50969. (2025) (2021).

66. Ghelich, P., Nolta, N. F. & Han, M. Unprotected sidewalls of implantable silicon-based neural probes and conformal coating as a solution. en. *npj Materials Degradation* **5,** 1–8. ISSN: 2397-2106. doi:10.1038/s41529-021-00154-9. (2025) (Feb. 2021).

67. Herron, J. *et al.* The convergence of neuromodulation and brain–computer interfaces. en. *Nature Reviews Bioengineering* **2,** 628–630. ISSN: 2731-6092. doi:10.1038/s44222-024-00187-0. (2026) (Aug. 2024).

68. Ansó, J. *et al.* Concurrent stimulation and sensing in bi-directional brain interfaces: a multi-site translational experience. en. *Journal of Neural Engineering* **19,** 026025. ISSN: 1741-2552. doi:10.1088/1741-2552/ac59a3. (2026) (Mar. 2022).

69. Centre, O. B. R. *Bioelectronic implant offers intelligent therapy to treat incontinence | NIHR Oxford Biomedical Research Centre* en-US. Accessed on 02/02/2026. Feb. 2023. https://oxfordbrc.nihr.ac.uk/bioelectronic-implant-offers-intelligent-therapy-to-treat-incontinence/ (2026).

70. Musk, E. & Neuralink. An Integrated Brain-Machine Interface Platform With Thousands of Channels. EN. *Journal of Medical Internet Research* **21,** e16194. doi:10.2196/16194. (2026) (Oct. 2019).

71. Rajangam, S. *et al.* Wireless Cortical Brain-Machine Interface for Whole-Body Navigation in Primates. en. *Scientific Reports* **6,** 22170. ISSN: 2045-2322. doi:10.1038/srep22170. (2026) (Mar. 2016).

72. Chapin, J. K., Moxon, K. A., Markowitz, R. S. & Nicolelis, M. A. L. Real-time control of a robot arm using simultaneously recorded neurons in the motor cortex. en. *Nature Neuroscience* **2,** 664–670. ISSN: 1546-1726. doi:10.1038/10223. (2026) (July 1999).

73. Flesher, S. N. *et al.* A brain-computer interface that evokes tactile sensations improves robotic arm control. *Science* **372,** 831–836. doi:10.1126/science.abd0380. (2026) (May 2021).

74. Abbasi, A. *et al.* Brain-machine interface learning is facilitated by specific patterning of distributed cortical feedback. *Science Advances* **9,** eadh1328. doi:10.1126/sciadv.adh1328. (2026) (Sept. 2023).

75. LeCun, Y., Bengio, Y. & Hinton, G. Deep learning. en. *Nature* **521,** 436–444. ISSN: 1476-4687. doi:10.1038/nature14539. (2026) (May 2015).

76. Ahmadi, N., Constandinou, T. G. & Bouganis, C.-S. Robust and accurate decoding of hand kinematics from entire spiking activity using deep learning. en. *Journal of Neural Engineering* **18,** 026011. ISSN: 1741-2552. doi:10.1088/1741-2552/abde8a. (2026) (Feb. 2021).

77. Yamins, D. L. K. & DiCarlo, J. J. Using goal-driven deep learning models to understand sensory cortex. en. *Nature Neuroscience* **19,** 356–365. ISSN: 1546-1726. doi:10.1038/nn.4244. (2026) (Mar. 2016).

78. Zador, A. *et al.* Catalyzing next-generation Artificial Intelligence through NeuroAI. en. *Nature Communications* **14,** 1597. ISSN: 2041-1723. doi:10.1038/s41467-023-37180-x. (2026) (Mar. 2023).

79. Liu, X. & Richardson, A. G. Edge deep learning for neural implants: a case study of seizure detection and prediction. en. *Journal of Neural Engineering* **18,** 046034. ISSN: 1741-2552. doi:10.1088/1741-2552/abf473. (2026) (Apr. 2021).

80. Xu, J. *Comparison and Application of FPGA and ASIC in Digital System Design* in *Proceedings of the International Conference on Decision Science & Management* (Apr. 2024), 216–220. ISBN: 979-8-4007-1815-1. doi:10.1145/3686081.3686117. (2026).

81. *AMD Zynq™ UltraScale+™ MPSoC ZCU104 Evaluation Kit* en. Accessed on 08/02/2026. https://www.amd.com/en/products/adaptive-socs-and-fpgas/evaluation-boards/zcu104.html.

82. Thomas, S. Neural processors for the edge. en. *Nature Electronics* **4,** 172–172. ISSN: 2520-1131. doi:10.1038/s41928-021-00563-3. (2026) (Mar. 2021).

Chapter 3

µLED array systems for chronic *in-vivo* experiments

This chapter details the design of optogenetic cortical surface stimulators based on a 100-pixel, matrix-addressable µLED array that has been custom-fabricated within the group (Emma Butt & Yunzhou Cheng; see Acknowledgements). The design and fabrication of the gallium nitride (GaN)-on-sapphire µLED arrays was detailed in Chapter 2. The main result of this chapter was the successful integration of µLED arrays into implantable systems to enable *in-vivo* experiments in the mouse cortex. Two distinct packaging methodologies have been developed. Firstly, the "acute system" which is used for precise placement of the µLED array onto a cortical window over the cortical surface within a compact craniotomy, allowing for simultaneous electrophysiology recordings to be carried out within an acute preparation. Secondly, the "chronic system", where the packaging is partially flexible, allowing the µLED array to be implanted chronically in mice, in direct contact with the cortical tissue. These were used to carry out behavioural experiments in mice over several days/weeks. Device encapsulation strategies for chronic *in-vivo* stability are evaluated through accelerated ageing testing. This chapter concludes with the demonstration of a device incorporating both a 100-pixel µLED array and 32-channel µECoG (electrocorticography) array (Amélie Albon and Charles Rezaei-Mazinani; see Acknowledgements). Further characterisation of the optical and thermal considerations of implanting and operating µLEDs on the cortical surface for optogenetics, are discussed in Chapter 4, and the results of both sets of *in-vivo* experiments using these devices are detailed in Chapter 5. The key results of this chapter including the acute and chronic device design, and accelerated testing of the devices, have been published in [1].

3.1 µLED array operation and interconnection

A photograph of the µLED array devices on a 2" wafer is shown in Fig. 3.1a and individual chips after dicing shown in Fig. 3.1b and 3.1c.

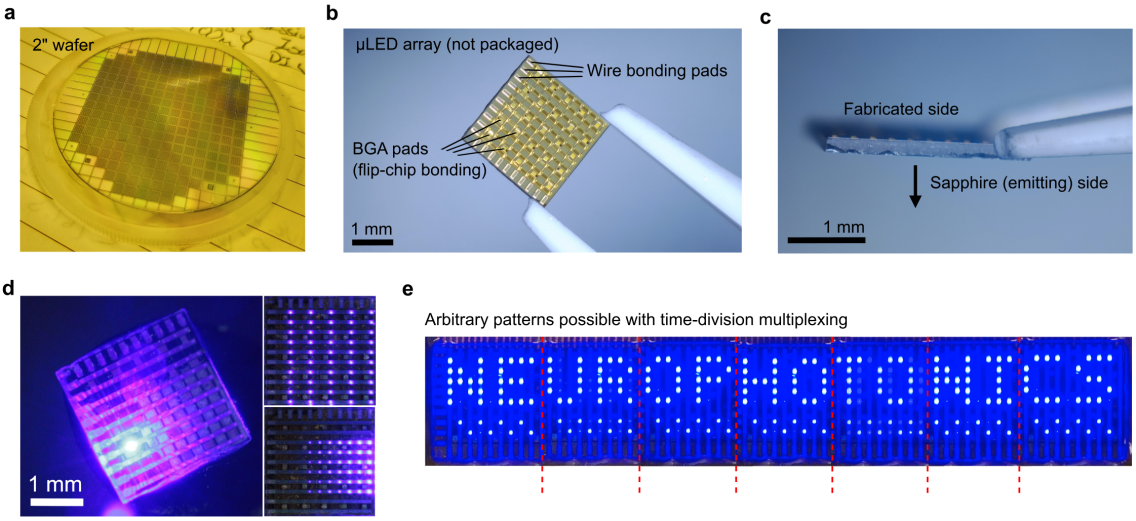


Figure 3.1: **Summary of µLED arrays. (a)** Photograph of 2" wafer with 220 µLED arrays. **(b)** Individual µLED array (fabricated side). **(c)** Individual µLED array (side-view). **(d)** Sapphire (emitting) side of µLED array show individual pixel and patterned illumination. **(e)** Example of arbitrary pattern generation using time-division multiplexing (several images overlapped; vertical dashed lines indicate overlap).

The key features of the µLED array are listed below:

- 40 µm square pixels arranged in a 10x10 matrix-addressable array
- 200 µm pixel spacing
- 2.2 x 2.2 mm^2 x 150 µm form factor
- ~450 nm emission with FWHM of ~20 nm

Fig. 3.2 explains how to address the µLEDs through the bond pads and ball-grid array (BGA) as well as how time-division multiplexing (TDM) can be used to generate arbitrary patterns. Fig. 3.2a also shows the numbering scheme which has been used to label the tracks connected to the anodes and cathodes; here we refer to these as p and n tracks, not to be confused with the p-type and n-type GaN material that the µLEDs are fabricated from. We ID individual µLEDs throughout this report based on this numbering scheme; i.e. in Fig. 3.2a the illuminated µLED on p track 5 and n track 2 is P5N2. Note that this is "looking" from the fabricated side and not the sapphire side (where numbering would have to be mirrored so the

µLEDs maintain consistent IDs). To make this even clearer with an example, the µLED located in the corner of the bond pads is always P10N1 regardless of the orientation. Care has been taken throughout this project to maintain this consistent numbering scheme.

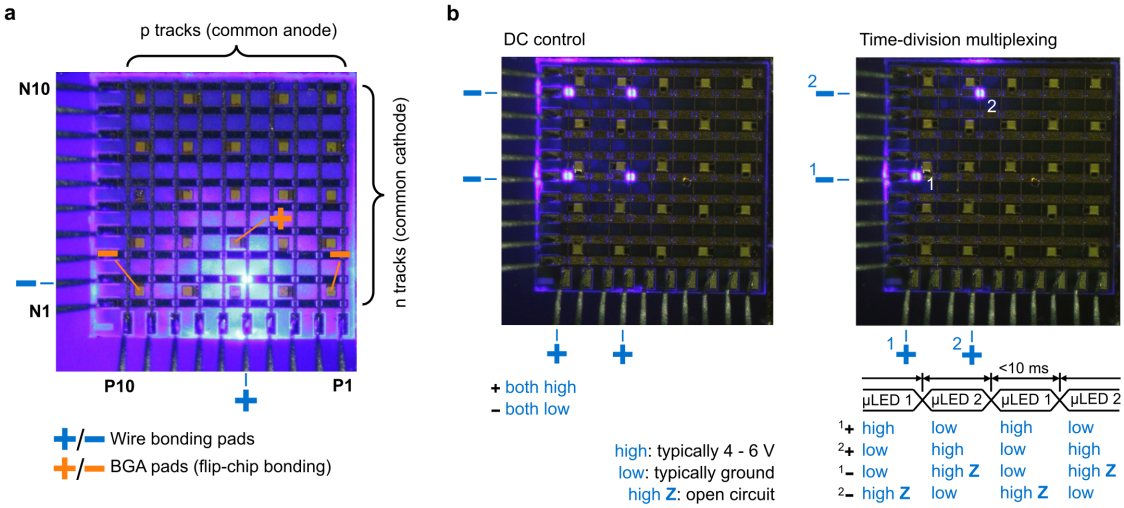


Figure 3.2: **Illustration of electronic control of µLED array. (a)** Fabricated side of µLED array with addressing of illuminated pixel through either wire bonding or BGA pads. **(b)** Illustration of direct current (DC) control (left) and time-division multiplexing control (right).

The devices are operated using a custom electronic driver system to implement either mode of operation. This system is detailed fully in Chapter 6.

3.2 Integration and packaging of µLED arrays

3.2.1 Rigid implants for acute experimentation

The primary output of this chapter has been the development and evaluation of flexible device packaging to enable chronic *in-vivo* experimentation in the auditory cortex of the mouse model. This section details the initial device - termed the "acute device" - which enabled simultaneous electrophysiology recordings to be carried out from the auditory cortex of awake, head-fixed mice under optogenetic stimulation from the µLED array. Fig. 3.3b shows a schematic of the experiment where the µLED array was placed into a compact craniotomy (∼5 mm diameter) over a glass cortical window. This experimental procedure is standard in optogenetic and imaging studies to provide an optical interface while protecting the brain [2].

This experiment, and the subsequent analyses of the electrophysiological data are described in more detail in Chapter 5. For this section, we will focus on the geometric constraints of the experiment on the device packaging. The general requirement of this was to have a simple package neuroscientists could use to lower the µLED array precisely onto the cortical window

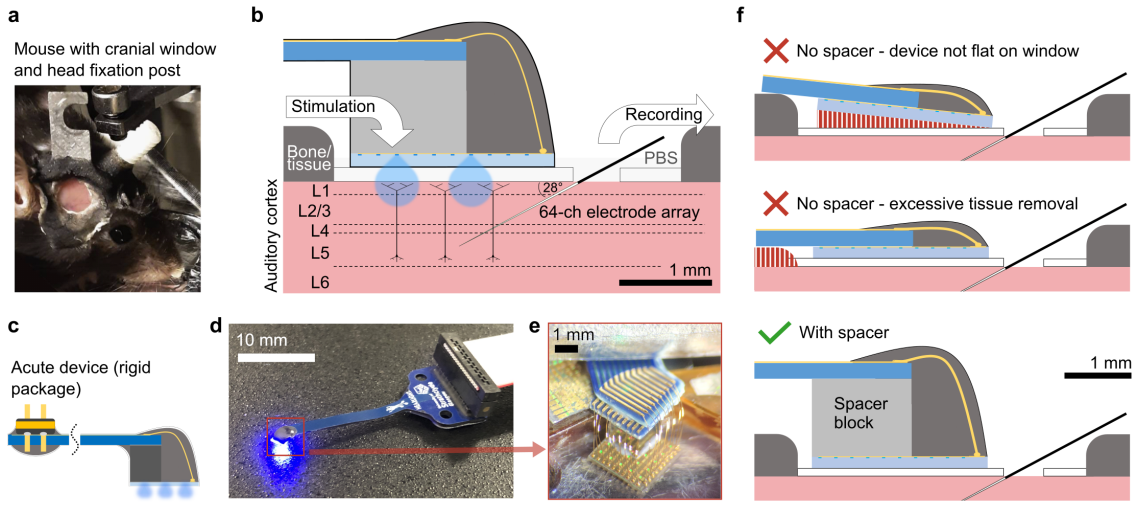


Figure 3.3: **Summary of head-fixed *in-vivo* electrophysiology experiments and acute device packaging. (a)** Photograph of Emx-IRES-Cre x Ai27d mouse [3, 4] with cranial window implanted over right auditory cortex and head-fixation post; in this image, the mouse also has a medial forebrain bundle (MFB) stimulator electrode implanted (white connector) which was not used for electrophysiology experiments. **(b)** Schematic summary of electrophysiology experiment with µLED array placed on cranial window, utilising the rigid package. **(c)** Cross-sectional schematic of acute device. **(d)** Photograph of acute device with 20-pin ribbon cable used for µLED addressing. **(e)** Zoom of device tip showing gold ball-wedge wire bonds connecting µLED array to PCB system. **(f)** Schematic illustration motivating the use of the spacer block in the acute device packaging.

using a micro-manipulator, with enough space to insert an electrophysiology recording probe. Another requirement was that all 100 µLEDs should be addressable using a simple connector scheme. Furthermore, the packaging should be robust and resistant to biological conditions. The final packaged acute device is shown in Fig. 3.3d. Prior to packaging, µLED arrays were fully characterised to ensure high quality devices (see Chapter 4 for details of device characterisation).

To rapidly prototype this packaging, commercial printed circuit board (PCB) manufacturers (PCBway) were utilised. Low-cost PCBs can be manufactured with track widths and spacings as small as 60-100 µm. Furthermore, they can be plated with electroless nickel immersion gold (ENIG) facilitating easier interconnection through wire bonding or flip-chip bonding. To achieve the delicate placement of components to assemble the acute devices, a flip-chip bonder was used (Finetech FINEPLACER Pico ma). This machine consists of a vacuum-enabled head, capable of picking up very small components, of a few hundred microns in dimension. The component is optically aligned over the pads for bonding, and then lowered onto the substrate. The heating element is then engaged (hotplate and/or heated pick-up head) to melt solder, or to cure epoxy in this case. For the acute device, the flip-chip bonder was only used to place components and cure epoxy, although the same machine has been used

for thermosonic gold-bump flip-chip bonding to make the interconnection denser and more compact for chronically implantable devices, as explained in the next section.

Table 3.1 summaries the specific process steps for the interconnection of the µLED array to the PCB packaging.

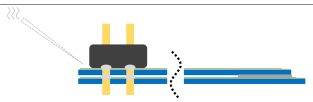
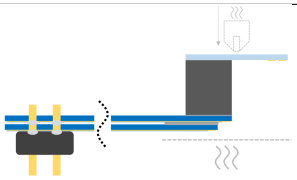
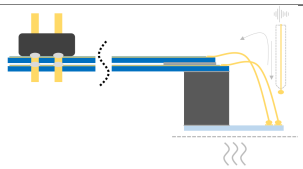
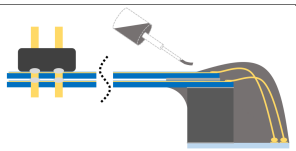
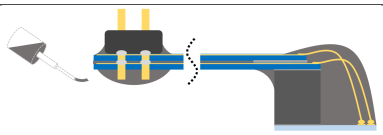
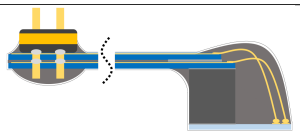
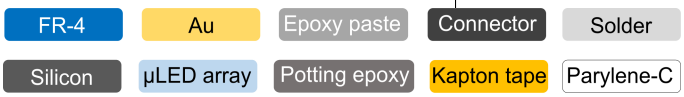
Process step description	Cross-sectional schematic
Step i: Gluing PCBs together • Material: MG Chemicals 9460TC	
Step ii: Soldering connector • Material: lead-free wire solder	
Step iii: Gluing spacer block and µLED array to PCB assembly • Material: MG Chemicals 9460TC	
Step iv: Wire bonding • Bonding parameters: first bond (power: 3.5, time: 5.5, force: 3); second bond (power: 3, time: 6, force: 3); temperature (150 °C)	
Step v: Potting wire bonds • Material: MG Chemicals 9510	
Step vi: Potting connector and PCB shanks • Material: MG Chemicals 9510	
Step vii: Parylene-C encapsulation	
	

Table 3.1: **Summary of process steps for acute device packaging.**

The acute device packaging consists of two 200 µm thin FR-4 PCBs, each with a set of ten 100 µm wide tracks which run ∼3 cm from a connector to an exposed end for wire bonding to (Fig. 3.4a). The dual-PCB design was chosen as the packaging needed to be miniaturised as much as possible for acute *in-vivo* experiments, and the PCB manufacturer has limited capabilities. The PCBs were glued together and a 1.25 x 1.25 x 1 mm^3 "spacer" block was glued to the end using an electrically insulating, thermally conducting epoxy adhesive paste (MG Chemicals 9460TC) (Fig. 3.4b). The µLED array was then glued to the spacer block using the same epoxy. The purpose of the spacer block was to provide a stand-off so the µLED array could be placed into the cranial recess without having to remove excess bone and tissue to permit space for the PCB. The first iteration of this packaging did not include the spacer block, resulting in major challenges during the first phase of *in-vivo* experiments (Fig. 3.3f). Initially silicon was chosen for the spacer block material as wafers were readily available, easy to machine into small blocks, and thermally-conductive which is helpful for wire bonding. However, on occasion if the top-most encapsulation layer on the fabricated side of the µLED array was flawed, the metal tracks could short onto the spacer block; going forward an electrically-insulating material should be used to mitigate this.

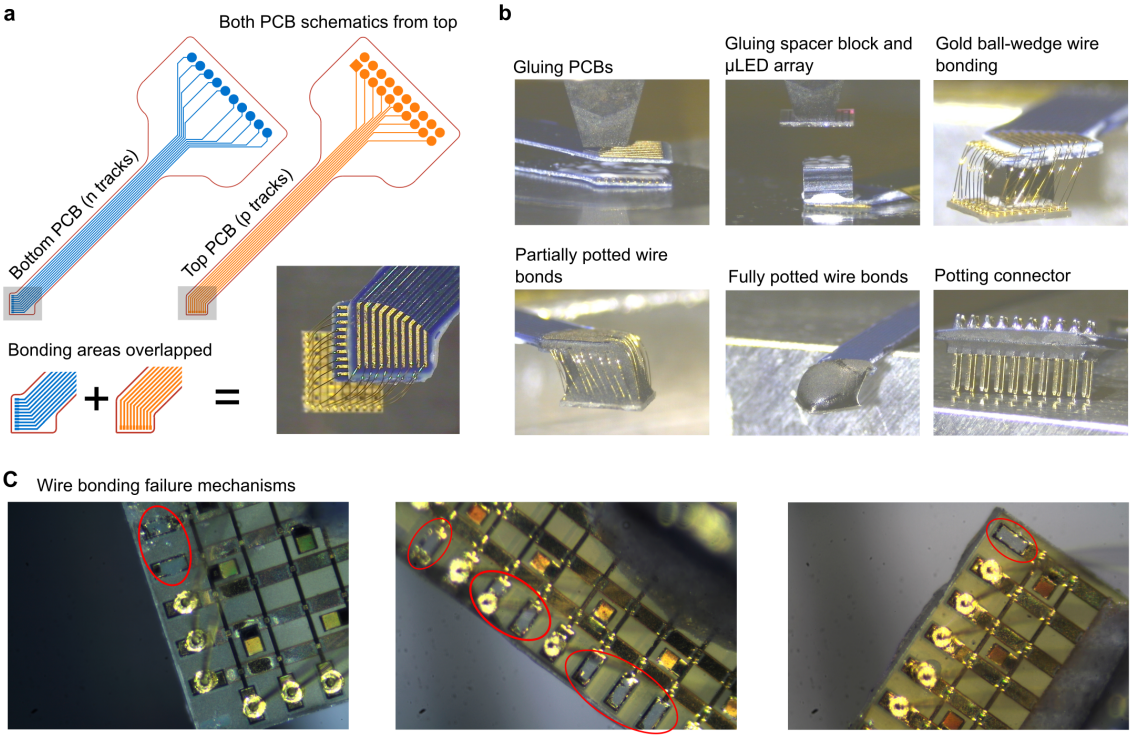


Figure 3.4: **Assembly process of acute device. (a)** PCB layout for acute device. **(b)** Images of assembly process for acute device. **(c)** Images of wire bonding failure mechanisms showing bond pad removal.

Wire bonding is a technique usually used for the interconnection of bare die onto a substrate.

Typically, the wire is made from aluminium or gold with wire thicknesses as low as 10s of microns. Wire bonder machines utilise ultrasonic energy to form a bond between the wire and pad, or substrate; in the case of gold-wire bonding thermal energy is also employed to enhance the quality of the bond. These machines are often operated manually by a skilled operator [5]. In this project, for the acute device, gold ball-wedge wire bonding was employed for the interconnection of the µLED array to the PCB packaging. Prior to wire bonding, the PCBs were cleaned in an ultrasonic bath in acetone (10 minutes), isopropyl alcohol (IPA) (5 minutes) and dried with compressed nitrogen (N_2), to ensure a clean bonding surface. Using a manual wire bonder (MPP iBond5000), gold ball-wedge wire bonding (25 µm thick wire) was used to connect the p and n bond pads on the µLED array to the two rows of exposed 75 nm thick ENIG (electroless nickel immersion gold) electrical tracks on the PCBs (Fig. 3.4a, 3.4b). A custom chuck was designed to hold the assembly in place during wire bonding. A 75 µm diameter gold ball was deposited on the µLED array bond pad and the second wedge bond placed on the PCB track, after setting a high loop height to overcome the 1 mm step. This process was challenging due to the geometry of the assembly, and density of the wires; occasionally tweezers were used to manually manipulate the wires to avoid shorts. A programmable adhesive dispenser (Metcal DX-350) was used to pot the wire bonds using a potting compound (MG Chemicals 9510) to both mechanically and electrically protect them. The final assembly steps for the acute device packaging consisted of soldering a 20-pin rectangular connector (1.27 mm pitch) through both PCBs, so a single connector/cable could address all the µLEDs; the connector area was also potted. Finally, to make the devices resistant to conductive solution, they were encapsulated with a 7 µm layer of parylene-C using a vapour deposition tool (Specialty Coating Systems PDS2010 Labcoater). To prevent parylene-C inadvertently insulating the connector pins during deposition, a cable was plugged in and the gap tightly sealed with Kapton tape, which was carefully removed with a scalpel later.

Major challenges were encountered in the packaging of the acute devices, particularly with the wire bonding. The gold pads on the µLED array would peel off a small percentage of the time; usually 1/10 - 1/20 bonds would fail despite a sweep of bonding parameters being assessed. These could not be recovered, therefore a single failed bond rendered a row or column of µLEDs non-functioning. These random failures hindered assembly time, with only a single device with 100% interconnection achieved out of 17 attempted devices. Larger, thicker or better-adhered gold pads should be considered for devices requiring interconnection of this nature. Nevertheless, once assembled, the acute devices are extremely robust and are able to withstand physical stresses such as bending and dropping. The device used for all electrophysiology experiments (∼10-15 hours of active use in solution) suffered from minor degradation in optical performance which is discussed in more detail in Chapter 5. However, the packaging itself did not deteriorate, with all wire bonds remaining connected. The miniaturisation efforts

employed in the design of the packaging ensured minimal additional surgery and distress to the animals. The acute devices enabled successful electrophysiology recordings to be carried out from the auditory cortex of awake, head-fixed mice under simultaneous stimulation from the µLED array, allowing irradiance thresholds and spatial resolution, and therefore an operational framework for these devices to be established (full detail of experiments is provided in Chapter 5).

3.2.2 Flexible implants for chronic implantation

The chronically implantable device (chronic device) packaging is based on a PCB made from polyimide; a flexible, biocompatible and mechanically robust polymer [6]. The PCB is 100 µm thin, with twenty 60 µm wide tracks which run from a connector site (Omnetics Nano circular) to exposed bump bonding pads which are pitch-matched to the BGA layout on the µLED array (Fig. 3.6a). Utilising the BGA pads for flip-chip bonding results in a highly compact interconnection (Fig. 3.5a, 3.5b), important for chronically implantable devices which should be robust, light-weight and minimally-invasive [7]. This section summarises the assembly process for the chronically implantable devices, as well as advanced designs to enable wireless operation or control of multiple µLED arrays simultaneously.

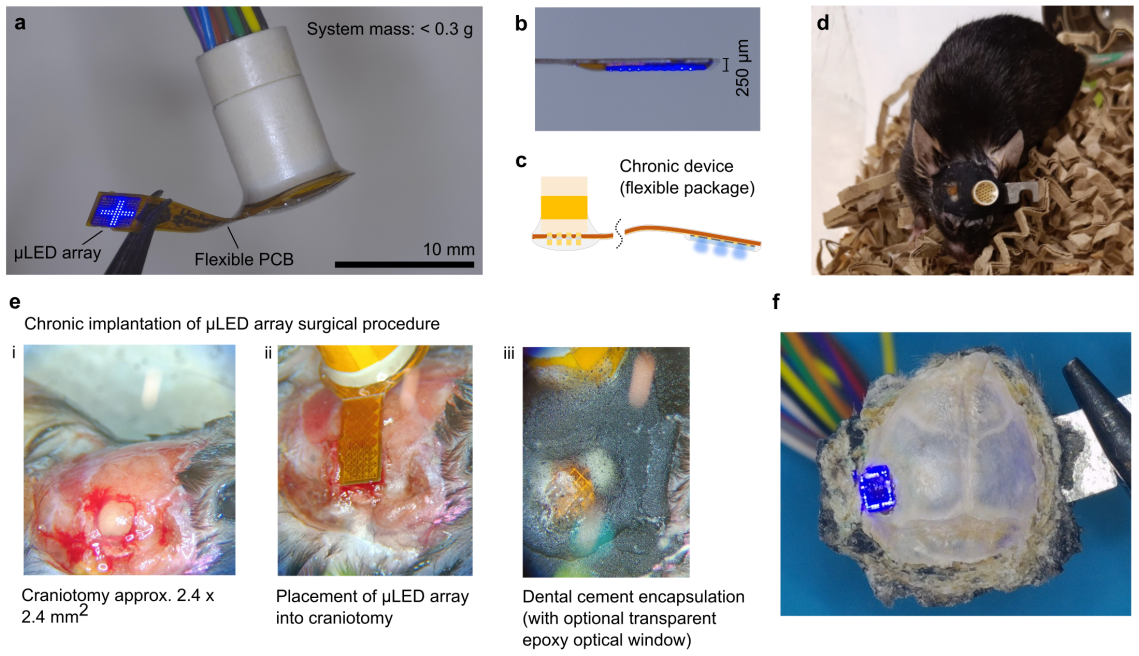


Figure 3.5: **Summary of chronic device packaging and implantation. (a)** Photograph of µLED array, packaged to a flexible system for chronic implantation (chronic device). **(b)** Photograph of side-view of chronic device packaging. **(c)** Cross-sectional schematic of chronic device. **(d)** Photograph of mouse chronically implanted with µLED array and head-fixation post; protruding connector permits addressing of µLED array. **(e)** Surgical steps for chronic implantation. **(f)** Photograph of µLED array embedded within craniotomy after explantation; view from inside the skull. Outside edge illuminated to provide scale bar of array.

Table 3.2 summaries the specific process steps for the flexible device packaging. The PCBs were cleaned using the same process described previously, prior to assembly. The packaging process starts by applying bump bonds to the PCB for the flip-chip bonding interconnection. For bump bonding, the PCB was mounted on a vacuum-enabled aluminium chuck and taped around the edges to mitigate flexing of the PCB during the bump bonding process. A gold stud bump (∼75 µm diameter) was attached to each of the 25 bond pads; the dual-mode wire bonder is operated in the "bumping" mode for this process. Because the gold wire is optimised for wire bonding, a tail extends from the top of each stud/ball which varies in height (Fig. 3.6b left). This should be removed using a coining tool, which is flat and squashes the tail using ultrasonic energy and force (Fig. 3.6b right). For coining, the machine is operated without gold wire and in the "TAB" mode (which stands for "tape-automated bonding"), a process with similar requirements to coining.

The subsequent interconnection of the µLED array to the PCB utilises thermosonic flip-chip bonding and the same flip-chip bonder tool described previously. The "ultrasonic module" replaces the heated "chip-contact module" which was used previously for picking and placing

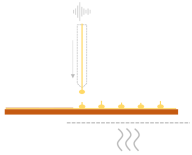
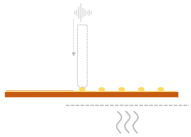
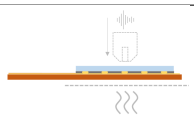
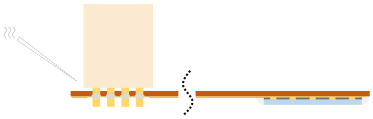
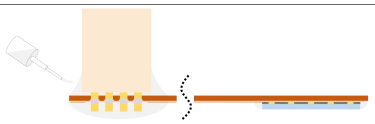
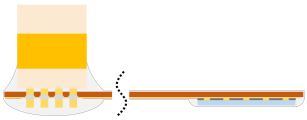
Process step description	Cross-sectional schematic
Step i: Bump bonding • Bonding parameters: bond (power: 3.9, time: 5.5, force: 1); temperature (75 °C)	
Step ii: Bump bond coining • Bonding parameters: bond (power: 3.9, time: 5.5, force: 1); temperature (75 °C)	
Step iii: Pre-underfill • Material: MG Chemicals 9510	
Step iv: Flip-chip bonding • Flip-chip bonding parameters: ultrasonic (power: 1500 mW, time: 500 ms); force (20 N); temperature (75 °C) • Underfill cure: temperature (150 °C)	
Step v: Bevelled underfill • Material: EPO-TEK MED-301-2	
Step vi: Soldering connector • Material: low-temperature solder paste	
Step vii: Potting connector • Material: EPO-TEK MED-301-2	
Step viii: Parylene-C/medical-grade silicone encapsulation • Material: MED-2214 (+MED-161 silicone primer)	
	

Table 3.2: **Summary of process steps for flexible, chronic device packaging.**

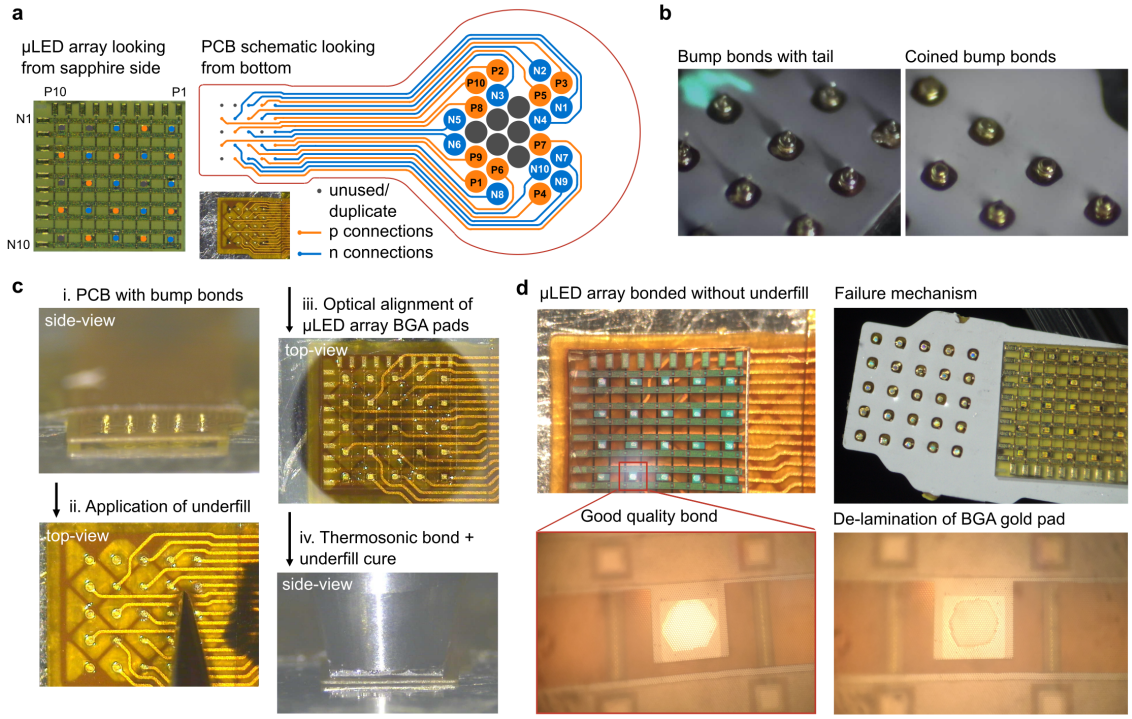


Figure 3.6: **Assembly process for flexible, chronic packaging. (a)** Schematic of interconnection and PCB layout. **(b)** Gold bump bonds before and after coining. **(c)** Flip-chip bonding process steps. **(d)** Illustration of failure mechanism related to BGA pad removal.

components; the hotplate provides the necessary heating. The PCB is taped to an aluminium chuck. The vacuum-enabled chuck was not used due to its thickness being too high for the vertical range of the flip-chip bonder operation; a slimmer, vacuum-enabled chuck would improve the process as the PCB currently does not sit flat due to the flexibility. Prior to the following steps, the simple calibration process must be carried out according to the user manual. The µLED array is then picked up using the ultrasonic pick-up head, and then the BGA pads aligned over the corresponding bond pads on the PCB (Fig. 3.6c). After alignment, the chip would normally be lowered into contact with the gold bumps on the bond pads, and thermosonic energy applied to create the connection. However, a stable mechanical connection was never achieved despite a range of ultrasonic energy and force parameters being assessed. The µLED array would almost always come off the PCB, while the BGA pads peeled off and stayed on top of the bumps (Fig. 3.6d), suggesting the thermosonic bonding was successful but there was an underlying adhesion problem with the gold BGA pads to the substrate of the µLED array. After flip-chip bonding, often an epoxy underfill is applied for mechanical stability and electrical insulation [8]. In our case, applying the underfill would lift the chip off the PCB due to the aforementioned mechanical instability. To work around this issue

and achieve eventual successful interconnection, a "pre-underfill" (MG Chemicals 9510) was applied in between the bond pads on the PCB (Fig. 3.6c bottom left). The flip-chip bonding process was then carried out, with the μLED array lowered onto the bump bonds and ultrasonic energy applied to make the connection. At this stage, the chuck was not lifted up; instead the chuck was left in contact so the chip would not move and the temperature was ramped to the cure point of the epoxy and a full cure of the pre-underfill was done. This meant the interconnection was mechanically stable before any movement of the chip or flexing of the PCB could occur potentially compromising mechanical stability. It is important to dispense an appropriate amount of epoxy; too little will leave voids, potentially resulting in future devices failures, and too much could result in epoxy spilling out the sides and flowing onto the top of the sapphire which could inadvertently bond the chip to the ultrasonic pick-up head. We believe the reason for this instability was due to the gold pads being fabricated directly on the patterned-sapphire substrate, resulting in a failure mechanism where the sapphire "nodules" would break or crack under the ultrasonic energy and force resulting in mechanical instability and eventual interconnection failure (Fig. 3.6d). The solution was to apply a pre-underfill, but an obvious question arises on whether this could electrically insulate the BGA pads from the PCB. After the pre-underfill and flip-chip bonding procedure was optimised, we found no degradation in electrical or optical performance, with all 20 connections being consistently successful and never failing (18 consecutively assembled devices). As discussed in the next section, the interconnection remained stable in solution and after chronic implantation, *in-vivo* experiments and explantation. However, the pre-underfill procedure is challenging for the engineer and introduces inherent variabilities in the underfill quality (e.g. voids). Furthermore, the assembly time is increased requiring an additional $\sim$1 hour underfill cure per device, limiting manufacturing throughput and production scalability. Most recent batches of fabricated μLED arrays are manufactured with the gold BGA pads deposited on a flat and more robust material; the obvious choice being the nGaN, as the photolithography masks could be easily modified to facilitate this. This change has since resolved the discussed issue, resulting in higher throughput device assembly and potentially improved device lifetime due to no voids being present in the new MED-301-2 underfill, which is flowed under the μLED array after flip-chip bonding; although accelerated testing has not been carried out with these new devices.

The remaining assembly steps involve applying a medical-grade epoxy underfill (Epoxy Technology, EPOTEK MED-301-2) around the μLED array to improve encapsulation and biocompatibility, and soldering and potting the connector (Fig. 3.7a). Since these devices were intended for chronic implantation, an encapsulation strategy was devised for long-term stability *in-vivo*, rather than just resistance to solution as with the acute devices. The chronic devices received a thicker coating of parylene-C (15 μm), or two 15 μm coatings for extra

encapsulation. The second coating was intended to cover any defects in the first coating. Prior to parylene-C coating, the devices were cleaned using IPA only in an ultrasonic bath (acetone degrades the medical-grade epoxy) and thoroughly dried with compressed N_2. This cleaning was done in a cleanroom environment to avoid contamination (the parylene coater is also located in the same cleanroom). As before, a cable is plugged into the connector and then tightly sealed with Kapton tape to avoid parylene-C depositing on and inadvertently insulating the connector pins, which would be very difficult to remove due to the small form factor of Omnetics connectors.

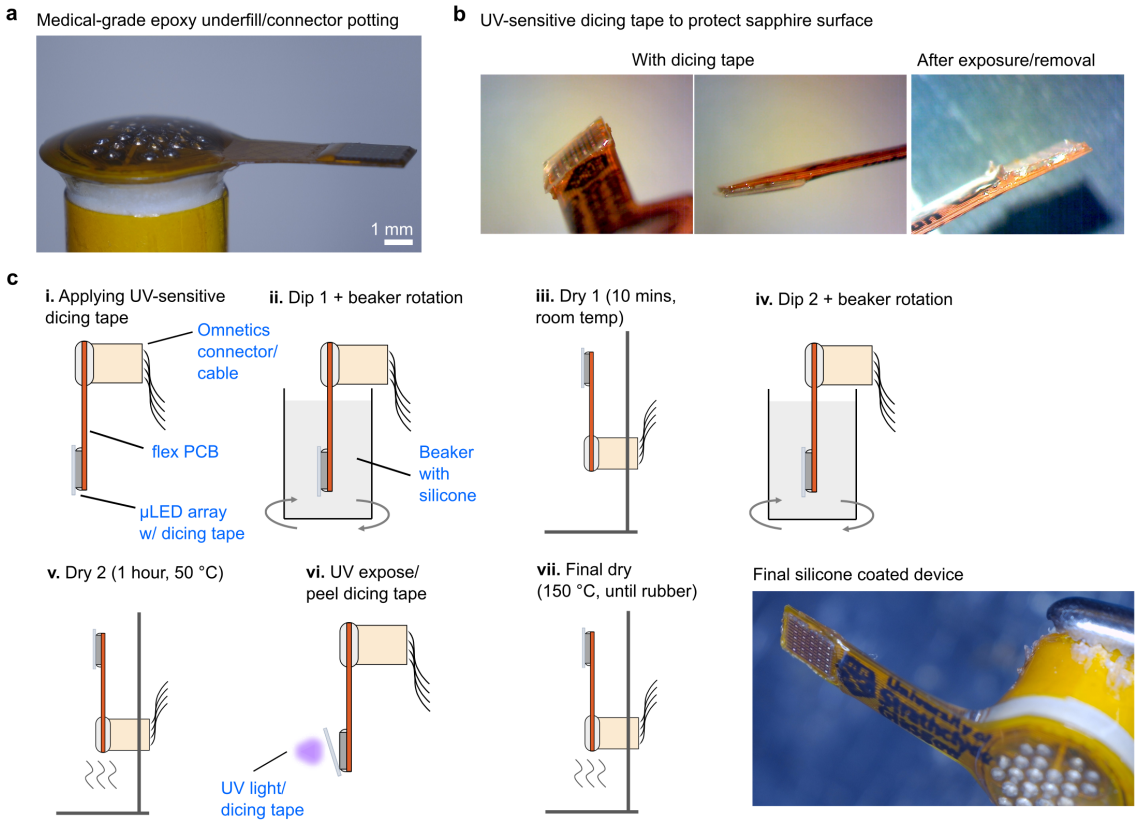


Figure 3.7: **Encapsulation of chronic implants.** **(a)** Photograph showing medical-grade epoxy underfill and potting of Omnetics connector. **(b)** Images showing process of applying small square of UV-sensitive dicing tape to protect sapphire surface during silicone dip coating. **(c)** Full procedure for silicone dip coating of chronic devices.

Parylene-C alone has been shown have limited effectiveness as an encapsulation strategy for chronically implanted devices [9]. Thus, we investigated the use of medical-grade silicone which is biocompatible and already proven in the clinic for implanted devices [10]. We utilised NuSil MED-2214 silicone dispersion which is typically used for dip or spray coating. The dispersion contains the silicone elastomers dispersed in a solvent (Xylene, in this case) and

can result in thin coatings on the order of 100-500 µm. Fig. 3.7c shows the steps involved in applying the silicone dip coating. For our devices, a silicone primer (NuSil MED-161) was firstly applied using a small bristle brush and allowed to dry for one hour. A small piece (3 x 3 mm^2) of UV-sensitive dicing tape was attached to the sapphire surface of the device. Manually, by hand, the device was fully submerged in MED-2214 which had been decanted into a beaker. While submerged, the beaker was rotated slowly several times with the device held at different angles, before carefully lifting the device out and leaving it suspended on a drying rack for 10 minutes at room temperature until the silicone cured to a "yoghurt" consistency. The cure was done with the device tip facing upwards so that excess silicone would drip towards the connector where it was easier to remove without damaging the µLED array. A second coating was applied in the same manner as before and the device suspended on a drying rack, this time above a hotplate set to just over 50 °C, for one hour. The dicing tape was exposed to UV light and peeled off, so the sapphire surface was clear but the rest of the device was encapsulated; this was necessary as the µLEDs would be too far away from the brain surface if there was silicone in front of the sapphire (Fig. 3.7b). A final cure was carried out at 150 °C until the silicone had fully cured. In the following section, we show that devices encapsulated with both parylene-C and medical-grade silicone achieve longer stability *in-vivo* versus parylene-C alone. There are open questions on whether these silicone-coated devices can still be easily implanted in the mouse model as we only implanted devices with a parylene-C coating, except in one case where a silicone-coated device was implanted. The silicone coating makes the devices bulkier and therefore slightly more difficult to implant. Furthermore, there is a standoff between the µLED array and shank which which is lost and therefore may introduce difficulties in placing the device in direct contact with the cortical surface, particularly in the small square-shaped craniotomies we perform (Fig. 3.5e).

3.2.3 Wireless system prototype

Chronic implantation offers the possibility to perform experiments with behaving animals [11, 7]. This is shown with tethered animals in Chapter 5, although here we investigated the integration of the µLED arrays into a fully wireless, battery-powered platform based off a wireless Bluetooth headstage developed within the group (Svetoslav Stoyanov; see Acknowledgements). Fig. 3.8a compares different experimental paradigms highlighting the evolution of the packaging to eventually facilitate freely-behaving experiments.

Both the acute device and chronic device in Fig. 3.8a have been discussed previously. The chronic device has also been translated to behaving experiments (using a flexible, lightweight cable) as discussed in Chapter 5. The schematic showing the chronic device with a wireless headstage (Fig. 3.8a, right) is proposed in this section, with a custom daughter board to interface with the Bluetooth wireless headstage. This system is only a prototype, still requiring

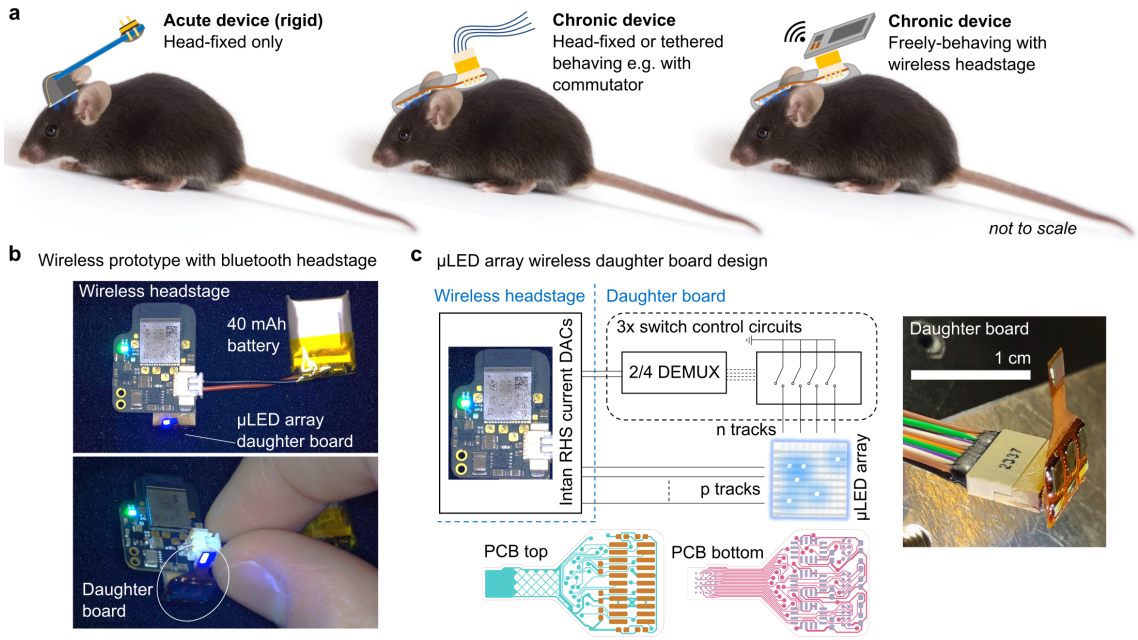


Figure 3.8: **Evolution of implantable µLED arrays towards wireless prototype. (a)** Comparison of acute device (for head-fixed experimentation) and chronic device, which allows behaving experiments through tether or fully wireless control. **(b)** Images of µLED array integrated to custom daughter board for interfacing with Bluetooth wireless headstage; line of µLEDs illuminated to demonstrate wireless power and addressing. **(c)** Design of daughter board and PCB layout (left); photograph of populated daughter board plugged in with cable for testing (right).

optimisations to the electronics and headstage software, and has not been tested *in-vivo*, however the prototype successfully operates as shown in Fig. 3.8b. The focus of this short section is the wireless daughter board design which was carried out as part of the work in this thesis.

The daughter board integrates a single µLED array on a flexible polyimide PCB, as discussed previously. Additional electronics, along with a 22-pin Omnetics connector, were integrated onto the same PCB to allow the wireless headstage to control the µLEDs. The headstage contains a microcontroller operating an Intan RHS 16-channel stimulation/recording ASIC. 10 channels of the RHS were connected to the 10 p tracks. The remaining 6 channels operate the control lines of three 2/4 demultiplexers which each operate a 4-channel switch matrix, connecting the respective n tracks to ground. Miniaturised (flip-chip bonding) component packaging was chosen to reduce the size of the daughter board, however this introduced difficulties with the soldering to the flexible PCB and therefore a fully-operational system was not realised within the time constraints of this thesis. It is expected that a fully-operating system could be realised within a short time-frame utilising this work. Nevertheless, this system demonstrated the possibility of fully-wireless operation of the µLED arrays with an

overall system weight (headstage, battery and daughter board) of less than 3 grams, permitting wireless, freely-behaving experiments in unencumbered mice.

3.2.4 Scalable integration of multiple µLED arrays to flexible package

The flip-chip bonding process described in this section is rapidly scalable such that multiple µLED arrays could be integrated onto a single flexible PCB, allowing scalability across brain regions or implantation in animal models with larger brains, such as non-human primates (NHPs). Fig. 3.9a shows a PCB design for packaging 4 µLED arrays electrically addressed as a 20x20 matrix although they are arranged in a linear fashion separated by 1 mm gaps. Fig. 3.9b shows photographs of the actual device after assembly, incorporating a miniature rectangular Omnetics connector with 36 pins to address the 20x20 matrix. Four pairs of pins are optionally connected together via surface mount resistors on the bottom side of the PCB; this was due to the connector only having 36 pins. The electronic control system described in Chapter 6 enables control of up to 20 common p and 20 common n tracks, therefore making this advanced design easily controllable. If a larger number of µLED arrays were to be integrated to a single PCB, then rather than scaling the size of the connector and cable, it may be more appropriate to integrate switching (multiplexing) electronics onto the PCB as was shown in Fig. 3.9 with the wireless daughter board. Due to the use of commercial polyimide flexible PCBs, it is very easy to produce new designs rapidly and integrate custom electronics as well as an arbitrary number of µLED arrays, potentially extending the technology to larger animal models.

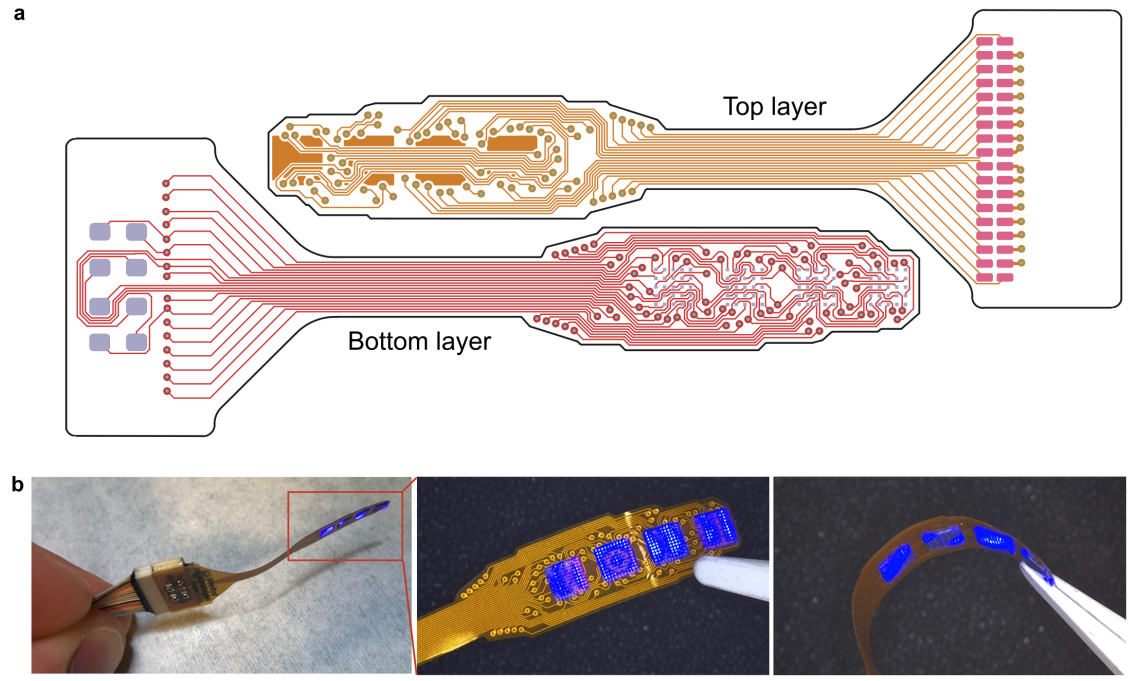


Figure 3.9: **Chronically implantable device integrating 4 µLED arrays. (a)** Schematics of top and bottom layers of PCB design monolithically integrating 4 µLED arrays. **(b)** Photographs of actual device with different patterns of µLEDs independently controlled across each array.

3.3 Accelerated ageing for the evaluation of encapsulation strategies for long-term chronic implantation

Long-term stability of the implanted devices is crucial for the intended chronic *in-vivo* experimentation. Although, it is important to note that there is a field dedicated to lifetime testing of implantable devices [12] and the work carried out here is only intended to give us confidence in the device longevity over the course of *in-vivo* mouse experiments. A dedicated study could investigate specific failure mechanisms providing a more definitive estimate of device lifetime. A comprehensive investigation of encapsulation materials and methods could be carried out in the context of chronically implantable cortical surface implants based on µLEDs. Nevertheless, this section provides a thorough evaluation of our chronic device encapsulation stack, which includes parylene-C and/or medical-grade silicone, through accelerated ageing experiments. It should be noted that our encapsulation stack also includes the sapphire substrate, SU-8, polyimide PCB and medical-grade epoxy underfill. This section extensively evaluates the chronically-implantable devices, and not the acute device packaging used for electrophysiology; although the latter were briefly tested in solution to ensure no immediate device failures, or bubble formation through electrolysis from potential exposed electrical pathways.

Accelerated ageing testing allows the device encapsulation to be evaluated over an accelerated timescale. Exposing a device to the conditions under test, i.e. submerging the device in realistic biological fluid at 37 °C to simulate implantation on the surface of the brain, will provide a "real-time" simulation of the encapsulation performance. Simply by increasing this temperature we can accelerate the timescale, thereby evaluating months of years of operation within a few weeks of laboratory time. Mathematically, the acceleration factor can be described by the Arrhenius relation [13]:

$$Ageing factor = 2^{(T_{accel}-37)/10} \tag{3.1}$$

For example, accelerated ageing at a temperature of 60 °C, results in an acceleration factor of ∼4.9, thus one week of laboratory time corresponds to approximately one month of implantation *in-vivo*. Accelerated testing of medical devices is an advanced research field involving the study of encapsulation materials and more realistic biological conditions. Here, an "entry-point" approach has been taken, to provide an estimate of device longevity *in-vivo*. Accelerated ageing was facilitated by a custom system constructed in the laboratory (Fig. 3.10a). This consists of a heated water bath (Cole Parmer WB-400) to provide a safe, stable heat source. A custom-machined aluminium lid was placed over the bath and sealed with electrical tape. Devices under test are placed in a tank filled with phosphate-buffered saline (PBS) solution, and an opening in the lid allows the tank to be placed in the bath water (for heating) and also protrude through so devices can be optically monitored with microscope cameras. The µLED arrays are controlled using custom electronic current drivers, operated using a LabVIEW programme running on a PC (see Chapter 6); this programme also monitors PBS temperature using a 4-wire PT100 sensor connected via microcontroller, as well as recording and saving images taken by the microscope cameras. The system facilitates the accelerated testing of two devices simultaneously, however, is designed such that it can be scaled to more devices, or applied to other device configurations. In total, six devices were tested, as detailed in Table 3.3.

Under pulsed operation with the same protocol as used in *in-vivo* experiments: 5 mA (except Test 3: 10 mA; and Test 6: 2 mA per µLED), 20 Hz, 50% duty cycle, we measured the voltage required to drive a fixed current. As it would be impractical to test every µLED across a device, and this would not represent typical device operating conditions, we selected ten µLEDs across the device ensuring fair distribution across different rows, columns, edges and corners. There was a 5 second wait between measurements, meaning a cycle of each µLED approximately once per minute; after each measurement, the µLED was illuminated for three seconds at a lower drive current (∼0.1 mA) so a photograph could be taken to observe optical output. The variation in temperatures in Table 3.3 are due to experimental specificities with the testing rig, although have been accounted for in the acceleration factors in the relevant experiments. Experiment length refers to the overall time the devices were tested for, which

Device ID/description	Average temperature (°C)	Experiment length (hours)		First µLED failure time (hours)	Surviving µLEDs
		Real-time	Accelerated time		
Test 1 [H2-083] 15 µm PaC	56.0	42	155	9	6/10
Test 2 [H2-113] 15 µm PaC	56.0	41	154	40	6/10
Test 3 [H2-061] No PaC	44.4	429	1024	75	1/6
Test 4 [H2-120] 30 µm PaC + MED-2214	48.6	337	1026	75	3/10
Test 5 [H2-106] 30 µm PaC + MED-2214	48.6	337	1026	324	3/10
Test 6 [H2-112] 15 µm PaC	55.0	71	248	-	9/9

Table 3.3: **Full summary of accelerated testing experiments.**

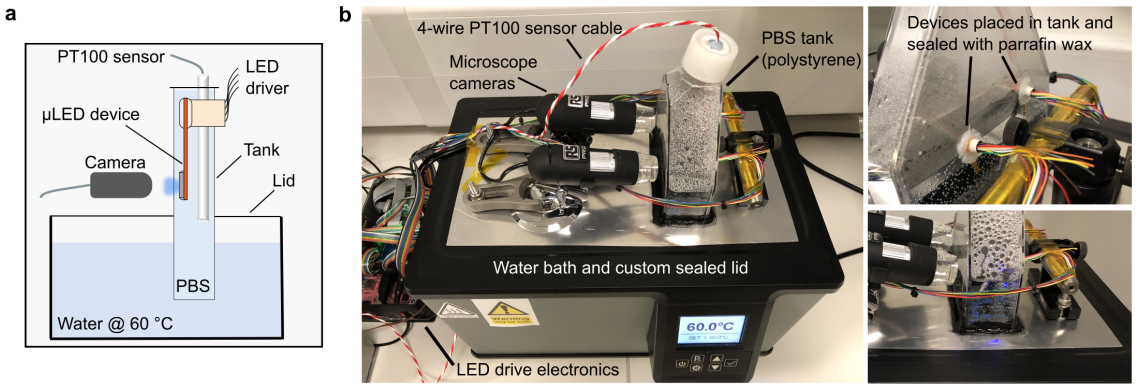


Figure 3.10: **Accelerated testing setup. (a)** Schematic of setup with device submerged in tank of phosphate-buffered saline (PBS) solution within a heated water bath; tank temperature measured with PT100 sensor and device imaged with microscope camera. **(b)** Annotated photographs of actual setup within laboratory.

varied depending on the failure point; devices were typically tested in pairs, also influencing this.

Fig. 3.11a, 3.11c and 3.11d show the measured voltage (normalised to the voltage before devices submerged in solution) over accelerated time. In Fig. 3.11a and 3.11d, each line represents an individually tested µLED with the mean shown in black. In Fig. 3.11c, the mean and standard deviation of voltage measurements over 3 sets of 3 µLEDs in parallel is shown. We define the failure time of a device as the point where a single µLED fails; often, a cascade of µLED failures is observed following this, consistent with solution penetrating the encapsulation and entering the device. A µLED failure is defined as the voltage required to drive the current falling significantly (the current finding an alternative path to ground through the conductive solution) and the µLED optically failing (no light output).

As shown in Fig. 3.11a, 3.11d (and Table 3.3) devices with a medical-grade silicone (MED-2214) encapsulation, as well as parylene-C, (Test 4, 5) survive for longer than those with parylene-C alone (Test 1, 2). Without any final encapsulation (Test 3), the device failed within 1 day. We focus on the best performing device (Test 5) for the following analysis. For this device, the experiment duration was over 1000 hours (42.5 days) of *in-vivo* equivalent time. As shown in Fig. 3.11a, the device failed after 324 hours (13.5 days) however this was under continuous operation of the µLEDs. To quantify the failure mechanism, µLEDs from untested rows and columns were evaluated before and after the entire experiment (i.e. 42.5 days) (Fig. 3.11b). These results confirm that the encapsulation strategy is stable when the device is passive, but that actively driving the µLEDs shortens the lifetime of the devices. Since *in-vivo* experiments require only intermittent µLED operation, we conservatively estimate the device should remain stable for at least one month of chronic implantation and experimentation.

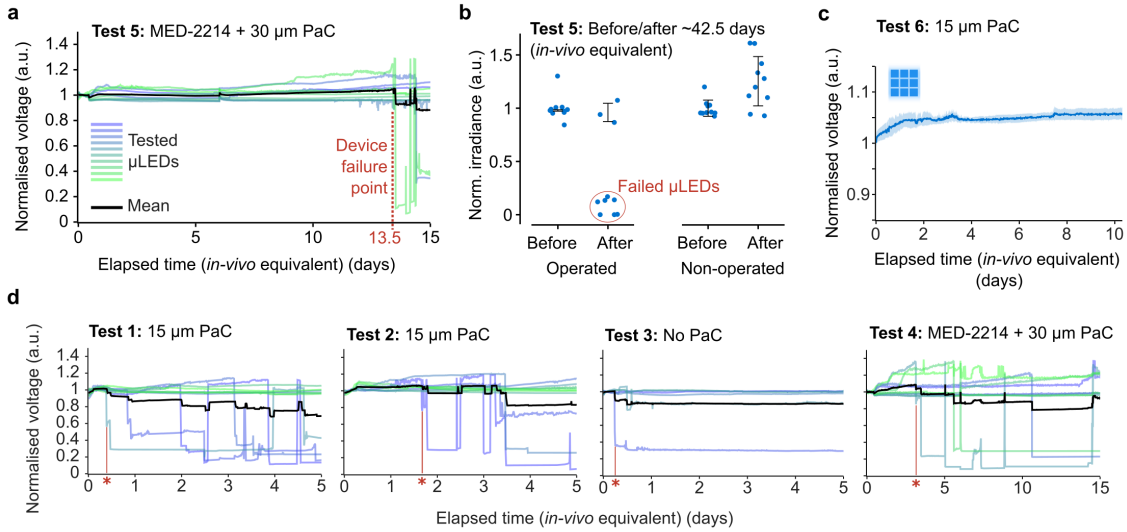


Figure 3.11: **Accelerated ageing results for all tested devices. (a)** Normalised measured voltage over *in-vivo* equivalent timescale showing individual µLED performance and device failure point as first µLED failure. **(b)** Optical performance of all tested µLEDs before and after full experiment for same device as in (a). **(c)** Normalised measured voltage during accelerated ageing for single 3x3 µLED pattern; experimental protocol same as in (a) with 2 mA per µLED. Mean and standard deviation of voltage measurements across 3 columns of 3 µLEDs in parallel. **(d)** Equivalent plot as in (a) for other evaluated devices.

It is worth noting that the variation in these results could be due to device encapsulation, but also inherent instability and inconsistency in the µLED array fabrication. It is difficult to directly separate these failure mechanisms. To attempt to elucidate this, destructively tested devices were inspected post-accelerated testing to try to identify the mechanism behind the encapsulation failures (Fig. 3.12). These images confirm our suspicion that PCB sidewall coverage of either parylene-C, silicone, or both was poor, resulting in a weak point where solution could ingress. Salt deposits were observed, particularly around manufacturing defects in the polyimide PCBs (Fig. 3.12a). Optical inspection of the µLED array itself shows corrosion as well as bubble formation around electrical tracks, the potential result of electrolysis. Voids in the underfill, present before accelerated testing due to the device assembly process, could be exacerbating the degradation, permitting solution to remain in between the µLED array and PCB (Fig. 3.12b).

Ideally more devices would have been tested, however due to time and resource constraints, many devices could not be committed to destructive testing. Instead, most devices were chronically implanted in mice for conducting behavioural experiments. These devices were, however, explanted and we evaluated both the electrical and optical performance demonstrating minimal degradation (Fig. 3.13).

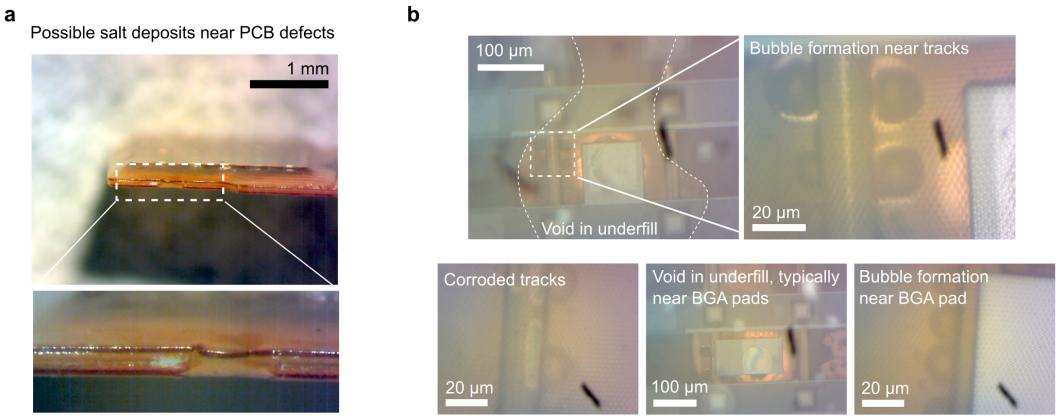


Figure 3.12: **Example failure mechanisms from accelerated testing for parylene-C coated device. (a)** Image showing possible salt deposit near a defect in the sidewall of the polyimide PCB stack. **(b)** Images showing failure mechanisms resulting from ingress of solution underneath µLED array due to voids in underfill around BGA pads. Bubbles form around tracks which were actively driven during accelerated testing.

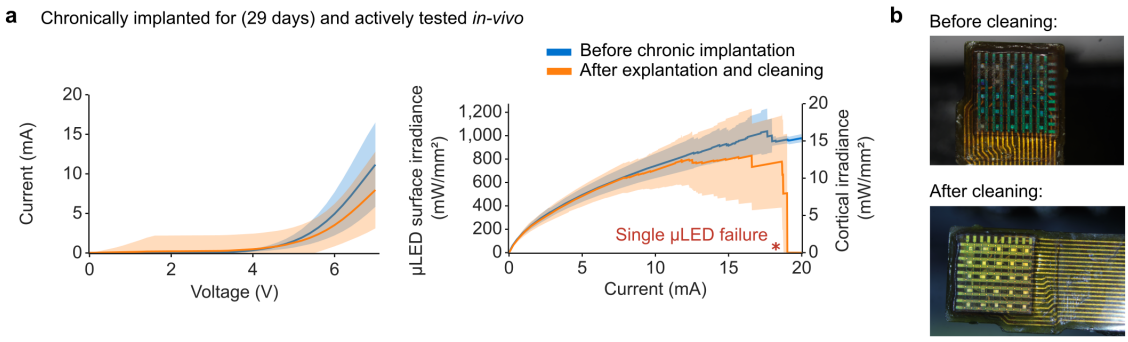


Figure 3.13: **Analysis of a device chronically implanted and explanted. (a)** I-V and L-I characteristics of all 100 µLEDs before and after chronic experiments (mean ± standard deviation). **(b)** Images showing small build-up of debris on device sapphire surface which was removed after cleaning (10 minutes with tergozyme in ultrasonic bath).

Evaluating explanted devices is important, in addition to accelerated ageing testing in the laboratory. This is because additional reactions occur in realistic biological conditions that cannot be emulated with submersion in PBS. Although the devices perform very well in solution and have acceptable stability, improvements could still be made to the encapsulation strategy; for example, PCB sidewall coverage is poor, resulting in a clear failure mechanism. Improving the underfill and reducing voids should further prevent solution remaining inside the device. This should be possible once the BGA pad adhesion issue is resolved as discussed in the previous section. We also believe there may be inherent instabilities in the µLED arrays themselves due to thermal effects on the SU-8 and/or polyimide passivation layer. This should be resolved such that the exact failure mechanism can be confidently established and future

encapsulation strategies optimised. Overall, these experiments confirm our devices are robust enough for chronic implantation and active usage over timescales consistent with months-long behavioural experimentation.

3.4 Integrated recording with flexible µECoG array

One limitation of our µLED array devices is a lack of integrated readout; notably, electrical recording functionality. This limits *in-vivo* experiments to utilise separate recording devices, for example penetrating silicon probes, which we demonstrate (Chapter 5). This approach poses challenges with experimental repeatability, and studies in behaving animals due to the likelihood of requiring a head-fixed experimental setup. We can observe behavioural readout alone, as we also demonstrate (Chapter 5), however, recording neuronal activity is critical in understanding circuit dynamics.

Large-scale electrical recording over the cortical surface offers insight into the neuronal circuits involved in a range of sensory and cognitive processes [14] with recording and stimulation capabilities being proven clinically [15, 16, 17]. Electrocorticography (ECoG) arrays offer large scale, and high-density recording across the cortical surface, without requiring penetrating electrodes in the brain tissue. These devices are often flexible and can conform to the contoured structure of the cortical surface in humans, offering a clear route to clinical use [18, 19]. Monolithic integration of implantable optogenetic stimulation and electrical recording from the cortical surface has significant potential in deepening our understanding of cortical processes. Great strides have already been made in this area [20, 21], however, further technological development and research is needed to understand how best to implement these devices for advanced neuroscience studies; and to the potential eventual use of optogenetics in the clinic [22].

In this section, a prototype "hybrid" device is presented based on the integration of the 100-pixel µLED arrays introduced earlier, and a 32-channel µECoG array based on a novel parylene-C/SU-8 fabrication process [23] (Amélie Albon and Charles Rezaei-Mazinani; see Acknowledgements). Briefly, the µECoG array is fabricated using gold deposited on parylene-C/SU-8, fabricated on a glass wafer which acts as the substrate. After patterning of the 50 µm diameter electrodes, electrical tracks and bonding pads, and deposition of a further SU-8/parylene-C layer, the devices are released from the glass substrate. The result is an extremely thin (∼5 µm), flexible device which can be implanted on the cortical surface [24] (Fig. 3.14a). In this section, we show integration of the flexible µECoG array in front of the sapphire surface of the µLED arrays and perform laboratory characterisation; fabrication of flexible µECoG array is outwith the scope of this thesis and was done by collaborators as discussed. The µECoG array is optically transparent and has been specifically designed such that electrodes and tracks do

not obstruct the light from the µLEDs (Fig. 3.14b bottom). This device is theoretically capable of measuring local field potential (LFP) signals, multi-unit activity (MUA) and potentially individual neuronal responses [19], however, due to project time constraints was not tested *in-vivo*. Discussion on the potential future direction of this project is provided at the end of this section.

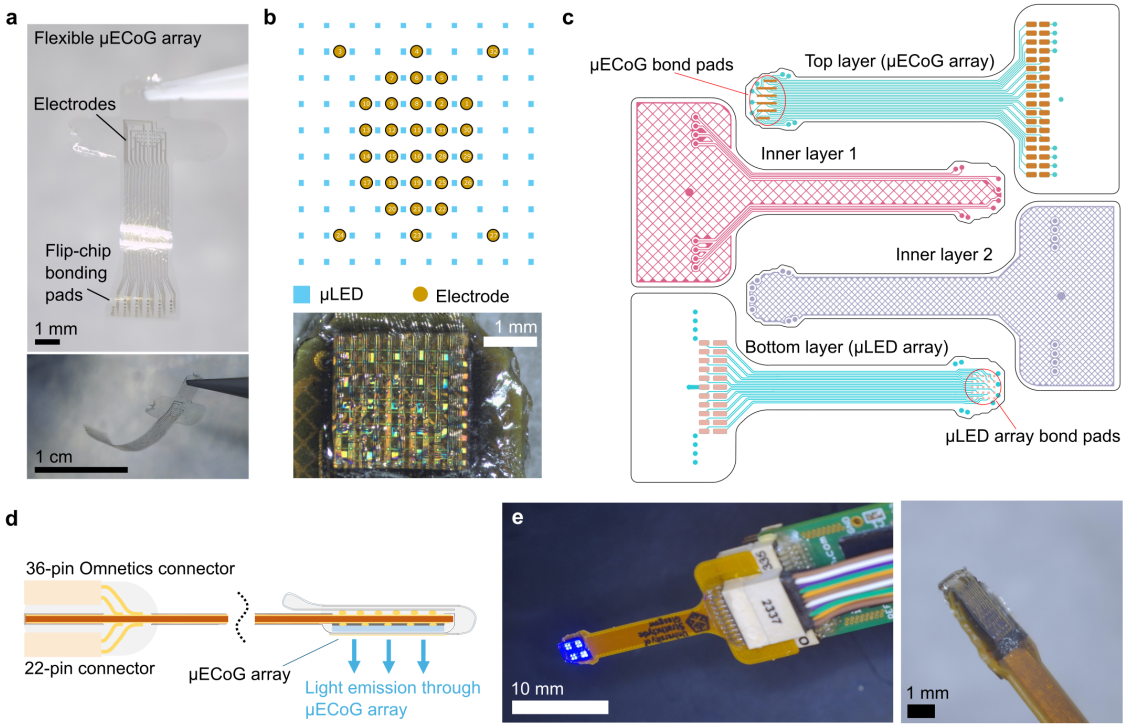


Figure 3.14: **Hybrid device integrating 32-electrode µECoG array with 100-pixel µLED array. (a)** Photographs showing µECoG array. **(b)** Illustration of electrode layout with respect to µLED array. **(c)** PCB layout of 4-layer PCB to facilitate monolithic integration of devices. **(d)** Schematic illustration of cross-section of hybrid device. **(e)** Final packaged hybrid device with µLEDs illuminated and Intan RHD headstage connected to electrodes (left); zoomed image of device tip showing electrode tracks (right).

Tables 3.4 and 3.5 summarise the process for integrating the µECoG array to the µLED arrays and the monolithic packaging of the overall system.

The process is similar to the packaging of the chronic devices described previously. A 4-layer, flexible polyimide PCB was designed with bond pads on both the top (µECoG array) and bottom (µLED array) for flip-chip bonding. The bond pads are bump bonded and the µLED array attached using thermosonic flip-chip bonding, as before (Steps i, ii, iii, iv). The PCB is then coated with parylene-C to provide the µLED array resistance against conductive solution; prior to this the top bond pads and connector pads are protected with Kapton tape (Step v). Bump bonds are attached to the top bond pads (opposite to the µLED array which is bonded on the bottom) (Step vi). The flexible µECoG array is then flip-chip bonded using

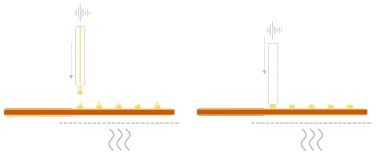
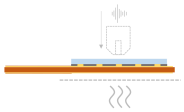
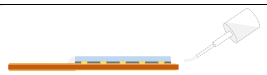
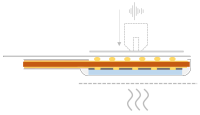
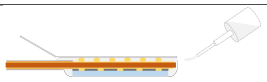
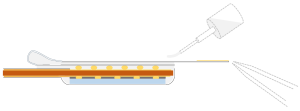
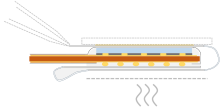
Process step description	Cross-sectional schematic
Step i: Bump bonding • Bonding parameters: bond (power: 3.9, time: 5.5, force: 1); temperature (75 °C)	
Step ii: Pre-underfill • Material: MG Chemicals 9510	
Step iii: Flip-chip bonding µLED array • Flip-chip bonding parameters: ultrasonic (power: 1500 mW, time: 500 ms); force (20 N); temperature (75 °C) • Underfill cure: temperature (150 °C)	
Step iv: Bevelled underfill • Material: EPO-TEK MED-301-2	
Step v: Parylene-C encapsulation	
Step vi: Bump bonding/coining µECoG side • Bonding parameters: bond (power: 3.9, time: 5.5, force: 1); temperature (75 °C)	
Step vii: Flip-chip bonding µECoG array • Flip-chip bonding parameters: ultrasonic (power: 1500 mW, time: 500 ms); force (20 N); temperature (75 °C)	
Step viii: Underfill µECoG array • Material: EPO-TEK MED-301-2	
Step ix: First fold • Material: EPO-TEK MED-301-2	
Step x: Second fold and µECoG array alignment • Bake temperature (100 °C)	

Table 3.4: **Summary of process steps for hybrid device packaging 1.**

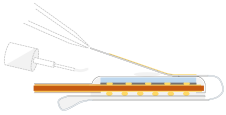
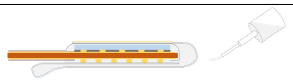
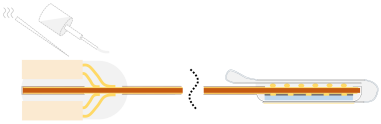
Process step description	Cross-sectional schematic
Step xi: Gluing µECoG array to sapphire surface • Material: EPO-TEK MED-301-2	
Step xii: Underfill remaining regions • Material: EPO-TEK MED-301-2	
Step xiii: Soldering and potting connectors • Materials: Low-temperature solder paste, EPOTEK MED-301-2	

Table 3.5: **Summary of process steps for hybrid device packaging 2.**

the same machine, process and parameters as the µLED array (Step vii). A piece of sapphire larger than the bonding area was used, as the transducer does not cover all the bonds; going forward, a custom collette could be manufactured. Flip-chip bonding of flexible polyimide devices has been previously demonstrated [25], however, not with parylene-C. Therefore further characterisation of the electrical and mechanical quality of the connection should be investigated. Nevertheless, the interconnection was resistant to mechanical force (pulling with tweezers) and impedance was not compromised (Fig. 3.15a). Furthermore, a "pre-underfill" was not required here as when flip-chip bonding the µLED arrays suggesting a mechanically robust interconnection. The bonded area was then underfilled using medical-grade epoxy (EPO-TEK MED-301-2) (Step viii). Since the µECoG array is bonded to the opposite side of the PCB than the µLED array, it must be folded around so that the electrodes are in front of the sapphire surface and can be secured in place. The µECoG array is firstly folded back on itself and glued in position (Step ix). A further fold brings the µECoG array in front of the sapphire surface (Step x). The electrodes must then be aligned manually into position, by hand. A glass slide was used to maneuver the µECoG array which attaches to the glass slide with static force. With the glass slide in place, the device is baked at 100 °C to maintain the positioning of the µECoG array. Using tweezers, the µECoG array is lifted up, a small volume of MED-301-2 dispensed on the sapphire surface and dropped back down (Step xi). Due to the baking step, the electrodes should maintain the previously-aligned position. The device is then reinforced using MED-301-2, for example to fill the gap where the µECoG array is folded around the front of the PCB (Step xii). Care should be taken at this step to ensure epoxy does not flow

onto the electrodes. The design uses a separate connector for the µLEDs and electrodes for interconnection to external electronics. For the electrodes, a 36-pin Omnetics rectangular nano connector is used for compatibility with Intan headstages. The µLEDs use a 22-pin Omnetics rectangular nano connector (2 pins connect to ground plane). The electrode connector is soldered first and potted, then the PCB is rotated and the µLED connector soldered and potted, completing the hybrid device packaging process.

The result is a monolithically-packaged device offering dense optogenetic stimulation and electrical recording of the cortical surface. The device is mechanically robust and potentially re-usable, with seemless integration to Intan headstages which are ubiquitous in neuroscience laboratories. The electrodes were coated with platinum (Pt)-black to increase the impedance and charge injection capacity, meaning the device could also be used for stimulation. All functioning electrodes (22/32 remained functioning after assembly) were coated with Pt-black and the impedance measured (Fig. 3.15a); these values are good enough to ensure robust recordings [23]. The optical power from the µLEDs was measured before and after integration of the µECoG array, demonstrating a small reduction in power and hence cortical irradiance (Fig. 3.15b), however this is still above-threshold as will be discussed in Chapter 5. This was probably influenced by a small misalignment causing some µLEDs to be obstructed by electrical tracks (Fig. 3.14b bottom).

Lastly, we characterised the cross-talk between optical stimulation of the µLEDs and electrical pickup on the recording electrodes and tracks. Cross-talk could arise from capacitive (electrical) coupling, or optical coupling in conductive solution through the Becquerel effect. To assess this, an experiment was set up with the device submerged in conductive solution (phosphate buffered saline, PBS). The µLEDs were connected to the electronic driver system and an Intan recording headstage (RHD 2000) connected to the electrodes (Fig. 3.15c top-right). The ground and reference pins on the RHD2000 were both in the solution. A µLED stimulation protocol was ran similar to that used in *in-vivo* experiments (delivering current pulses at 20 Hz, 50% duty cycle) with the electrode trace recorded at 30 kHz, to assess realistic *in-vivo* stimulation artefacts. From the waveforms, it can be seen that each current pulse causes an artefact, which reduces over time with a set time constant. There is an additional smaller artefact which consistently appears before onset of the current pulses. This arises from the switching circuitry used to control the common n, i.e. depending on which row/rows are being illuminated, the switch will be closed (µLED cathodes grounded) or open (µLED cathodes floating/high impedance) (Fig. 3.2b). Each switching action causes a change in the potential simultaneously on 10 µLED cathodes, which could be mitigated by setting the switch position earlier before the current pulses; or by optimising the electronic design. Nevertheless, after filtering for multi-unit analysis (MUA) only a few milliseconds of artefact remain, which can therefore be removed by blanking, and the transient also disappears. However, this device may

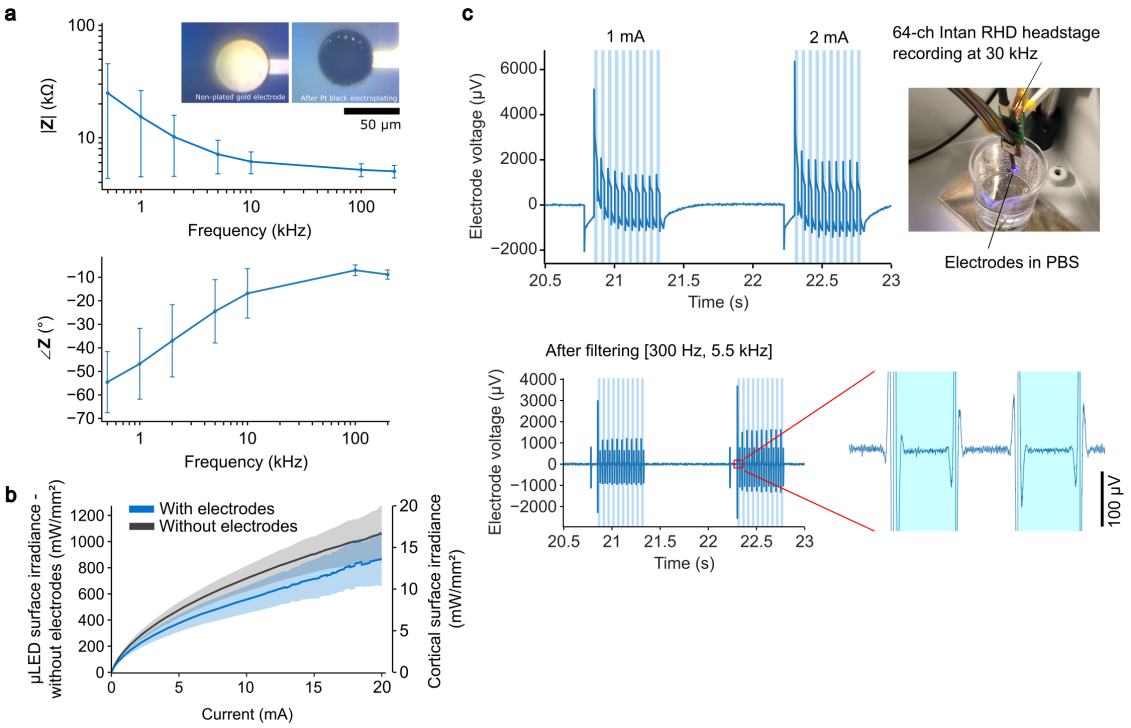


Figure 3.15: **Characterisation of hybrid integrated device. (a)** Impedance magnitude (top) and phase response (bottom) for all Pt-black plated electrodes; mean and standard deviation of 22 electrodes. **(b)** Irradiance-current (L-I) characteristic of all µLEDs across a device before and after integration of µECoG array (mean and standard deviation of n=100 µLEDs). **(c)** Recorded waveform (30 kHz) of single electrode under simultaneous pulsed operation of single µLED (10 pulses at 20 Hz, 50% duty cycle). The zoomed inset shows noise returning to baseline so electrophysiology recordings can be made within pulse time window. Photograph of experimental setup within Faraday chamber shown at top-right.

be limited in its ability to record local field potentials (LFPs), because the transients would appear at a similar frequency range to that being recorded.

The hybrid device offers a scalable platform to integrate high-resolution electrical recording or stimulation from the cortical surface, with optogenetic stimulation through our 100-pixel µLED arrays. Although the device shown here is in-principle ready for *in-vivo* experiments, due to project time constraints none were carried out. A pilot experiment trialing this device in an acute preparation, demonstrating stimulation and recording capabilities from the cortical surface, would be an obvious next step for the project. Nevertheless, there are some obstacles to scaling the production of this device, mainly due to the difficulty in reproducing the complex packaging steps and the time and care required to produce each device. The packaging process also makes the µECoG array lose its flexibility, resulting in a rigid device, which could be a limiting factor over other ECoG recording/stimulation devices [19], or devices which integrate both µLEDs and electrical recording [21]. These latter devices are impressive, and match

our system on specification, however there remain open questions on optical intensity and whether stable, robust behavioural responses can be elicited, as well as long-term chronic operation. Both our µLED arrays and packaging process have been shown to be stable *in-vivo*, allowing potentially months-long chronic experiments. Low-power operation further opens the prospect of wireless operation. The hybrid device is also very mechanically robust and could be readily re-used, although Pt-black electrodes are not ideal as they may need re-plated. Another material like PEDOT-PSS, a conductive polymer, would be a better choice [26]. Lastly, the ease of integrating the system with the already ubiquitous Intan recording headstages, could make this system an appealing choice for advanced neuroscience studies into cortical function.

Bibliography

1. Greer, R. *et al.* Chronically implantable LED arrays for optogenetic cortical surface stimulation in mice. en. *Nature Communications* **17,** 878. ISSN: 2041-1723. doi:10.1038/s41467-025-68191-5. (2026) (Jan. 2026).
2. Holtmaat, A. *et al.* Long-term, high-resolution imaging in the mouse neocortex through a chronic cranial window. *Nature protocols* **4,** 1128–1144. ISSN: 1754-2189. doi:10.1038/nprot.2009.89. (2025) (2009).
3. Gorski, J. A. *et al.* Cortical Excitatory Neurons and Glia, But Not GABAergic Neurons, Are Produced in the Emx1-Expressing Lineage. en. *Journal of Neuroscience* **22,** 6309–6314. ISSN: 0270-6474, 1529-2401. doi:10.1523/JNEUROSCI.22-15-06309.2002. (2026) (Aug. 2002).
4. Madisen, L. *et al.* A toolbox of Cre-dependent optogenetic transgenic mice for light-induced activation and silencing. en. *Nature Neuroscience* **15,** 793–802. ISSN: 1546-1726. doi:10.1038/nn.3078. (2026) (May 2012).
5. Chen, A., Nguyen, L., Burke, T. & Belani, J. *Wire bonding: Present and future trends* in *Proceedings of 15th IEEE/CHMT International Electronic Manufacturing Technology Symposium* (Oct. 1993), 45–51. doi:10.1109/IEMT.1993.398223. (2025).
6. Constantin, C. P., Aflori, M., Damian, R. F. & Rusu, R. D. Biocompatibility of Polyimides: A Mini-Review. *Materials* **12,** 3166. ISSN: 1996-1944. doi:10.3390/ma12193166. (2025) (Sept. 2019).
7. Park, S. I. *et al.* Soft, stretchable, fully implantable miniaturized optoelectronic systems for wireless optogenetics. eng. *Nature Biotechnology* **33,** 1280–1286. ISSN: 1546-1696. doi:10.1038/nbt.3415 (Dec. 2015).
8. Zhang, Z. & Wong, C. Recent advances in flip-chip underfill: materials, process, and reliability. *IEEE Transactions on Advanced Packaging* **27.** Conference Name: IEEE Transactions on Advanced Packaging, 515–524. ISSN: 1557-9980. doi:10.1109/TADVP.2004.831870. (2025) (Aug. 2004).

9. Caldwell, R. *et al.* Characterization of Parylene-C degradation mechanisms: *In vitro* reactive accelerated aging model compared to multiyear *in vivo* implantation. *Biomaterials* **232,** 119731. ISSN: 0142-9612. doi:10.1016/j.biomaterials.2019.119731. (2025) (Feb. 2020).

10. Zare, M., Ghomi, E. R., Venkatraman, P. D. & Ramakrishna, S. Silicone-based biomaterials for biomedical applications: Antimicrobial strategies and 3D printing technologies. en. *Journal of Applied Polymer Science* **138,** 50969. ISSN: 1097-4628. doi:10.1002/app.50969. (2025) (2021).

11. Rajalingham, R. *et al.* Chronically implantable LED arrays for behavioral optogenetics in primates. en. *Nature Methods* **18,** 1112–1116. ISSN: 1548-7105. doi:10.1038/s41592-021-01238-9. (2026) (Sept. 2021).

12. Idil, A. S. *et al. On the Stability of Silicone-Encapsulated CMOS ICs for Active Implantable Devices: 4.3 Years of Accelerated Life Testing* en. Pages: 2025.05.21.655306 Section: New Results. May 2025. doi:10.1101/2025.05.21.655306. (2026).

13. Boehler, C., Carli, S., Fadiga, L., Stieglitz, T. & Asplund, M. Tutorial: guidelines for standardized performance tests for electrodes intended for neural interfaces and bioelectronics. en. *Nature Protocols* **15,** 3557–3578. ISSN: 1750-2799. doi:10.1038/s41596-020-0389-2. (2025) (Nov. 2020).

14. Maynard, E. M., Nordhausen, C. T. & Normann, R. A. The Utah intracortical Electrode Array: a recording structure for potential brain-computer interfaces. eng. *Electroencephalography and Clinical Neurophysiology* **102,** 228–239. ISSN: 0013-4694. doi:10.1016/s0013-4694(96)95176-0 (Mar. 1997).

15. Musk, E. & Neuralink. An Integrated Brain-Machine Interface Platform With Thousands of Channels. EN. *Journal of Medical Internet Research* **21,** e16194. doi:10.2196/16194. (2026) (Oct. 2019).

16. Mestais, C. S. *et al.* WIMAGINE: wireless 64-channel ECoG recording implant for long term clinical applications. eng. *IEEE transactions on neural systems and rehabilitation engineering: a publication of the IEEE Engineering in Medicine and Biology Society* **23,** 10–21. ISSN: 1558-0210. doi:10.1109/TNSRE.2014.2333541 (Jan. 2015).

17. Fernández, E. *et al.* Visual percepts evoked with an intracortical 96-channel microelectrode array inserted in human occipital cortex. en. *The Journal of Clinical Investigation* **131.** ISSN: 0021-9738. doi:10.1172/JCI151331. (2026) (Dec. 2021).

18. Viventi, J. *et al.* Flexible, foldable, actively multiplexed, high-density electrode array for mapping brain activity in vivo. en. *Nature Neuroscience* **14,** 1599–1605. ISSN: 1546-1726. doi:10.1038/nn.2973. (2026) (Dec. 2011).

19. Khodagholy, D. *et al.* NeuroGrid: recording action potentials from the surface of the brain. en. *Nature Neuroscience* **18,** 310–315. ISSN: 1546-1726. doi:10.1038/nn.3905. (2026) (Feb. 2015).

20. Pollmann, E. H. *et al.* A subdural CMOS optical device for bidirectional neural interfacing. en. *Nature Electronics* **7,** 829–841. ISSN: 2520-1131. doi:10.1038/s41928-024-01209-w. (2026) (Sept. 2024).

21. Gu, W., Wang, L., Wang, X., Zhao, C. & Guan, S. Large-Scale, High-Density MicroLED Array-Based Optogenetic Device for Neural Stimulation and Recording. *Nano Letters* **24,** 15252–15259. ISSN: 1530-6984. doi:10.1021/acs.nanolett.4c03645. (2026) (Dec. 2024).

22. Sahel, J.-A. *et al.* Partial recovery of visual function in a blind patient after optogenetic therapy. *Nature medicine* **27,** 1223–1229. (2026) (2021).

23. Cornuéjols, R. *et al.* Design, Characterization, and In Vivo Application of Multi-Conductive Layer Organic Electrocorticography Probes. eng. *ACS applied materials & interfaces* **15,** 22854–22863. ISSN: 1944-8252. doi:10.1021/acsami.3c00553 (May 2023).

24. Taylor, J. *et al.* Cortical neuroprostheses improve auditory coding and perception compared to cochlear implants. *bioRxiv.* doi:10.1101/2025.10.27.684780 (2025).

25. Steins, H. *et al.* A flexible protruding microelectrode array for neural interfacing in bioelectronic medicine. en. *Microsystems & Nanoengineering* **8,** 1–15. ISSN: 2055-7434. doi:10.1038/s41378-022-00466-z. (2025) (Dec. 2022).

26. Dijk, G., Pas, J., Markovic, K., Scancar, J. & O'Connor, R. P. PEDOT:PSS-coated platinum electrodes for neural stimulation. eng. *APL bioengineering* **7,** 046117. ISSN: 2473-2877. doi:10.1063/5.0153094 (Dec. 2023).

Chapter 4

Optical and thermal considerations for cortical surface implanted µLEDs

This chapter develops an operational framework for the implantable µLED array systems to support the *in-vivo* experimentation in the mouse auditory cortex. Specifically, the optical and thermal implications of optogenetic stimulation of the cortical surface using our µLED arrays has been investigated. Optical modelling is based on realistic Monte Carlo models of the µLED array to understand light propagation in the brain. The thermal analysis is then carried out incorporating both infrared imaging and mathematical modelling to understand thermal effects on the brain surface and at depth. This work is based off techniques developed within the lab [1, 2], and adapted to the µLED arrays presented here. Here, we have focussed on the µLED arrays presented in Chapter 3 as this work also has direct implications on the use of these devices in *in-vivo* experiments, as will be described in the next chapter, Chapter 5. These results could be applied to a multitude of cortical surface-implanted µLEDs, however. The key results of this chapter, including optical and thermal modelling and experimental work have been published in [3].

4.1 Optical modelling

Optical models were developed for the µLED arrays, employing a Monte Carlo (ray tracing) model (Zemax OpticStudio 2021 in non-sequential mode). In Monte Carlo modelling, light is simulated as packets of optical power (like a group of photons) which initially travel in random directions from a defined source, each starting with a known amount of power. They are influenced by how the model is defined, such as the geometry and materials (resulting in reflections or refractions) and the scattering and absorption properties of these materials. The result is the packets can change direction and lose power, eventually terminating once they have lost enough power [4]. Here, the model is defined in three dimensions, using 5 µm or 10

µm cubic voxels.

A Monte Carlo model of the µLED array was developed to understand how light propagates from the µLED, through the sapphire substrate and into the brain tissue (Fig. 4.1a).

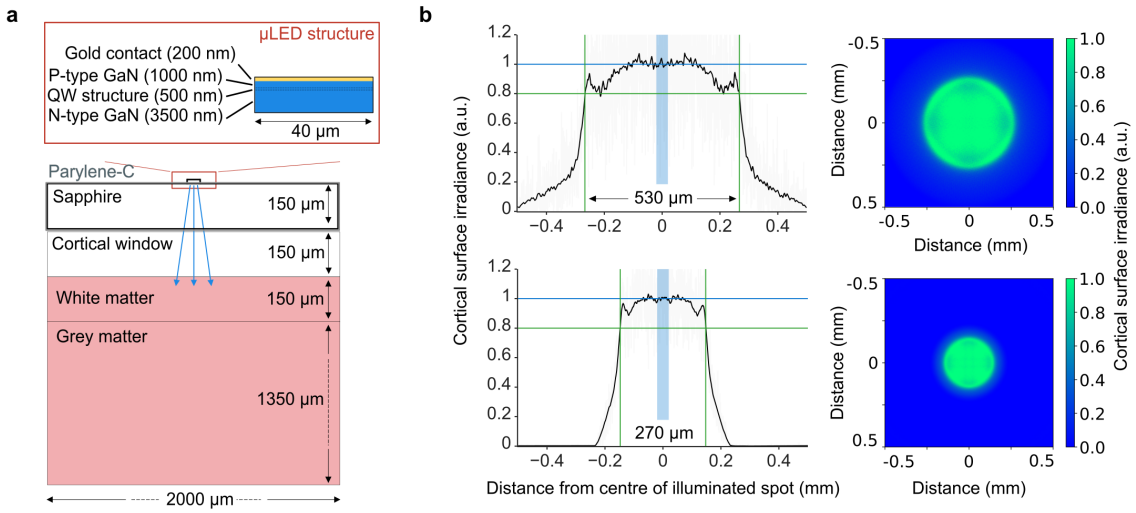


Figure 4.1: **Optical models for inference of cortical surface irradiance. (a)** Annotated schematic of modelled µLED atop sapphire substrate, with realistic cortical tissue modelled with the optical properties for brain tissue, and scattering coefficients for white matter and grey matter, as well as optional cortical window. **(b)** Normalised irradiance from 2D surface representing the cortical surface; with a cortical window (top) and direct implantation on cortical surface (bottom). In the left graphs, the blue shaded regions show a 40 µm width (equivalent to the µLED size). The horizontal blue line shows the average value of the distribution over this 40 µm (peak value). The green lines represent 80% from the peak (horizontal), and the corresponding distance (vertical).

Simulations were typically run with $2{\cdot}10^7$ photons. The resulting distribution was later scaled by the actual power generated in the µLED, typically in the 10s of mW range. A 2D surface can be placed into the model and the amount of power that reached the surface can be extracted. Fig. 4.1b shows a 1D cut of the normalised irradiance distribution on a 1 x 1 mm^2 surface representing the brain surface, both with and without a glass cortical window between the sapphire surface and brain tissue; the heat maps show the distribution of irradiance across the 2D surface. The purpose of the cortical window used in electrophysiology experiments is described in Chapter 5. These results show clearly an illuminated circular "spot" on the cortical surface, therefore if we know the power reaching the modelled 1 x 1 mm^2 surface and the area of this spot, then a cortical surface irradiance can be inferred:

$$L_{cortical\ surface} = \frac{P_{cortical\ surface}}{(\pi r^2)} = \frac{P_{cortical\ surface} \cdot 4}{(\pi d^2)} \tag{4.1}$$

Where L is the cortical surface irradiance, P is the optical power reaching the surface and d is the diameter of the illuminated spot. To find the diameter of the illuminated spot on the

cortical surface, the 80% level from the peak of the irradiance distribution was used. The peak was defined as the mean of the distribution within the central 40 µm. With a cortical window, the diameter is 530 µm and without, the diameter is 270 µm (Fig. 4.1b left).

To find the irradiance at the surface of the µLED itself (instead of the cortical surface), the 1 x 1 mm^2 surface can be positioned in front of the µLED and the area term is the surface area of the µLED:

$$L_{\mu LED\ surface} = \frac{P_{\mu LED\ surface}}{(40\ \mu m)^2} \tag{4.2}$$

Since the simulated power (i.e. the power generated within the µLED structure) is known, a scale factor can be determined to infer the power or irradiance on the surface of interest, from the power generated by the µLED:

$$P_{surface} = P_{\mu LED}\ \cdot\ scalefactor \tag{4.3}$$

Where $scalefactor = 0.207$ (cortical surface without window), $scalefactor = 0.205$ (cortical surface with window), $scalefactor = 0.345$ (µLED surface). The scale factor represents the fraction of photons generated by the µLED that reach the surface of interest (extraction efficiency), determined through optical modelling. The µLED surface scale factor is less than 1 because most photons remain trapped within the µLED structure. This represents a realistic LED where the external quantum efficiency is dictated by the internal quantum efficiency and the extraction efficiency.

Cortical surface irradiance gives us a reference to assess the efficacy of the devices to activate commonly used optogenetic opsins, such as Channelrhodopsin-2 (ChR2). The amount of light required for reliable activation of these opsins is typically quoted in mW/mm^2 and for ChR2 as used in this study, the threshold for reliable activation is generally accepted to be approximately 1 mW/mm^2 [5]. To achieve 1 mW/mm^2 at the target neurons located in layer 2/3, a higher cortical surface irradiance is required, for example 10 mW/mm^2, to account for attenuation due to scattering and absorption in the brain tissue. To understand the attenuation and spread of light within the tissue, the brain has been modelled incorporating suitable optical properties, with the scattering and absorption coefficients for the white and grey matter. Layer 1 consists of myelinated axons (white matter) and uses a scattering coefficient, $\sigma_s = 50\ mm^{-1}$, absorption coefficient, $\sigma_a = 0.14\ mm^{-1}$ and anisotropy factor, $g = 0.78$. The remaining cortical layers consist of cells and neurons (grey matter) and use $\sigma_s = 11.1\ mm^{-1}$, $\sigma_a = 0.07\ mm^{-1}$ and $g = 0.88$. These scattering and absorption coefficients are based on reputable literature where measurements were made from human tissue [6]; recent literature on mouse grey matter suggests similar optical properties [7]. The white matter layer was estimated as 150 µm thick [8, 9] with the remaining grey matter extending the length of the

model (to a depth of 1.5 mm, consistent with the thickness of the mouse neocortex).

A 3D detection volume was incorporated inside the modelled brain, allowing optical power flux to be extracted for each voxel. Dividing this flux by the surface area of the voxel – $(5\ \mu m)^2$ or $(10\ \mu m)^2$ – gives an irradiance distribution throughout the brain (Fig. 4.2b). The mean of the voxel values within the illuminated spot on the cortical surface of the 3D volume was assumed to be equivalent to the cortical surface irradiance; in both panels in Fig. 4.2, the cortical surface irradiance was set to 10 mW/mm^2 and a contour was drawn at the 1 mW/mm^2 level.

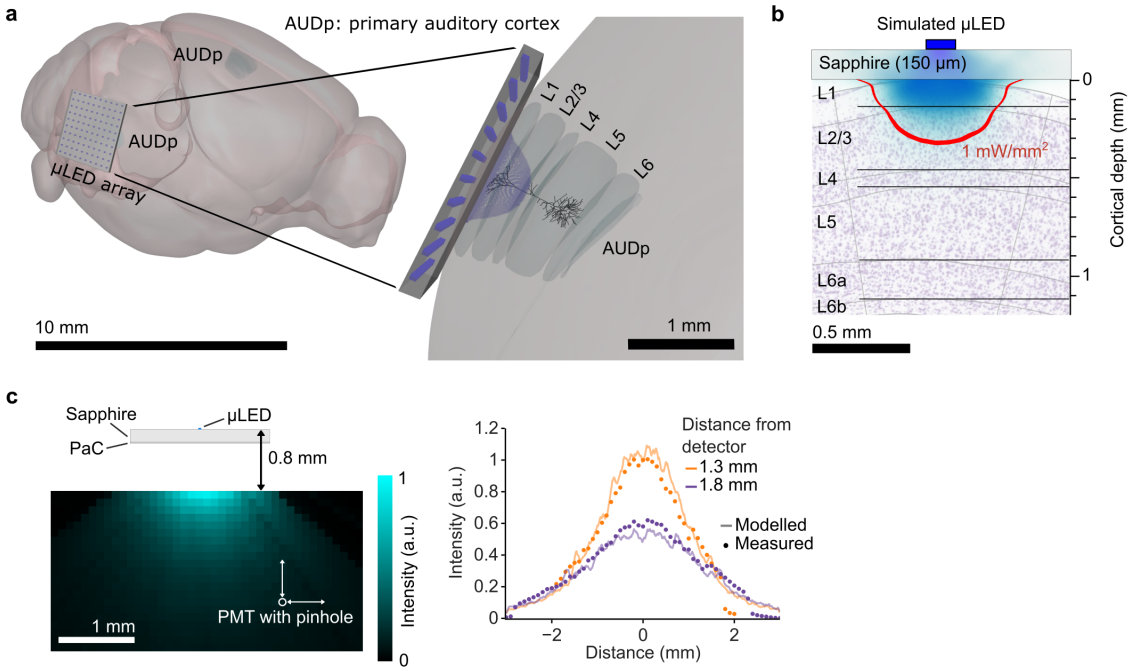


Figure 4.2: **Modelled optical profile of light penetration into mouse auditory cortex. (a)** 3D model of µLED array over mouse auditory cortex; the 1 mW/mm^2 level is shown given a single µLED illuminating a cortical surface irradiance of 10 mW/mm^2; layer 5 pyramidal neuron has been drawn for scale. Illustration produced using [8, 10]. **(b)** Modelled optical profile of light propagation from a single illuminated µLED; irradiance levels are the same as (a). **(c)** Emission profile of single µLED through air, measured using a PMT detector with pinhole manoeuvred through emission profile (100 µm measurement spacing) (left). Emission profile at different depths compared with cuts at equivalent distance from an optical model of device in air (right).

Fig. 4.2b shows that a cortical surface irradiance of 10 mW/mm^2 could allow reliable optogenetic activation of ChR2-expressing neurons within layer 2/3, as explained in Chapter 2, where the cortical layer structure is introduced. In this model, the sapphire surface of the device was positioned in direct contact with the brain tissue, however in our electrophysiology experiments there is a cortical window; the implications of this are addressed in Chapter 5.

For optical sources residing on the cortical surface, carrying out optical modelling is crucial. This is because the propagation of light within the brain is a function of both the delivered

optical power and the illuminated spot area. Referring solely to cortical surface irradiance is insufficient, as a proportional increase or decrease in both delivered optical power and illuminated surface area would result in an identical cortical surface irradiance. For example, if the optical power was doubled, and the area this power is present over was also doubled, then, despite an equivalent cortical surface irradiance, more photons are entering the brain resulting in a greater activated volume and depth.

Furthermore, experimental validation of optical models is important to ensure the model is representative of the real device. An experiment was carried out with a pinhole detector to validate the optical modelling experimentally. A pinhole (100 μm in diameter, 40 μm thick) is positioned in front of a photodetector (Fig. 4.2c). The pinhole was positioned as close to the detector as possible to ensure that all light that passes through the pinhole is collected by the detector. The pinhole-photodetector is then translated through the μLED emission profile measuring the light intensity at each point. The measurement is repeated at various μLED–pinhole distances. The light measurement at each point allows for the beam profile reconstruction with a resolution of approximately the pinhole size. This method is modified from [7] and was carried out within the group (Niall McAlinden; see Acknowledgements). Furthermore, the emission was also imaged through fluorescein and compared with an equivalent model (Fig. 4.7).

4.2 μLED array electrical and optical characteristics

To characterise the μLED arrays experimentally, and eventually feed this data into models, a laboratory setup was utilised along with a LabVIEW software programme. This setup was designed for efficiency as due to the quantity of μLED arrays, each consisting of 100 μLEDs, manual characterisation of individual μLEDs would be infeasible. Furthermore, only sampling a few μLEDs per array and not all of them could miss a non-functioning or poorly performing μLED. Even a single shorted μLED could cause significant issues due to matrix addressing. With the H2 batch primarily used in this thesis, most devices had 100/100 of the μLEDs functioning correctly.

The laboratory characterisation setup consists of a probe card, with two orthogonal rows of 10 pins, pitch-matched to the bond pad layout of a μLED array (Fig. 4.3a). The probe card was connected to a source/measurement unit and multiplexer to allow addressing of individual μLEDs; to characterise packaged devices, a custom-built cable was used instead of the probe card. The μLED array was placed atop an optical detector (at a known distance) which was connected to an optical power meter. The software programme engaged a sweep of voltages while measuring current and optical power; proper practice such as switching lights off and correctly zeroing the power meter were employed throughout these experiments. A schematic

illustration of the µLED array characterisation setup is shown in Fig. 4.3b.

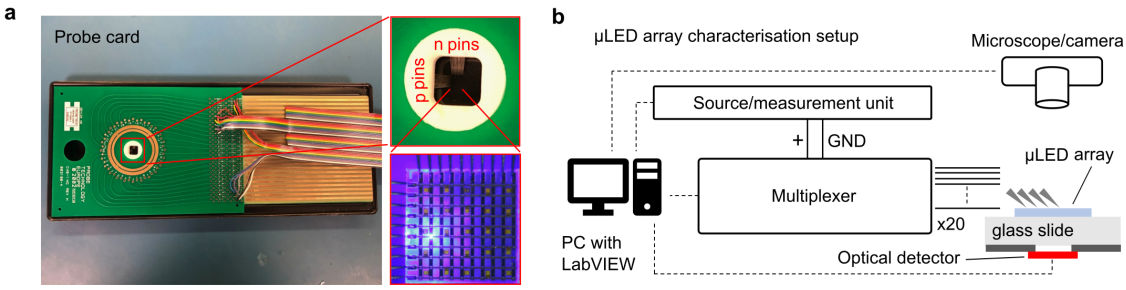


Figure 4.3: **µLED array laboratory characterisation setup. (a)** Photograph of probe card illustrating how unpackaged µLED arrays are tested. **(b)** Schematic of setup permitting rapid characterisation of all 100 µLEDs across a device.

The power meter measures optical power in watts (W) that reaches the detector, which should then be converted into the optical power generated by the µLED. This number can then be used along with the previously described optical modelling, to infer the cortical surface irradiance. A second model was developed to calculate the fraction of light which reaches the optical detector, simulating the laboratory characterisation setup as shown in Fig. 4.4a. This is equivalent to the solid angle calculation.

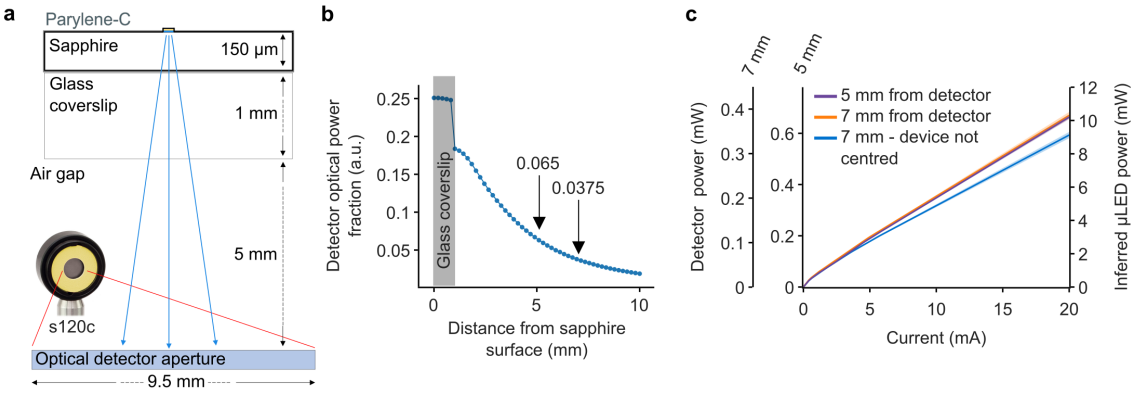


Figure 4.4: **Inference of µLED optical power from measurement. (a)** Annotated schematic of optical model emulating laboratory setup used for experimentally measuring µLED optical power. **(b)** Modelled fraction of optical power generated by the µLED that reaches an optical detector, with detector positioned at different distances from sapphire surface; in our laboratory setup, the detector aperture is at a distance of 5 mm or 7 mm, resulting in 6.5% or 3.75% of the total optical power generated by the µLED (including that which remains trapped within the µLED) reaching the detector. **(c)** µLED optical power (right axis) inferred from measurement (left axes). Both 5 mm and 7 mm distances were evaluated (using the respective scale factors). Shaded error region shows standard deviation over 3 repeated sets of measurements for the same µLED.

From the optical modelling, the fraction of light which reaches the detector at a 5 mm standoff is 0.065 (or 6.5%) (Fig. 4.4b), therefore the optical generated by the µLED can be

inferred (visually shown in Fig. 4.4c):

$$P_{\mu LED} = \frac{P_{measured}}{0.065} \tag{4.4}$$

Measurements were also made at a standoff distance of 7 mm. After applying the respective scale factor of 0.0375 (Fig. 4.4b), the inferred μLED power was identical (Fig. 4.4c, orange curve) demonstrating the robustness of this method and suitability over a range of experimental setups. Care should be taken to position the μLED array centrally over the detector aperture otherwise less power will be collected (Fig. 4.4c, blue curve); this was only an issue if the lateral offset of the array from the centre of the aperture was greater than a few millimetres, therefore the movement of the position of the μLED within the array itself did not affect the amount of power measured.

Utilising both optical models (Fig. 4.4a, 4.1a) and substituting Eq. 4 into Eq. 3 and the result into both Eq. 1 and Eq. 2 yields a calibration from experimentally measured optical power (at the detector) and irradiance (at the cortical surface or μLED surface):

$$L_{cortical\ surface} = \frac{P_{measured}}{0.065} \cdot \frac{scale\ factor \cdot 4}{\pi d^2} \tag{4.5}$$

$$L_{\mu LED\ surface} = \frac{P_{measured}}{0.065} \cdot \frac{scale\ factor}{(40\mu m)^2} \tag{4.6}$$

Since we have optical power measurements at known drive currents, we can infer μLED surface and cortical surface irradiance (both with and without cortical window) as a function of current and plot an L-I curve (Fig. 4.5b). Fig. 4.5a shows current as a function voltage, known as an I-V curve. Fig. 4.5 represents the most efficient μLED array batch (H3); whereas Fig. 4.4c represents a less efficient μLED array (H2), which is why the numbers do not exactly match (see Appendix A.1). The laboratory setup described previously, along with a Python script, allows an I-V and L-I curve to be automatically generated for each μLED across a device.

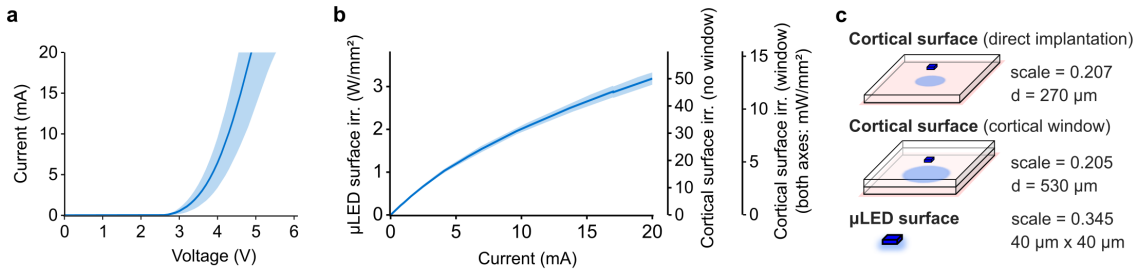


Figure 4.5: **Electrical and optical characteristics of µLED array. (a)** Current-voltage (I-V) curve. **(b)** Irradiance-current (L-I) curves at the µLED surface and cortical surface (with and without a cortical window). Both (a) and (b) show mean and standard deviation over n = 10 µLEDs **(c)** Illustration of different scale factors and illuminated areas which affect irradiance; when a cortical window is used, the illuminated area on the cortical surface increases, hence reducing the irradiance as shown in the rightmost axis in (b).

These results show that a cortical surface irradiance of 10 mW/mm^2 can be achieved with a drive current of < 5 mA, well within the capabilities of our µLED arrays. Referring back to the optically modelled results (Fig. 4.2), activation within layer 2/3 should therefore be possible at similarly low drive currents.

Due to the fabrication process, there is some inherent variability in electrical and optical performance across each array, shown by the error region in the I-V and L-I curves (Fig. 4.5a, 4.5b). This variability is, however, not uniform across the µLED arrays (Fig. 4.6a, 4.6b). It is important to address this, such that a known irradiance is projected to the cortex regardless of the µLED location on the array.

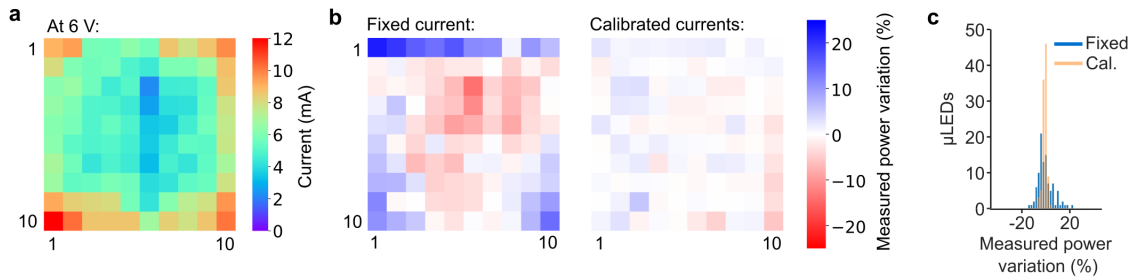


Figure 4.6: **Accounting for variability in µLED performance across devices. (a)** Average measured currents across µLEDs for 4 devices, for fixed voltage. **(b)** Measured optical power for fixed drive current (left). Measured optical power for drive current individually calibrated for each µLED to achieve desired power (right). **(c)** Histogram of measured optical power for both fixed and individually calibrated drive currents across all µLEDs, for same device as (b).

Fig. 4.6a shows the drive current achievable at 6 V averaged across four devices. However, variation in current as a function of voltage does not matter as we use current sources to drive the µLEDs during *in-vivo* experiments (see Chapter 6), so long as the voltage compliance (limit) is not reached. Fig. 4.6b (left) shows the variation in measured optical power, divided

by desired optical power, given the same fixed drive current for each μLED (optimised for a desired optical power/cortical surface irradiance). The μLEDs around the edges are more efficient than the ones in the centre, which is due to variability in the resistance of the tracks connectings the μLEDs. The driver electronics developed for this project (Chapter 6) are capable of driving a chosen current through each individual μLED, so it is possible to correct for this variation by applying a calibration. Quadratic regressions were fitted to the L-I data for each μLED (Fig. 4.5b), with the axes flipped so current was in terms of cortical surface irradiance. The coefficients were saved and imported into the software programme for driving the μLEDs during *in-vivo* experiments. Fig. 4.6b (right) shows the measured optical power (divided by desired optical power) given an individually calibrated drive current for each μLED based on the quadratic regressions.

In summary, utilising optical modelling based on experimentally measured optical data from the μLED array devices, we have shown that the devices can drive high irradiance, capable of penetrating through layer 2/3, and at modest drive currents. Furthermore, we have been able to equalise the irradiance incident on the cortical surface, and hence the neuronal response and behavioural outcomes from stimulating different regions using the μLED array. Advanced driver electronics implement this calibration *in-vivo*. However, there are some limitations to this approach when multiple μLEDs are driven in parallel and the current through each is no longer known. It should be noted that due to challenges with the fabrication, some of the characterisation data provided in this section is based on less efficient μLEDs, for example Fig. 4.4c and Fig. 4.6; for clarity, the data provided in Fig. 4.5a and 4.5b are representative of the best performing devices and should be the main point of reference for electrical and optical performance of our μLED arrays (for more detail, see Appendix A.1). This approach of using optical modelling also provides a benchmark for comparing spatial resolution with that determined through electrophysiology experiments; a more in-depth analysis of this is carried out in Chapter 5. Furthermore, in Chapter 5 optical modelling is utilised to investigate the effect of illuminating multiple μLEDs simultaneously to activate a much broader volume of cortical tissue in the context of behavioural experiments. With these robustly established models, supported through laboratory characterisation, improvements in device engineering to increase spatial resolution can then be explored, as will be discussed in the next section.

4.3 Design and analysis of an optical collimator for enhanced spatial resolution

A challenge with the μLED array over a digital light projector (DLP) approach [11] is the reduction of spatial resolution, due to the Lambertian or cosine dependence of emission from the μLEDs, combined with the standoff distance between source and brain (the sapphire

substrate in our case). Even though the pixel size is 40 μm, the illuminated spot on the cortical surface is 270 μm when the sapphire is in direct contact with the cortical surface (Fig. 4.1b). Within the brain, further spread of light and hence reduction in spatial resolution is caused by the highly scattering tissue medium (Fig. 4.2b). The pixels are spaced 200 μm apart and so to operate with the full fidelity of the device, the illuminated spot diameter should be less than 200 μm and possibly even lower to account for the scattering of light before reaching the target region in layer 2/3.

In this section, optical modelling, and preliminary laboratory measurements, are used to explore improvements in the spatial resolution through the design of an optical interposer. The optical interposer collimates the light by introducing small holes into a thin layer of otherwise opaque material which is adhered in front of the sapphire substrate (concept shown in Fig. 4.7a). The material to fabricate with would likely be silicon due to its strong absorption at the wavelength of the μLEDs (450 nm) [12], relative ease of fabrication and low cost. As an initial prototype an 80 μm thick interposer with 80 μm diameter holes was bonded to the surface of a packaged μLED array using transparent medical-grade epoxy (MED-301-2). Integration of the interposer and hole alignment to pixels was achieved using flip-chip bonding. Emission profiles of devices with and without the interposer were imaged with an SLR camera in 10 μM fluorescein solution (Fig. 4.7b left). To compare the lateral light spread, raw green pixel data was plotted (Fig. 4.7b right, blue lines - error bars computed from different shutter integration times) demonstrating a significant reduction in light spread with the interposer. Both images were compared with Monte Carlo models of equivalent geometry, modelling fluorescence by taking a cut through a 3D detector volume at equivalent distance from the device (Fig. 4.7b right, orange lines).

We have considered two potential fabrication techniques to produce (etch) the holes; deep reactive-ion (DRIE) etching and potassium hydroxide (KOH) etching. DRIE etching can produce vertical structures whereas KOH etching produces a V-shaped structure where the sidewall is at an angle of 54.74° to the horizontal plane [13]. For simplicity in the optical modelling, we have chosen to model the interposer as a perfect absorber and using vertical holes, i.e. those which would be produced using the DRIE etch method. The modified optical model schematic is shown in Fig. 4.8a.

Two different interposer thicknesses were modelled; 100 μm (easier to fabricate) and 50 μm (possible but more challenging to fabricate). For each thickness, three interposer hole diameters were modelled; 100 μm, 150 μm and 200 μm; 200 μm is the absolute upper limit due to the μLED spacing, the practical limit is likely lower. It was confirmed that the illuminated spot sizes on the cortical surface are equivalent to the interposer hole sizes, regardless of interposer thickness (Fig. 4.8b); this area can be used to infer cortical surface irradiance from the power landing on the cortical surface in the model, as described previously. The following

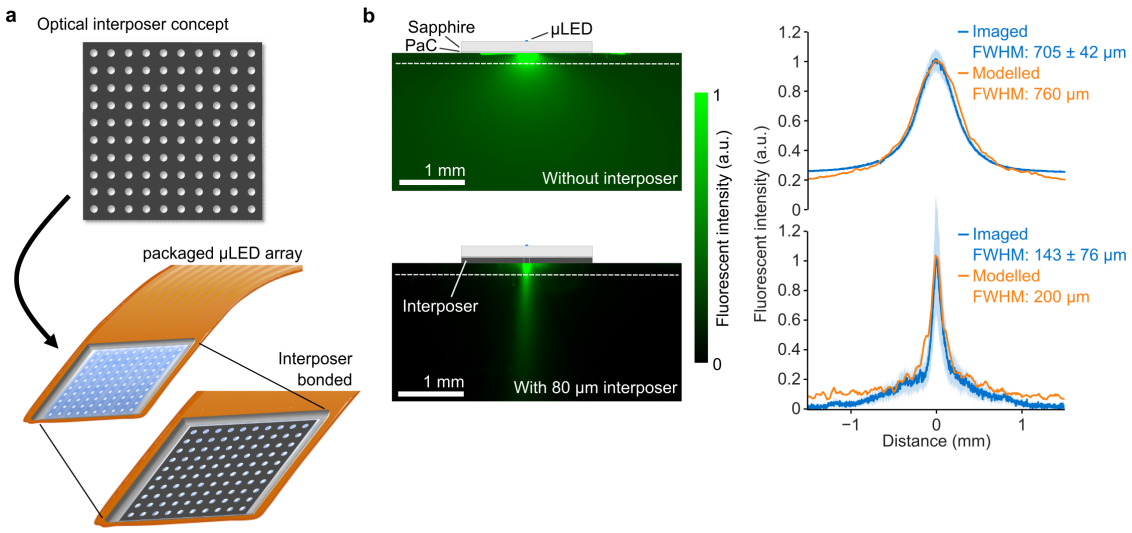


Figure 4.7: **Concept and validation of collimating interposer protoype. (a)** Illustration drawing of optical interposer placed in front of µLED array to collimate light. **(b)** Emission profile of µLED with and without collimating interposer taken with SLR camera in fluorescein; the dashed line corresponds to a depth of 150 µm (left). Emission profile at distance of 150 µm from sapphire surface compared with cuts at equivalent distance from an optical model of fluorescence, with equivalent geometry.

analyses are based on a single µLED and a single hole in the interposer, however in practice, stray light could pass through the adjacent interposer holes. A model incorporating a 3x3 grid of 150 µm diameter holes was evaluated at each interposer thickness (Fig. 4.8c). For the thinner interposer (50 µm), 52% more light power reaches the cortical surface and for the thicker interposer (100 µm), a smaller proportion (35%) more light power reaches the cortical surface. To simplify the following analyses, only the central hole is modelled as this is where most of the light passes through.

Fig. 4.9 summarises the different interposer designs in terms of the spread of light within tissue and the implications on power and drive current. For each interposer, the same amount of optical power has been delivered to the cortical surface, which is similar to the levels used in *in-vivo* electrophysiology experiments (0.29, 0.57, 1.15) mW (discussed in more detail in Chapter 5). These cortical surface power levels correspond to (5, 10, 20) mW/mm^2 of cortical irradiance when the device is implanted directly on the cortical surface without an interposer.

We are interested in the spread of light at some depth within the tissue; in this work, that is layer 2/3 approximately 150 µm from the cortical surface. This spatial spread is evaluated by showing the 1 mW/mm^2 threshold contour (approx. threshold irradiance for activation of ChR2) (Fig. 4.9a). We have chosen to deliver the same optical power to the brain surface for each modelled interposer to provide a direct comparison; this means that the cortical irradiance is different for each interposer design (Fig. 4.9b).

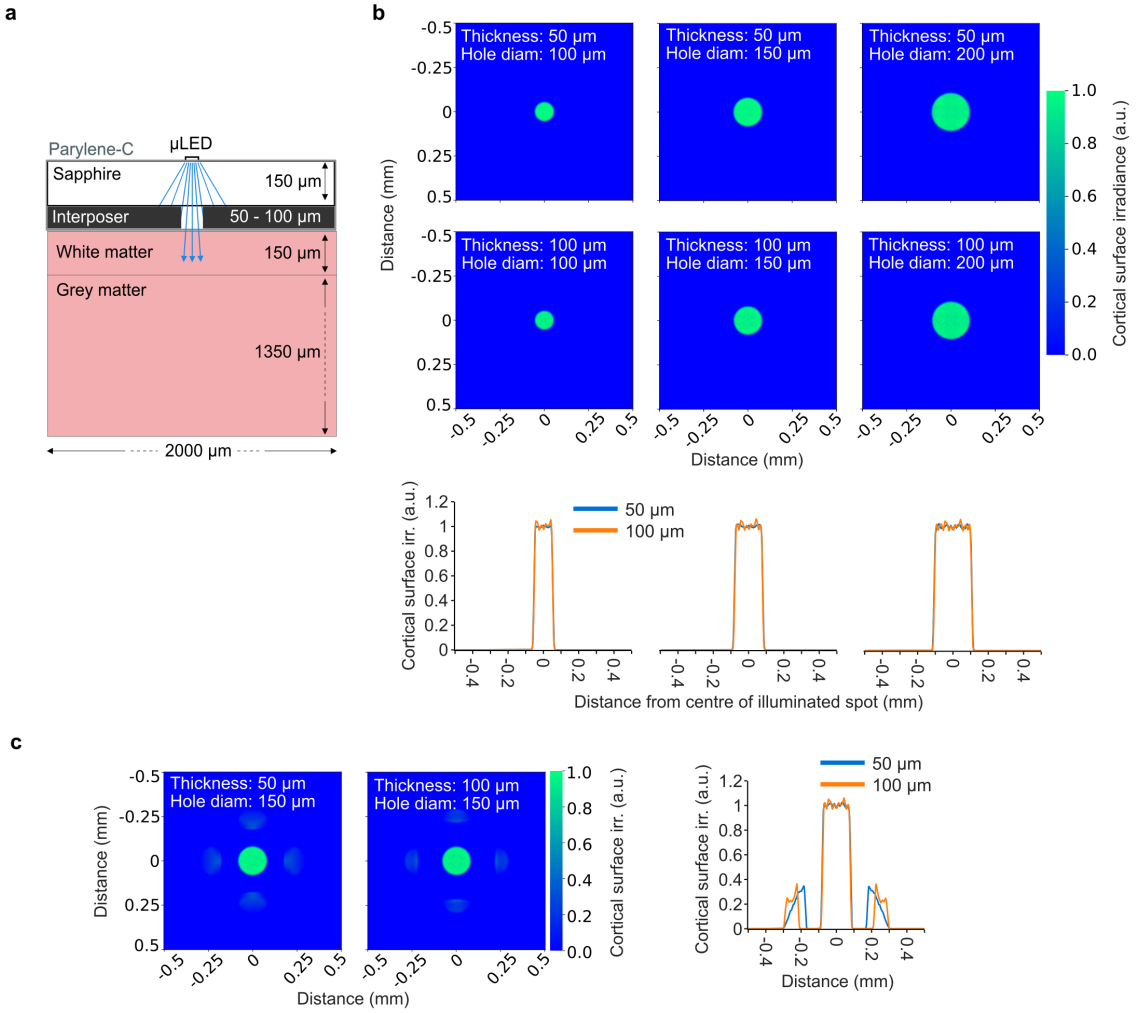


Figure 4.8: **Optical model incorporating collimating interposer. (a)** Annotated schematic of optical model incorporating collimating interposer. **(b)** Illuminated spot profiles from a single µLED over a 1 x 1 mm^2 surface for different interposer hole sizes and thicknesses (top); corresponding 1D cross-sections through centre of illuminated spot profile (bottom). **(c)** Illuminated spot profiles (left) and 1D cross-sections (right) for an optical interposer modelled with a 3x3 grid of holes (single µLED illuminated).

From Fig. 4.9a, we can see that introducing a collimating interposer can reduce the lateral width of the 1 mW/mm^2 contour in layer 2/3 versus without an interposer (Fig. 4.2b), therefore improving resolution. Further analysis of the lateral spread and depth of light penetration within the cortical tissue is evaluated and compared between approaches in the next section. However, increasing the hole diameter does not result in a significant increase in the lateral width of activation, suggesting light scattering is the limiting factor when activating deeper cortical layers. Furthermore, reducing the interposer thickness from 100 µm to 50 µm also has

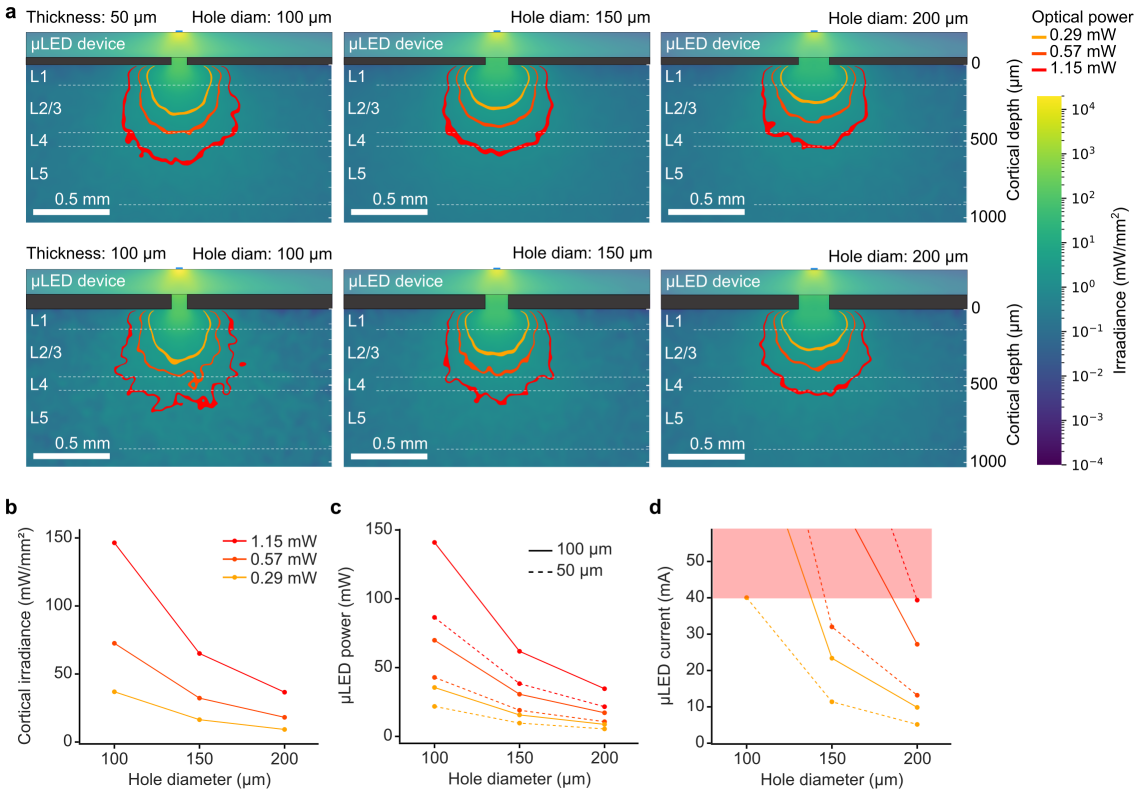


Figure 4.9: **Summary of optical interposer design on light spread within tissue and µLED drive requirements. (a)** Optical profiles corresponding to each interposer design. Coloured contours show 1 mW/mm^2 level given cortical surface powers of (0.29, 0.57, 1.15) mW. **(b)** Equivalent cortical surface irradiance required to drive fixed cortical surface powers based on different interposer hole diameters; interposer thickness does not matter as illuminated spot diameter is the same. **(c)** Equivalent µLED power required to drive fixed cortical surface powers based on different interposer hole diameters, and interposer thicknesses. **(d)** Equivalent drive current to achieve µLED power levels in (c); red shaded region represents thermal rollover region of µLEDs.

minimal impact on this spatial spread. These are important considerations as having too small a hole diameter, or too thick an interposer, results in excessive loss of light requiring µLEDs to be driven at significantly higher power (Fig. 4.9c, 4.9d).

4.4 Comparison of Lambertian emitting µLEDs versus collimated digital light projector approach

Digital light projectors (DLPs) – sometimes known as digital micromirror devices (DMDs) – play a key role in allowing patterned optical illumination of the cortical surface in the context of optogenetic stimulation [11, 14, 15]. In this thesis, an implantable approach has been explored using a µLED array. However, natural questions arise on how this compares

to the fairly ubiquitous DLP approach, in terms of the spatial resolution possible. The key difference between the µLEDs and a DLP system is that µLEDs emit light with a Lambertian spread, exhibiting a cosine dependence, whereas DLPs with focussing optics can project collimated light. With the µLED array, the standoff due to the sapphire substrate reduces the spatial resolution by increasing the illuminated spot size on the cortical surface as discussed previously. In this section, the DLP and implantable µLED array approaches are compared, and the interposer described previously is evaluated in terms of its comparability to the DLP as a collimator.

The optical model used before was modified to include a collimated light source and cortical window (Fig. 4.10a); the cortical window is required as DLP-based experiments are acute and head-fixed.

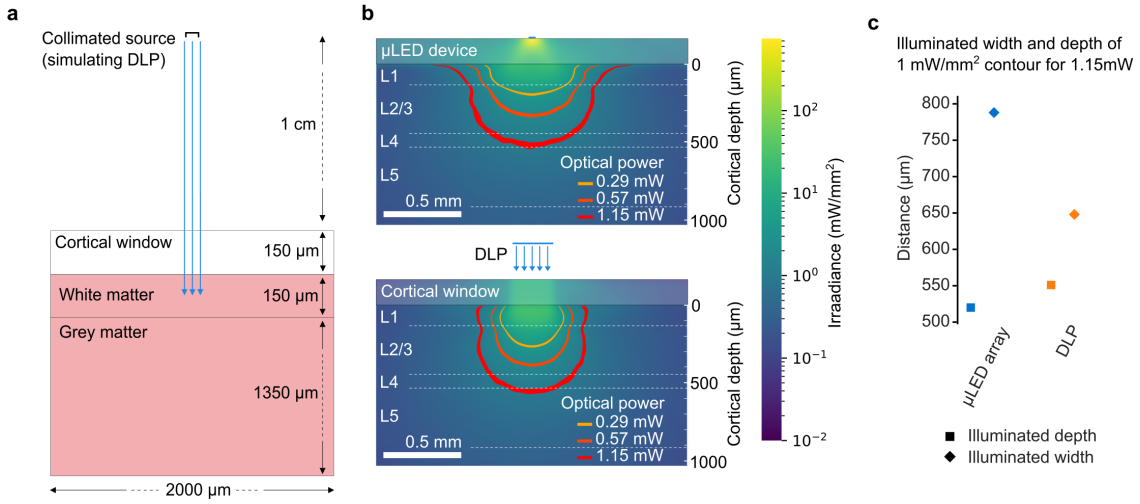


Figure 4.10: **Comparison of light spread within tissue for collimated light source (DLP) versus µLED array. (a)** Annotated schematic of optical model for DLP. **(b)** Optical profiles corresponding to µLED array (top) and DLP (bottom). Coloured contours show 1 mW/mm^2 level given cortical surface powers of (0.29, 0.57, 1.15) mW. **(c)** Modelled comparison of spatial resolution and depth of penetration of 1 mW/mm^2 contour of µLED array versus DLP from optical models in (b); spatial extent of activation determined from the width of the 1 mW/mm^2 threshold contour at a depth of 150 µm.

Firstly, the µLED array was compared to the DLP directly by projecting (5, 10, 20) mW/mm^2 onto a 270 µm diameter area – equivalent to (0.29, 0.57, 1.15) mW of optical power delivered to cortical surface (Fig. 4.10b). The depth and width (at layer 2/3, 150 µm depth) of the 1 mW/mm^2 contour corresponding to the highest optical power (1.15 mW) were compared. These show that using a collimated DLP source results in a significantly reduced lateral spread of light within layer 2/3, from ∼790 µm to ∼650 µm while maintaining similar depth of light penetration (Fig. 4.10c).

A comparison was then made between the spread of light in the brain from a true collimated

source (DLP) and the µLED array with collimating interposer. Although we describe the interposer as "collimating", this is not strictly true as light can still pass through the hole at an angle. The DLP model was run for each of three illuminated spot sizes, corresponding to the previously evaluated interposer designs (Fig. 4.11a). These distributions seem broadly similar to that using the interposer, however the illuminated region is generally slightly wider (Fig. 4.11b). Although, the depth of illumination from the DLP is also slightly larger suggesting that reducing the power with the DLP slightly would bring the widths more into alignment.

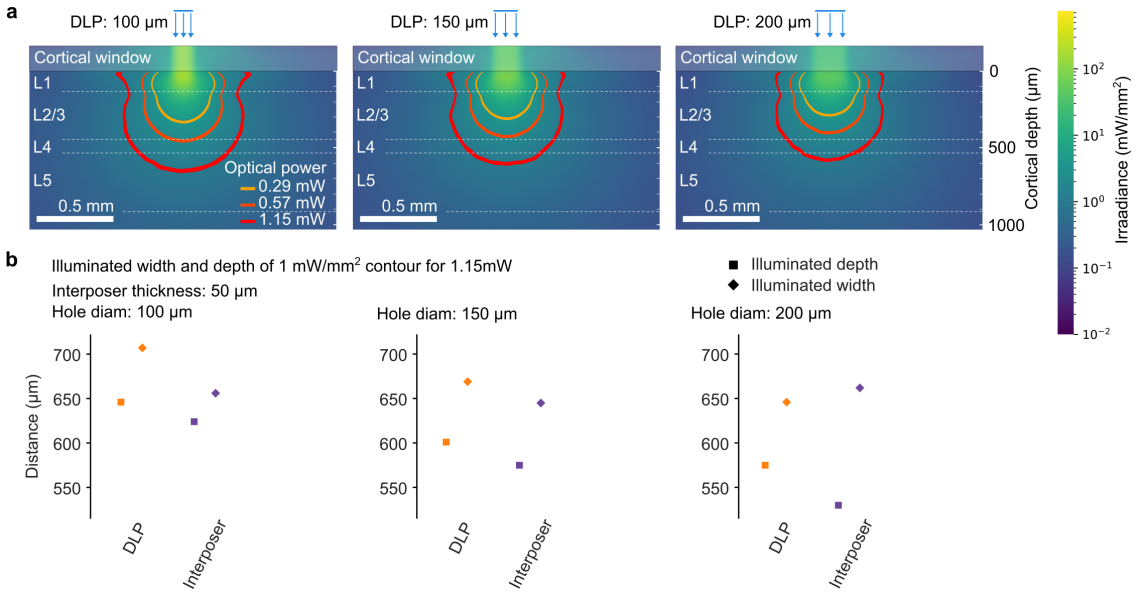


Figure 4.11: **Comparison of light spread within tissue for collimated light source (DLP) versus µLED array with optical interposer. (a)** Optical profiles for DLP projecting collimated light onto cortical surface at diameters equivalent to previously characterised optical interposer holes. **(b)** Modelled comparison of spatial resolution and depth of penetration of 1 mW/mm^2 contour of DLP versus µLED array with optical interposer; as before, these are based on the 1 mW/mm^2 contour corresponding to cortical surface power of 1.15 mW.

The µLED array with optical interposer provides a light distribution which is similar to the collimated DLP source (Fig. 4.11b). However, if no interposer is used, then lateral spread of light is significantly larger when using the µLED array (Fig. 4.10b). Furthermore, the interposer hole diameter has minimal effect at the target depth, suggesting a device with up to 200 µm diameter holes would be the best candidate for an improved version of the device (Fig. 4.9a). Ideally the thinner design of 50 µm would be used due to the significantly reduced drive current requirements (Fig. 4.9b). If this is not possible, then using a more sensitive opsin than ChR2, for example Chronos [16] in combination with the optical interposer may be the best approach. This modelling shows that the Lambertian emission from the µLEDs is the primary cause for reduction in spatial resolution beyond the superficial cortical layers, however that

after collimating the light, scattering becomes the dominant limiting factor. This analysis confirms that it is possible to achieve similar spatial resolution to current optogenetic studies, that use DLPs, using the µLED arrays. Coupled with low drive currents, this makes the devices a promising candidate for advanced studies into cortical function in behaving animals.

4.5 Thermal characterisation and modelling

Heating of the brain is known to perturb neuronal activity [17, 18] and potentially damage tissue and cells. LEDs are actively powered devices which generate light from electrical power but dissipate most of their power as heat. Therefore, it is of particular importance to study thermal effects as the µLED array is implanted directly on the cortical surface. Ideally, the temperature increase would remain within the commonly quoted limit of 1 – 2 °C [19]. Our device geometry and custom material stack results in complex heat flows, necessitating an in-depth thermal evaluation [20]. To thermally characterise our µLED arrays, infrared (IR) imaging experiments were conducted in conjunction with a thermal model of the device chronically implanted in the mouse cortex.

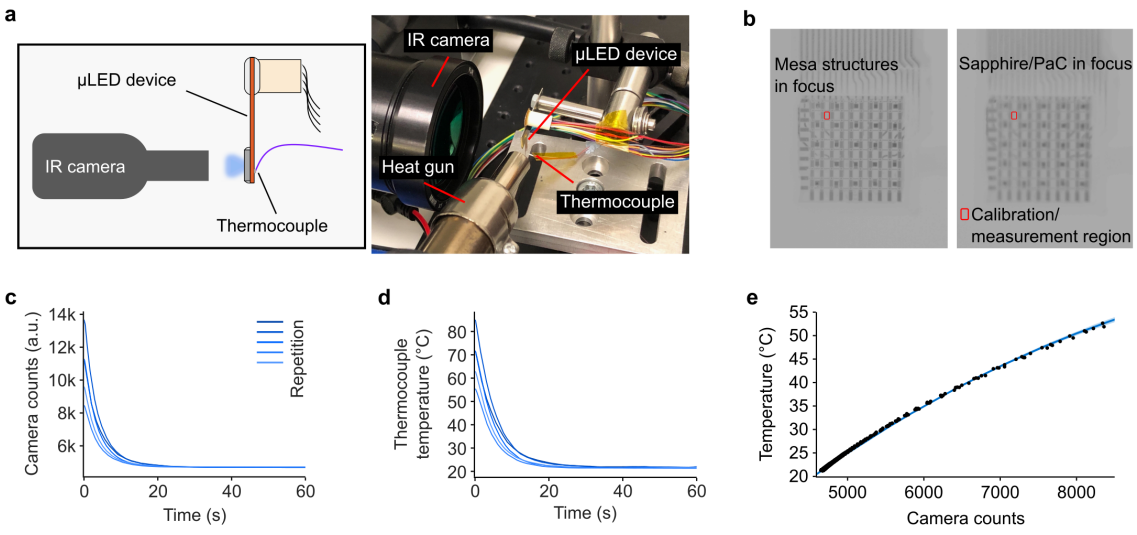


Figure 4.12: **Infrared (IR) imaging experiment for validation of thermal models. (a)** Schematic of experimental setup (left) and photograph (right). **(b)** IR image showing locations used for thermal measurement and calibration, taken as an average within rectangular area. **(c)** Camera count measurements over first 60 seconds of calibration experiment. **(d)** Thermocouple temperature measurements over first 60 seconds of calibration experiment. **(e)** Quadratic regression fitted to thermocouple data plotted against camera counts, for calibration. 5 sets of measurements were taken over a 5 minute period to produce calibration curves and the average taken.

The setup of the IR imaging experiment is shown in Fig. 4.12a, with the IR camera (FLIR SC7000-Series) focussed on the sapphire surface of the µLED array, packaged on the flexible

device. An integration time of 1.5 ms was used to ensure the camera collected enough photons to operate within its linear regime, with a refresh rate of 100 Hz for sufficient temporal resolution. Since the camera returns a digital value, proportional to the number of detected IR photons, a calibration is necessary to convert this value into temperature in °C. The calibration uses an E-type thermocouple which is itself calibrated using cold junction compensation. The thermocouple is fixed to the back surface of the flexible device using thermal compound. The sapphire surface of the device is where we want to measure the temperature as it is in direct contact with the brain tissue. To bring the sapphire surface into focus, firstly the mesa structures are brought into focus as the features are distinct, then the focus is moved back through the known thickness of the sapphire (150 μm) by using the micrometer screw of the stage the device is mounted on (Fig. 4.12b).

To perform calibration, the device and thermocouple were heated simultaneously and uniformly, using a heat gun, to just over 50 °C, and allowed to cool down to room temperature. Temperature measurements from the thermocouple were stored and, simultaneously, camera counts were measured from several areas located near to the μLEDs being tested in the sapphire area; both sets of measurements were taken at an interval of 100 ms (Fig. 4.12c, 4.12d). This process was repeated 5 times, and a second order polynomial fitted between the thermocouple measurements to camera counts (Fig. 4.12e). This allowed us to infer the actual temperature at the surface of the sapphire which would be in contact with brain tissue, with μLEDs illuminated.

Thermal modelling was then carried out using the COMSOL Multiphysics 5.6 heat transfer module. For simplicity, 2D rotationally symmetric models were developed and the volumes of each material calculated such that the resulting 3D model is representative. 2D models were used as they are simpler to develop and adapt, and much faster to simulate than full 3D models. The model of the device implements a layer of gallium nitride (GaN) atop sapphire with the parylene-C encapsulation. The flexible device is implemented as polyimide with a layer of copper on each side. Between the device and PCB, there are gold bump bonds and an epoxy underfill. The GaN μLED was implemented as a heat source. Two models were developed, one to simulate the thermal imaging experiment to verify that the model of the device itself is accurate (Fig. 4.13a) and another to simulate the chronic implantation (Fig. 4.13b).

Firstly, to compare the model in Fig. 4.13a to the results from the IR imaging experiment, the thermal power generated by the μLED was experimentally measured. This was done by measuring the electrical power (current times voltage) and optical power from the same device (and tested μLEDs) that was thermally imaged; this is shown in Fig. 4.14a under a sweep of drive currents. By subtracting the optical power from electrical power, the resulting thermal power as a function of drive current can be determined and used as an input to the model (Fig. 4.14b).

Temperature was evaluated under several stimulation protocols and μLED drive currents

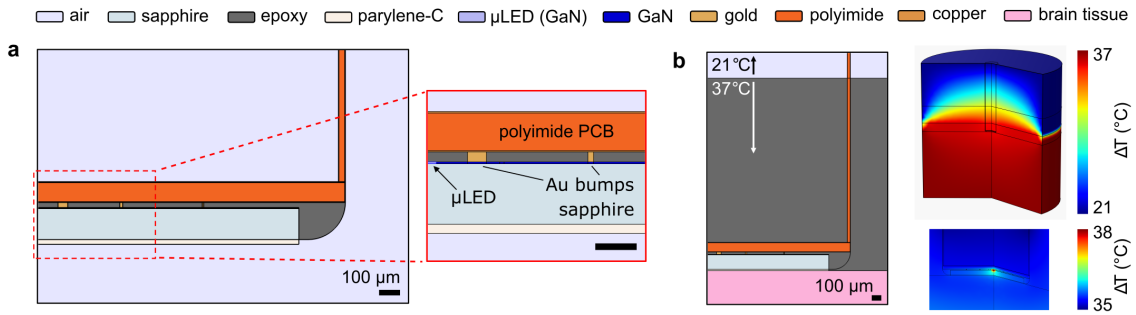


Figure 4.13: **Mathematical models developed in COMSOL for thermal analysis of µLED array. (a)** Schematic of 2D rotationally symmetric thermal model of flexible µLED device suspended in air, for comparison with IR imaging experiment. **(b)** Schematic of thermal model of flexible µLED device in chronic implantation (left). Screenshot of simulated model after 3D rotation after rotation about the axis of symmetry (right).

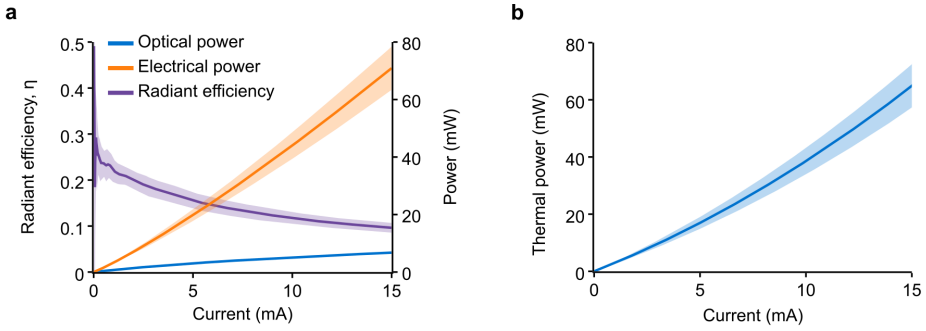


Figure 4.14: **Thermal characterisation of µLED array. (a)** Measured electrical and optical power (actual optical power generated within µLED was determined through optical models) and radiant efficiency as a function of drive current. **(b)** Thermal power as function of µLED drive current. Both (a) and (b) show mean and standard deviation of n = 10 µLEDs from representative device.

representative of those being used in *in-vivo* experiments. To verify the thermal model of the device and assess the time constant of the thermal drop-off, a simple protocol was used; single µLED pulsed at 5 mA (1 Hz, 50% duty cycle). The thermal drop-off refers to the time after the µLED is switched off, that the temperature recovers to its previous level. This is an important consideration to avoid a pulse pile-up effect whereby the temperature increases stack, potentially resulting in unwanted thermal effects on tissue. Thermal evaluation of the sapphire surface showed the peak temperature and time constant were in good agreement (Fig. 4.15a) between the model in Fig. 4.13a and thermal imaging experiment for a 5 mA drive current. The IR image in Fig. 4.15b shows a frame corresponding to the peak temperature from the IR camera calibrated using the thermocouple measurements. The calibration was carried out using a measurement from the sapphire area of the image (Fig. 4.12b) with the remaining image scaled accordingly, however, this assumes linear emissivity between the

materials and so is only indicative. The temperature gradient shown in Fig. 4.15b (left) shows the heat originating from the tested µLED (see Fig. 4.15a inset), and additionally demonstrates that the device packaging is able to dissipate this heat, for example through the electrical tracks. There was also good agreement when a faster pulse rate (20 Hz, 50% duty cycle, 10 pulses) was used (Fig. 4.15c). To model multiple µLEDs being illuminated simultaneously, a concentric ring was used with a radius similar to the distance between the rows of µLEDs. The volume of this ring is equivalent to the volume of 3 µLEDs (2x2 case) or 8 µLEDs (3x3 case), with the centre cylinder having a volume equivalent to a single µLED as before; this geometry choice results from the rotational symmetry constraint of the model. These were additionally found to agree well under drive currents per µLED of 3 mA and 1 mA respectively (Fig. 4.15d).

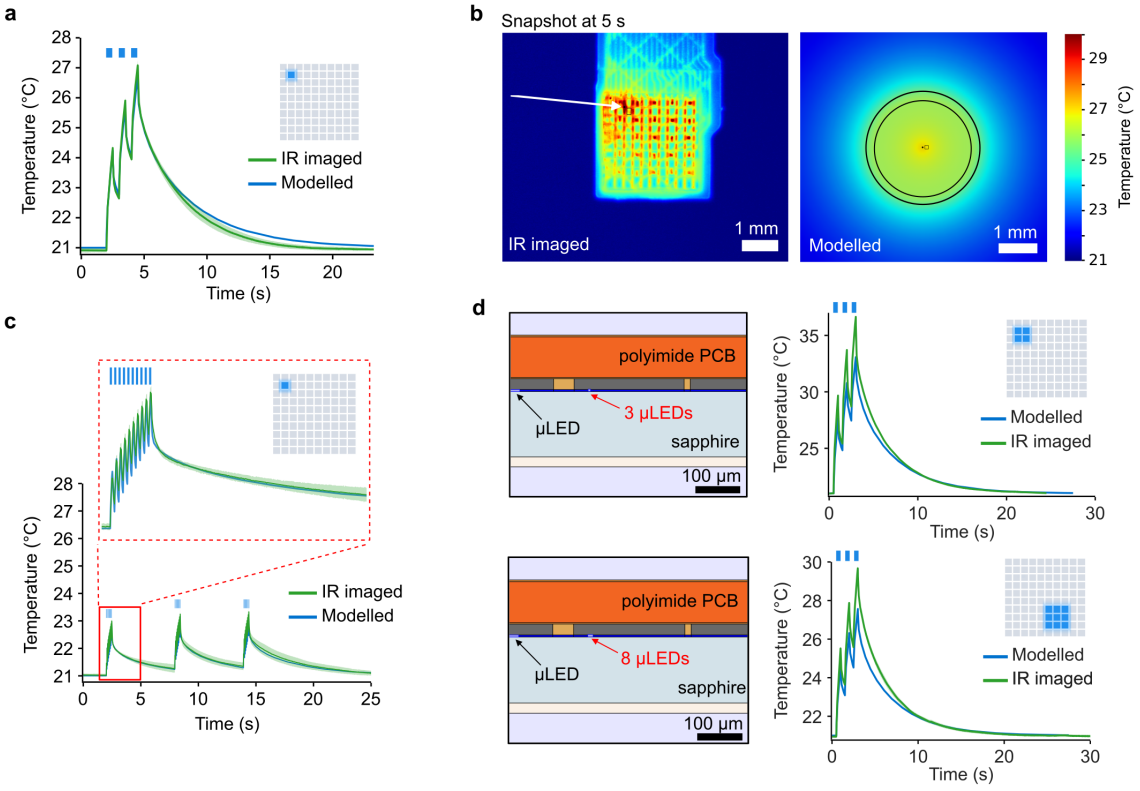


Figure 4.15: **Verification of thermal model through IR imaging. (a)** Comparison of IR imaging with equivalent thermal model; protocol is 1 Hz, 50% duty cycle, 5 mA drive current for a single µLED. **(b)** Full thermal images corresponding to (a) showing temperature across the surface in contact with the brain in both IR-imaged and modelled cases. Black dot indicates µLED and square outline indicates area averaged to evaluate temperature. **(c)** Single µLED comparison as in (a) but using faster pulses (20 Hz, 50% duty cycle, 5 mA drive current). **(d)** Same comparison as in (a) but for 2x2 square of illuminated µLEDs, with additional µLEDs modelled as a concentric ring (3 mA drive current for each µLED) (top). 3x3 case (1 mA drive current for each µLED) (bottom).

These results gave confidence that the model represented the actual device well, and that

the model of the chronic implantation (Fig. 4.13b) could be used to inform drive current limits for chronic *in-vivo* experiments. This models the device over a volume of brain tissue (water) with the encapsulating cement atop. The air above the device is at 21 °C and the brain is 37 °C at initial conditions, which in the model reach equilibrium after about 30 seconds. The temperature corresponding to a representative stimulation protocol of 3 mA and 5 mA per µLED (20 Hz, 50% duty cycle) was input as the heat source in the model (Fig. 4.16a).

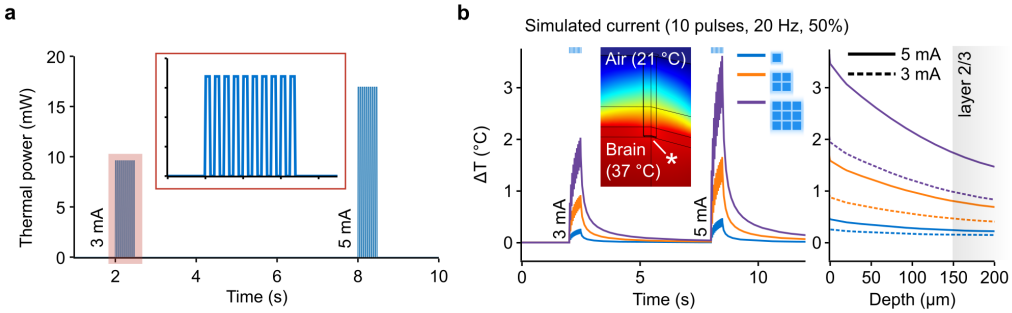


Figure 4.16: **Modelled temperature increase in the brain under pulsed stimulation from µLED array. (a)** Thermal pulse protocol input to thermal model for a single µLED. **(b)** Modelled temperature increase on brain surface (left); modelled heat protocol is equivalent to that used in *in-vivo* experiments, with simulated thermal power equivalent to a drive current of 3 mA and 5 mA (per µLED). Peak temperature as a function of cortical depth (right).

The temperature at the contact surface of the device with the brain is shown in Fig. 4.16b and for a single µLED driven at 5 mA, the modelled peak temperature increase remained well below 1 °C. For 2x2 and 3x3 patterns, the peak temperature increase can exceed 1 °C, thus the drive current must be selected carefully. As an example, driving a 3x3 pattern at 3 mA per µLED results in a temperature increase of ∼2 °C on the brain surface and ∼1 °C in layer 2/3. Despite being minimal, temperature increases stress the need for control experiments to rule out thermal effects influencing neuronal or behavioural responses. *In-vivo* electrophysiology recordings under stimulation with these devices suggest responses can be driven at even lower drive currents (Chapter 5). These results are very positive for the implementation of µLED cortical implants based on GaN-on-sapphire technology. This thermal model further provides a framework for evaluating any desired stimulation protocol, for example changing the duty cycle, pulse frequency or drive current in the context of optogenetic experiments utilising opsins with different kinetics or activation thresholds.

Fig. 4.17a shows the peak temperature increase over 500 ms of pulsed stimulation for different pulse repetition rates and duty cycles. This confirms duty cycle is the most dominant factor influencing thermal effects, which is logical as duty cycle dictates the actual time the µLED is active over the 500 ms time window in this case. Examination of the temperature increase over a range of drive currents demonstrates at the drive current required to achieve

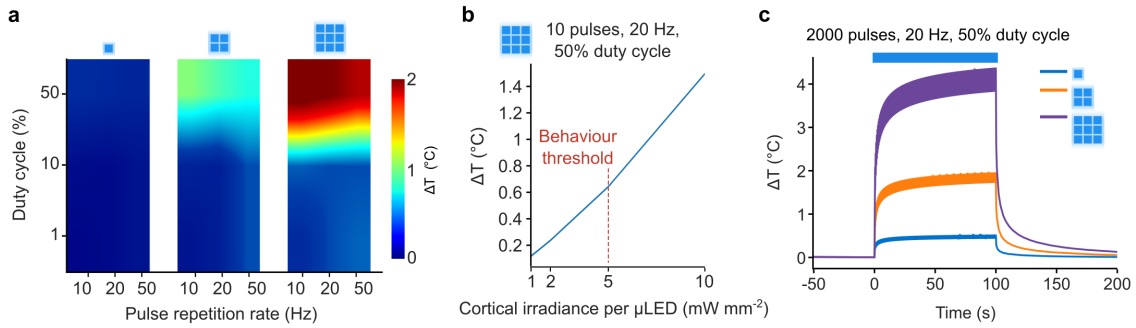


Figure 4.17: **Evaluation of temperature increase under different stimulation protocols. (a)** Evaluation of modelled peak temperature increase on brain surface, over 500 ms stimulation time, across different pulse repetition rates and duty cycles; interpolation is used on colour map between 9 data points per pattern. Modelled thermal power corresponds to 3 mA equivalent per µLED. **(b)** Modelled peak temperature increase as a function of cortical irradiance; equivalent currents per µLED are (0.19, 0.38, 1.02, 2.26) mA. **(c)** Modelled temperature increase on brain surface over prolonged operation (100 seconds); modelled heat protocol (20 Hz, 50% duty cycle), with simulated thermal power equivalent to a drive current of 3 mA per µLED.

threshold of activation for behaviour, remains below 1 °C (Fig. 4.17b). Lastly, prolonged continuous pulsing of the µLEDs beyond a 500 ms time window (for example, 100 s) results in a significantly higher temperature increase.

4.6 Discussion

The characterisation and modelling carried out in this chapter has enabled a clearer understanding of the operating conditions of the devices. This has been imperative in enabling a series of successful in *in-vivo* electrophysiology and behavioural experiments. Developing an operational framework for implantable optogenetic devices based on their optical and thermal properties, therefore, should be used to inform potential *in-vivo* experiments such as those detailed next in Chapter 5. From optical modelling, we determined a limitation in the spatial resolution of the devices, particularly for targeting deeper structures in the cortex (explained in Chapter 2), due to the scattering of light within the brain tissue. This could be improved through the use of an interposer to collimate the light, although scattering still ultimately limited the spatial resolution. Furthermore, the use of the interposer would require higher drive currents (factor of 4-5 times) to deliver above-threshold irradiance to an equivalent depth as the non-interposer case. The use of more sensitive opsins, for example Chronos [16] could help overcome this limitation. The fabrication of such interposer devices, using a DRIE method, and the subsequent integration to the µLED array devices as discussed in Chapter 3, would be an interesting future direction for the project. Further studies could investigate devices such as micro-lenses to improve spatial resolution, limiting the effect of optical losses in the

interposer [21]. Due to project time constraints, only the interposer was investigated here as it would be relatively simple to fabricate and integrate to the existing chronic devices. In particular, evaluating and comparing the behavioural response of animals under stimulation with a device with enhanced spatial resolution, could offer insight into the effectiveness of such approaches. Lastly, computational optogenetics as a complementary approach could be used to fine-tune the stimulation parameters by providing an understanding of optical activation at a cell and circuit level [22]. This could further motivate the approach of using more sensitive [16], or soma-targeted [23], opsins to improve information transfer capability of our devices. In the upcoming chapter, the implications of using opsins that also make the dendrites of the neurons sensitive to optogenetic stimulation are explored. Computational tools could further elucidate this stimulation paradigm [24].

Bibliography

1. Scharf, R. *et al.* Depth-Specific Optogenetic Control In Vivo with a Scalable, High-Density μLED Neural Probe. *Scientific Reports* **6,** 28381. doi:10.1038/srep28381 (2016).
2. McAlinden, N. *et al.* In Vivo Optogenetics Using a Utah Optrode Array with Enhanced Light Output and Spatial Selectivity. *Journal of Neural Engineering* **21,** 046051. doi:10.1088/1741-2552/ad69c3 (2024).
3. Greer, R. *et al.* Chronically implantable LED arrays for optogenetic cortical surface stimulation in mice. en. *Nature Communications* **17,** 878. ISSN: 2041-1723. doi:10.1038/s41467-025-68191-5. (2026) (Jan. 2026).
4. McAlinden, N., Gu, E., Dawson, M. D., Sakata, S. & Mathieson, K. Optogenetic activation of neocortical neurons in vivo with a sapphire-based micro-scale LED probe. English. *Frontiers in Neural Circuits* **9.** ISSN: 1662-5110. doi:10.3389/fncir.2015.00025. (2026) (May 2015).
5. Aravanis, A. M. *et al.* An optical neural interface: in vivo control of rodent motor cortex with integrated fiberoptic and optogenetic technology. en. *Journal of Neural Engineering* **4,** S143. ISSN: 1741-2552. doi:10.1088/1741-2560/4/3/S02. (2025) (June 2007).
6. Yaroslavsky, A. N. *et al.* Optical properties of selected native and coagulated human brain tissues in vitro in the visible and near infrared spectral range. eng. *Physics in Medicine and Biology* **47,** 2059–2073. ISSN: 0031-9155. doi:10.1088/0031-9155/47/12/305 (June 2002).
7. Yona, G., Meitav, N., Kahn, I. & Shoham, S. Realistic Numerical and Analytical Modeling of Light Scattering in Brain Tissue for Optogenetic Applications. en. *eNeuro* **3.** ISSN: 2373-2822. doi:10.1523/ENEURO.0059-15.2015. (2025) (Jan. 2016).
8. Wang, Q. *et al.* The Allen Mouse Brain Common Coordinate Framework: A 3D Reference Atlas. *Cell* **181,** 936–953.e20. ISSN: 0092-8674. doi:10.1016/j.cell.2020.04.007. (2025) (May 2020).

9. Mota, B. *et al.* White matter volume and white/gray matter ratio in mammalian species as a consequence of the universal scaling of cortical folding. eng. *Proceedings of the National Academy of Sciences of the United States of America* **116,** 15253–15261. ISSN: 1091-6490. doi:10.1073/pnas.1716956116 (July 2019).

10. Claudi, F. *et al.* Visualizing anatomically registered data with brainrender. *eLife* **10** (eds Mathis, M. W., Wassum, K. M. & Nunez-Iglesias, J.) e65751. ISSN: 2050-084X. doi:10.7554/eLife.65751. (2025) (Mar. 2021).

11. Ceballo, S., Piwkowska, Z., Bourg, J., Daret, A. & Bathellier, B. Targeted Cortical Manipulation of Auditory Perception. eng. *Neuron* **104,** 1168–1179.e5. ISSN: 1097-4199. doi:10.1016/j.neuron.2019.09.043 (Dec. 2019).

12. Green, M. A. & Keevers, M. J. Optical properties of intrinsic silicon at 300 K. en. *Progress in Photovoltaics: Research and Applications* **3,** 189–192. ISSN: 1099-159X. doi:10.1002/pip.4670030303. (2025) (1995).

13. Yun, S.-S., You, S.-K. & Lee, J.-H. Fabrication of vertical optical plane using DRIE and KOH crystalline etching of (1 1 0) silicon wafer. *Sensors and Actuators A: Physical* **128,** 387–394. ISSN: 0924-4247. doi:10.1016/j.sna.2006.02.015. (2025) (Apr. 2006).

14. Abbasi, A. *et al.* Brain-machine interface learning is facilitated by specific patterning of distributed cortical feedback. *Science Advances* **9,** eadh1328. doi:10.1126/sciadv.adh1328. (2026) (Sept. 2023).

15. Chong, E. *et al.* Manipulating synthetic optogenetic odors reveals the coding logic of olfactory perception. *Science* **368,** eaba2357. doi:10.1126/science.aba2357. (2025) (June 2020).

16. Klapoetke, N. C. *et al.* Independent Optical Excitation of Distinct Neural Populations. *Nature methods* **11,** 338–346. ISSN: 1548-7091. doi:10.1038/nmeth.2836. (2025) (Mar. 2014).

17. Kim, T. *et al.* Thermal effects on neurons during stimulation of the brain. en. *Journal of Neural Engineering* **19,** 056029. ISSN: 1741-2552. doi:10.1088/1741-2552/ac9339. (2025) (Oct. 2022).

18. Owen, S. F., Liu, M. H. & Kreitzer, A. C. Thermal constraints on in vivo optogenetic manipulations. en. *Nature Neuroscience* **22,** 1061–1065. ISSN: 1546-1726. doi:10.1038/s41593-019-0422-3. (2026) (July 2019).

19. Acharya, A. R. *et al.* In vivo blue light illumination for optogenetic inhibition: effect on local temperature and excitability of the rat hippocampus. en. *Journal of Neural Engineering* **18,** 066038. ISSN: 1741-2552. doi:10.1088/1741-2552/ac3ef4. (2025) (Dec. 2021).

20. McAlinden, N. *et al.* Thermal and optical characterization of micro-LED probes for in vivo optogenetic neural stimulation. EN. *Optics Letters* **38,** 992–994. ISSN: 1539-4794. doi:10.1364/OL.38.000992. (2025) (Mar. 2013).

21. Bi, X. *et al.* A Flexible, Micro-Lens-Coupled LED Stimulator for Optical Neuromodulation. *IEEE Transactions on Biomedical Circuits and Systems* **10,** 972–978. doi:10.1109/TBCAS.2016.2599406 (2016).

22. Nikolic, K., Jarvis, S., Grossman, N. & Schultz, S. *Computational models of optogenetic tools for controlling neural circuits with light* in *2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)* (2013), 1290–1293. doi:10.1109/EMBC.2013.6610903.

23. Mardinly, A. R. *et al.* Precise multimodal optical control of neural ensemble activity. en. *Nature Neuroscience* **21,** 881–893. ISSN: 1546-1726. doi:10.1038/s41593-018-0139-8. (2026) (June 2018).

24. Jarvis, S., Nikolic, K. & Schultz, S. R. Neuronal gain modulability is determined by dendritic morphology: A computational optogenetic study. *PLoS Computational Biology* **14,** e1006027. doi:10.1371/journal.pcbi.1006027. https://doi.org/10.1371/journal.pcbi.1006027 (2018).

Chapter 5

Device validation through *in-vivo* experimentation in mice

This chapter details two sets of *in-vivo* experiments utilising the implantable µLED array devices developed and characterised in the previous two chapters. Electrophysiology recordings were carried out from the auditory cortex of awake, head-fixed mice under simultaneous optogenetic stimulation from the µLED arrays, packaged in the acute device configuration (as described in Chapter 3). These were used to establish the stimulation parameters needed for reliable optogenetic activation of neuronal ensembles within the mouse auditory cortex. Spatial resolution was examined from electrophysiology recordings and compared with the optical models previously developed in Chapter 4. µLED arrays were then chronically implanted (using the flexible, chronically implantable packaging) in several mice and behavioural discrimination experiments were carried out by training mice to respond to patterned optogenetic stimulation of distinct sub-regions of the auditory cortex. These experiments confirmed our ability to transfer information to neuronal circuits using the µLED arrays in both head-fixed and freely-behaving mice. This chapter concludes by examining current and promising directions for *in-vivo* experimentation using our devices, with a focus on gaining a clearer understanding of the resolution with which information can be transferred, and the potential for implementing encoding models for complex information transfer to the cortex, driving clinical implications. This follows into Chapter 6 where new hardware has been developed, specifically tailored for implementing encoding models and closed-loop control of the µLED arrays for advanced *in-vivo* experimentation. All *in-vivo* experiments described here were carried out by the Bathellier lab, including surgery and data acquisition (Antonin Verdier, Ella Callas, Alicia Aniorte, Eya Mabrouk, Magdalena Pereyra; see Acknowledgements for details). Data analyses were carried out between the two labs; all analyses shown here were carried out by the author unless specified otherwise. Every experiment involving animals has been carried out following a protocol approved by an ethical commission, in accordance with the French Ethical Committee

(Direction Générale de la Recherche et de l'Innovation) and European legislation (2010/63/EU). Procedures were approved by the French Ministry of Education in Research after consultation with the ethical committee #59 (authorization number APAFIS#28757-2020122111048873). The in-vivo experimental results, both electrophysiology and behaviour, and histology, have been published in [1].

Throughout electrophysiology and head-fixed behavioural experiments, the Emx-IRES-Cre transgenic mouse strain was used [2]. This was crossed with the Ai27d optogenetic variant [3] with the resulting mice expressing ChR2 in all pyramidal cells of the pallium (Emx-IRES-Cre x Ai27d). Control mice were used to ensure recorded neuronal or behavioural responses were not the result of thermal effects or an electrical artefact; these mice were not crossed with the optogenetic strain so neurons would not be sensitive to the blue light. For freely-behaving experiments only, mice were instead transfected with AAV (adeno-associated virus) vectors encoding ChRmine under the control of the CaMKIIα promoter which were injected to drive expression in pyramidal neurons of the auditory cortex (pAAV-CamKIIα-ChRmine-mScarlet-Kv2.1-WPRE (AAV8). Addgene number: 130991). ChRmine's activation spectrum is red-shifted but it is broad with less than 50% reduction of elicited current at the wavelength of the µLED arrays (λ = 450 nm) [4]. Since ChRmine has much higher photocurrents than ChR2, blue light illumination was sufficient to elicit robust enough neuronal activation for behavior. The implications of this are described in more detail later alongside the results of the freely-behaving experiments. As described in previous chapters, we intend to activate the pyramidal neurons present in layer 5 of the auditory cortex, through optogenetic stimulation of their apical dendrites which extend up to layer 2/3 [5]. However, it should be noted that other secondary network effects could be contributing to measured neuronal or behavioural responses.

5.1 Acute *in-vivo* electrophysiology experiments

The purpose of the experimentation and subsequent analysis of recorded electrophysiology data undertaken in this section was to establish the stimulation threshold and spatial resolution of the µLED arrays for activating neuronal ensembles within the auditory cortex (AC). Stimulation, or activation, thresholds represent the illumination level required for reliable activation of the target neurons, and are critically important to establish as they inform the required device drive currents. This in turn has implications on drive electronics, and whether wireless approaches could be explored [6, 7] as well as thermal effects [8, 9], as was discussed in Chapter 4.

5.1.1 Experimental setup

For this acute electrophysiology experiment, mice were prepared with a 150 µm thin transparent cortical window of approximately 5 mm diameter over one AC with the head-fixation post positioned at the opposite side. For the electrophysiology experiment, a grounding electrode was implanted contralaterally (Fig. 5.1a, 5.1b). Mice were co-housed and received food and water ad-libitum for the duration of experimentation. Immediately prior to recording sessions, a small hole (approx. 0.5 mm diameter) was drilled into the cortical window using a diamond drill bit. As discussed previously, all electrophysiology experiments were conducted in an acute preparation, with head-fixation used to secure the brain in a stationary position. Micro manipulators were used to firstly position the acute device over the cortical window, then a 64-channel NeuroNexus linear recording electrode array (NeuroNexus A1x64 Poly2-6mm-23S-160 - 2019 version) was inserted ∼1.4 mm through the previously drilled hole at an angle of 28 degrees with respect to the brain surface (Fig. 5.1a). The electrode array was positioned underneath the illumination area of the µLED array (Fig. 5.1b).

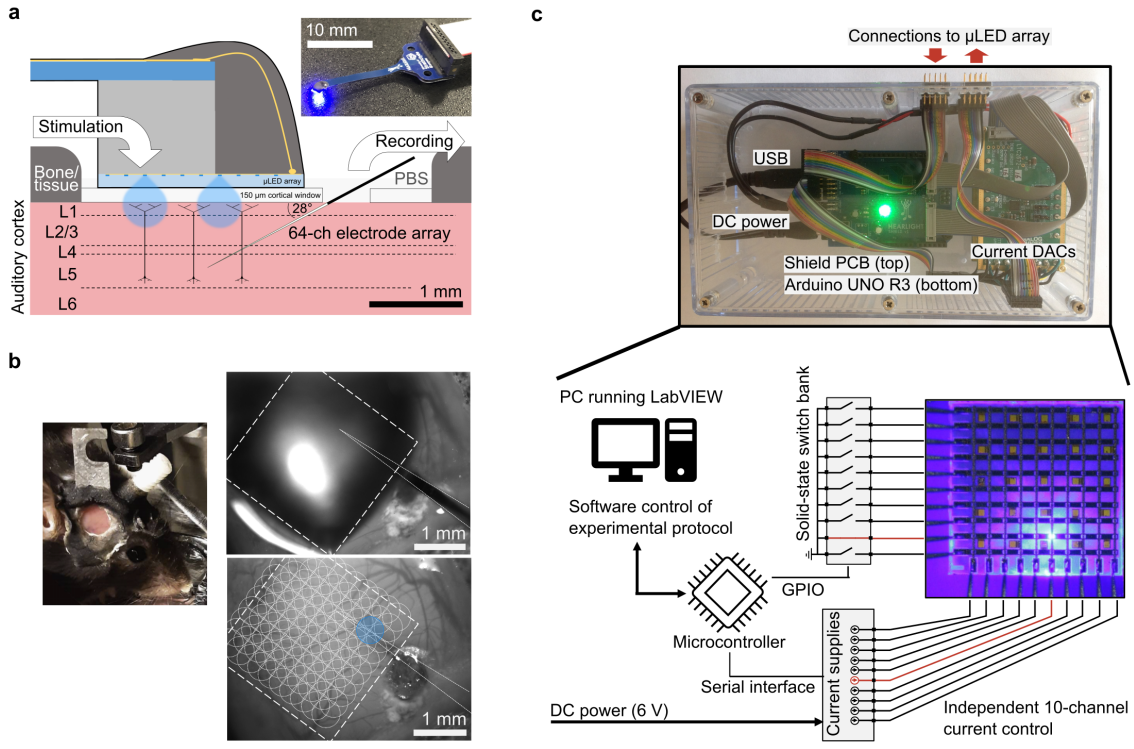


Figure 5.1: **Acute electrophysiology experimental setup. (a)** Cross-sectional schematic of experimental setup showing µLED array optogenetically stimulating mouse auditory cortex (AC) neurons through cranial window, and recording electrophysiology signals. **(b)** Photograph of mouse with craniotomy and cranial window over right AC (left). Micrographs of craniotomy illustrating stimulation site and recording geometry. **(c)** Photograph and schematic of µLED array electronic driver system.

To run the experiment, the µLED array was controlled using an Arduino-based electronic driver system and LabVIEW programme (Fig. 5.1c). This system is described in more detail, alongside the latest generation controller board, in Chapter 6. Briefly, the system utilises a 10-channel current supply connected to the p tracks of the µLED array with a 10-channel switch matrix allowing selected n tracks to be connected to ground. This allows addressing of individual µLEDs across the device, as well as simple patterns such as lines, squares and grids; but is limited in its ability to control diagonal patterns as discussed previously in Chapter 3. Throughout the electrophysiology experiments only single µLEDs were illuminated making this simple, but robust, system adequate. An Arduino microcontroller operates the described circuit and facilitates connection to the PC through a UART/USB interface. A simple LabVIEW programme allows the user to specify sequences of illumination patterns and control the cortical surface irradiance. This cortical surface irradiance is then automatically calibrated in the software to µLED drive current on a pixel-by-pixel basis as described previously in Chapter 4. The voltage traces for electrophysiology were recorded using an Intan Technologies RHD2000 recording system. The sample rate was set to 20 kHz to monitor action potential waveforms as well as monitor a trigger signal from the Arduino in one of the digital input ports, allowing recorded electrophysiology data to be temporally associated with the onset and offset of µLED illumination, for post-analysis.

Each recording session lasted approximately 1-1.5 hours and consisted of a stimulation protocol cycling through each µLED in a known pseudo-random sequence. Each µLED was pulsed 10 times (20 Hz, 50% duty cycle) at currents corresponding to five cortical irradiance levels: 1.0 ± 0.3, 1.2 ± 0.6, 2.8 ± 0.5, 3.2 ± 0.8, and 4.6 ± 1.2 mW/mm^2 with a 500-800 ms wait between each µLED/cortical irradiance level; i.e. each "trial". Fig. 5.2a shows an example recorded waveform from a strongly responding electrode-µLED pair; processed with a 1024-coefficient finite impulse response (FIR) bandpass filter [300 Hz, 5.5 kHz], common average (median) reference (CAR) from all channels subtracted from each channel to suppress common-mode noise and artefacts [10], and stimulation artifact blanking using rolling average. The illuminated pattern is shown by the grey-coloured illustration, which also indicates the relative electrode array placement (Fig. 5.2a) - this scheme is consistent across the figures in this chapter. Each µLED-cortical irradiance combination was repeated over 50, 20 or 10 trials, prioritising µLEDs near the recording electrode array (Fig 2b). This was necessary due to time constraints on animal head-fixation. Eight µLEDs were tested at all five cortical irradiance levels whereas all others were only tested at three: 1.0 ± 0.3, 2.8 ± 0.5, and 4.6 ± 1.2 mW/mm^2. Cortical irradiances are quoted as pooled mean ± pooled standard deviation over the tested µLEDs. Non-functional µLEDs or those with insufficient power were excluded. µLED drive currents corresponded to 2.4 ± 0.2, 4.0 ± 0.3, 5.8 ± 0.5, 7.7 ± 0.6 and 10.0 ± 0.8 mA. Currents are quoted as mean ± standard deviation. Both cortical irradiance and current values always refer

to the peak, i.e. during the 25 ms on-phase and are not averaged across the 10 pulses. Each mouse underwent a 5030-trial experiment. There were 5 ChR2-expressing mice (ChR2+) and one control mouse not expressing the opsin (ChR2-).

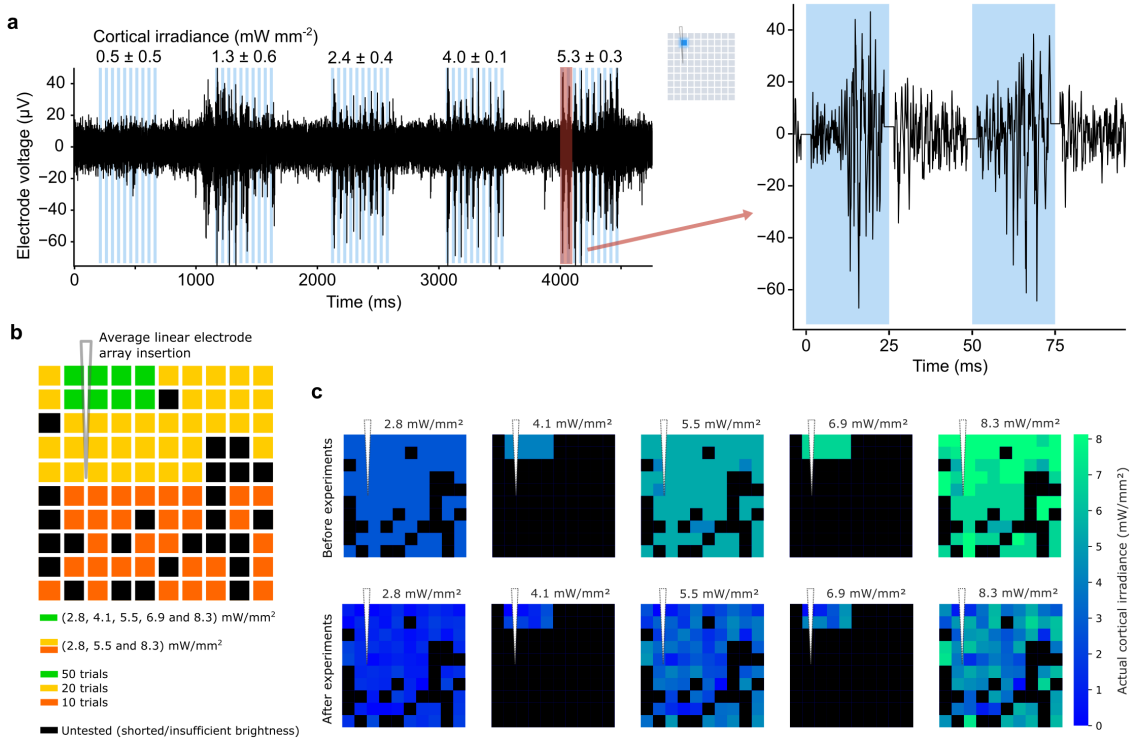


Figure 5.2: **Electrophysiology experimental protocol design. (a)** Example waveform recorded at 20 kHz from AC under pulsed stimulation from single μLED; pre-processed using FIR bandpass filter (300 Hz, 5.5 kHz), common median reference subtraction and artifact blanking using rolling average. **(b)** Schematic summary of experimental protocol optimised for animal head restraint time limitations. **(c)** Summary of degradation of device H3-019 used throughout all electrophysiology experiments; target cortical irradiance levels are shown above charts.

Throughout electrophysiology experiments, a single device in the acute configuration was used (H3-019) which experienced some degradation. Due to a lack of in-situ electrical or optical measurement capabilities, it was not possible to monitor the degree of degradation during the course of the experiments. Instead, a robust evaluation was carried out to assess what the range (mean ± standard deviation) of cortical irradiance levels delivered during the experiments were (Fig. 5.2c). This used current-voltage (I-V) and irradiance-current (L-I) measurements carried out both before and after the series of electrophysiology experiments. The cortical irradiance levels quoted throughout this section refer to the possible range of values, often quoted as pooled mean and pooled standard deviation to encompass a collection of μLEDs, for example when we are analysing rows of μLEDs or the full array. Despite this degradation, the acute packaging nevertheless provided a commendable performance over the

duration of several hour-long experiments spread over multiple weeks.

All of the analysis in this section relates to the six experiments as described. However for completeness, Table 5.1 summarises all electrophysiology experiments that were carried out over the duration of this thesis.

Experiment	Date	Device	Description	Analysis
1	October 2023	H3-019	-	Spike sorting/MUA
2	October 2023	H3-019	-	Spike sorting
1-2	October 2023	H3-019	Data corrupted due to recording probe breakage	Spike sorting
3	October 2023	H3-019	-	Spike sorting
4	October 2023	H3-019	-	Spike sorting
5	October 2023	H3-019	Control experiment	Spike sorting
A	July 2022	Device F	Utah quarter	-
B	April 2023	H3-019	Too few trials	-
C	April 2023	H3-019	Too few trials	-

Table 5.1: **Summary of all electrophysiology experiments.**

Experiments A, B and C were carried out earlier while the experimental process was less robust, therefore the results were not used to analyse activation thresholds or spatial resolution in the following sections.

5.1.2 Data analysis methods

Single-unit and multi-unit analysis (MUA) methods were used to analyse recorded electrophysiology data. A suite of custom Python classes and methods were developed to analyse the electrophysiology data and produce the figures presented in this section.

Prior to MUA, the waveforms were filtered and processed as described previously. The result of both pre-processing steps individually is shown in Fig. 5.3a. Common-mode noise arises from two sources; the electrical (capacitive coupling) or optical (Becquerel effect) stimulation artefact from driving the µLEDs in solution, and the motion artefact resulting from the mouse moving. CAR does not fully suppress the stimulation artefact, therefore as described previously, the trigger signal from the microcontroller was used to automatically identify the onset and offset of the µLED stimulus and apply a "blanking" during a -1/+2 ms window (Fig. 5.2a zoomed inset). Fig. 5.2a shows an example of a recorded waveform after all pre-processing was carried out.

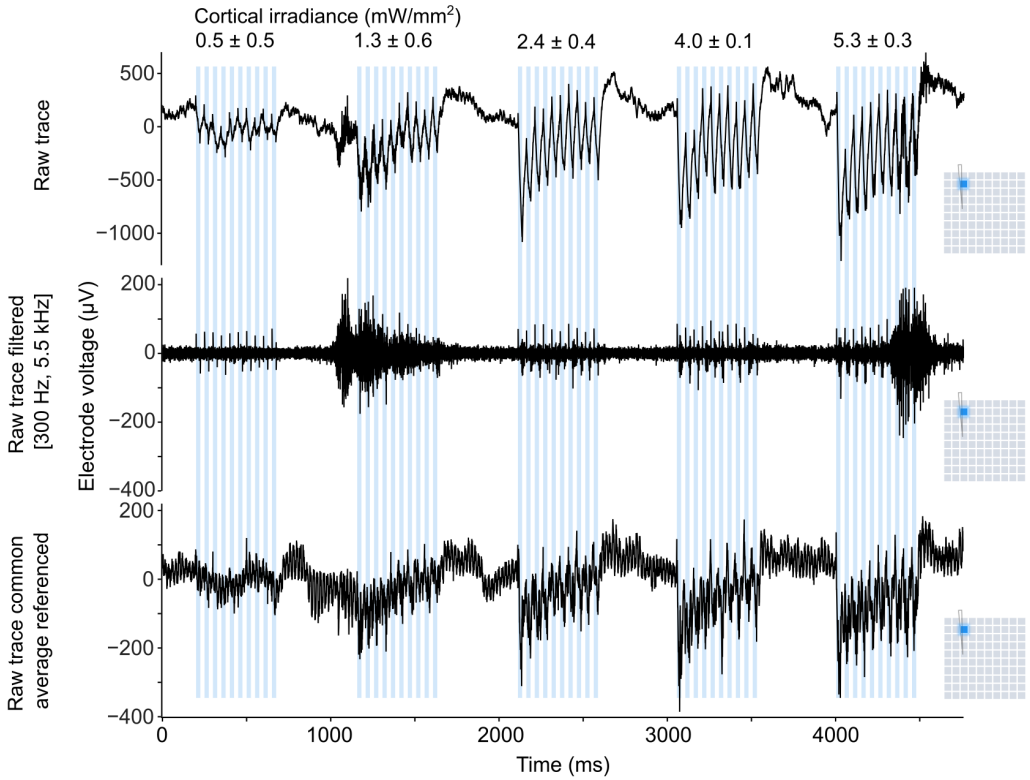


Figure 5.3: **Electrophysiology waveforms**. These correspond to same µLED and trial as Fig. 5.2a. Top: raw trace; middle: raw trace after filtering with bandpass FIR filter (cutoffs 300 Hz and 5.5 kHz); bottom: raw trace after common median reference subtracted (no filtering).

Spike sorting was used to attribute the recorded electrophysiology signal to the neuronal firing activity of individual neurons (further below in Fig. 5.5a right) allowing us to determine if neuronal responses were time-locked to the optogenetic stimulus and under what conditions. Spike sorting was carried out by collaborators using Kilosort 2.5 and Phy to remove incorrectly identified clusters (Antonin Verdier; see Acknowledgements). All subsequent analysis (using single-unit labels and corresponding spike times) was then carried out by the author using the aforementioned Python suite.

5.1.3 Stimulation threshold results

For the following analyses, only single unit activity extracted from spike sorting analysis is considered. Six recording sessions were conducted in total, five ChR2+ mice (one mouse was repeated) and a control mouse (ChR2-) to ensure detected neuronal responses were not thermally induced, or stimulation artifacts. The repeated mouse data (Experiment 1-2) was not directly used for analysing stimulation threshold, as the recording electrode array snapped

before the recording session corrupting the data, although raw data is still shown here. The linear electrode array was positioned at approximately the same location relative to the μLED array for all experiments (Fig. 5.4).

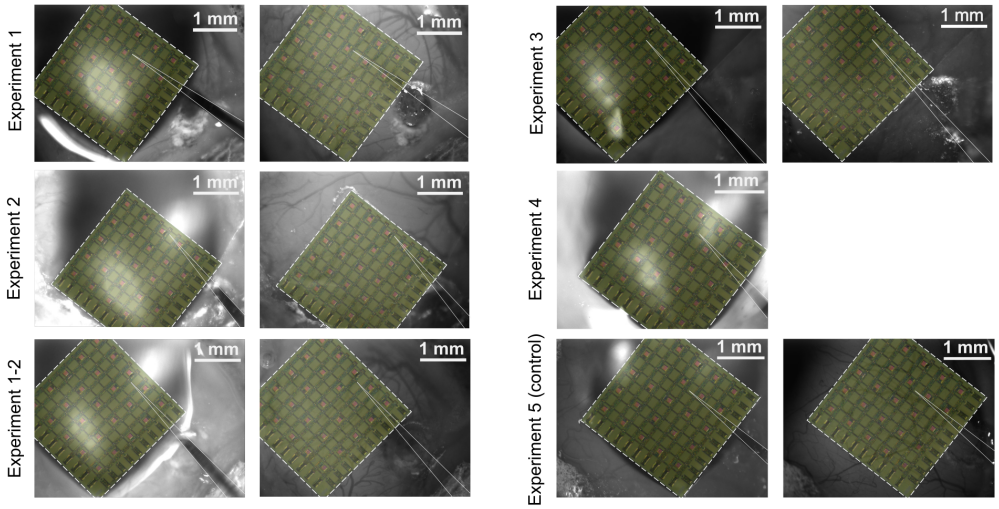


Figure 5.4: **Micrographs of craniotomy for each experiment**. These show the position of the linear electrode array indicated relative to the placement of the μLED array.

To analyse stimulation threshold, a particular μLED was selected that both maintained high performance throughout experiments, and was located near the insertion region of the linear electrode array (P3N2). Fig. 5.5 shows example recorded waveforms from each mouse, including control. For Mouse 1, two spikes have been highlighted along with an average spike waveform representative of each (Fig. 5.5a).

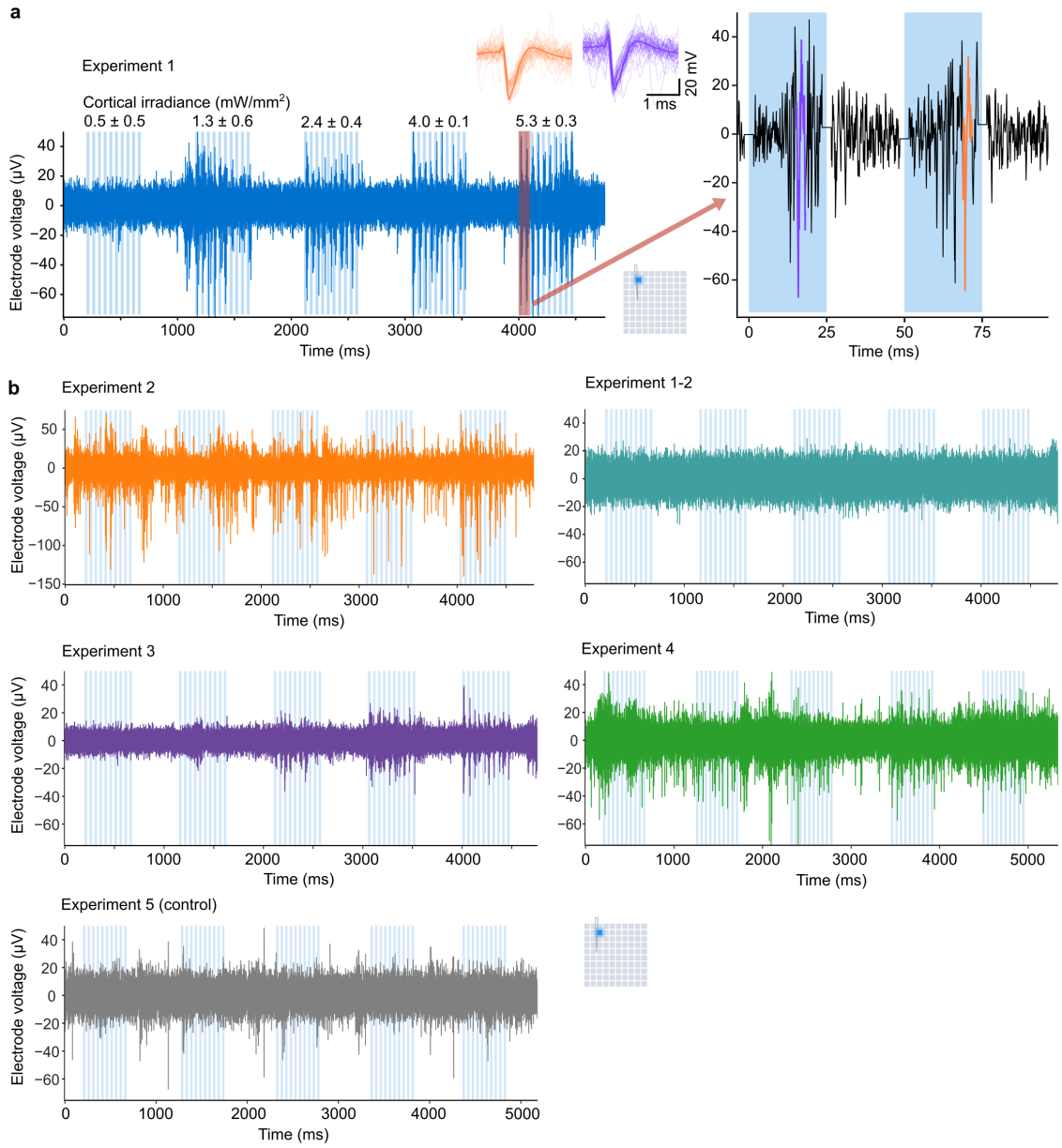


Figure 5.5: **Electrophysiology recordings. (a)** Example of electrophysiology recording from a single electrode (filtered using bandpass FIR filter (cutoffs 300 Hz and 5.5 kHz), common median reference subtracted, and stimulation artefact blanked using rolling average). Each µLED is pulsed 10 times at 20 Hz, 50% duty cycle for 5 cortical irradiances (mean ± standard deviation) with corresponding µLED drive currents of 2.4, 4.0, 5.8, 8.0, 10.4 mA. Average waveforms of highlighted sorted spikes are shown (right). **(b)** Similar electrophysiology recordings for all other experiments.

Fig. 5.5 appears to demonstrate high variability across animals. Although these waveforms are only indicative (as they represent a very small sample of the dataset), we later show that

there are significant statistical differences in the neuronal responses across mice. This is discussed in more detail later in this section, though is most likely the result of differences in the relative positioning of the µLED array on the cortical window, relative to the recording electrode array.

Spike sorting was carried out for all experiments and spike times are shown as a raster plot for all neurons for Experiment 1, accumulated over all 50 trials (Fig. 5.6a top). Spike times were summed within 25 ms bins and averaged over all neurons to yield peri-stimulus time histogram (PSTH) plots (Fig. 5.6a bottom). The background was taken as the mean spike counts per 25 ms bin in the 200 ms periods before and after the 500 ms of pulsed stimulation time.

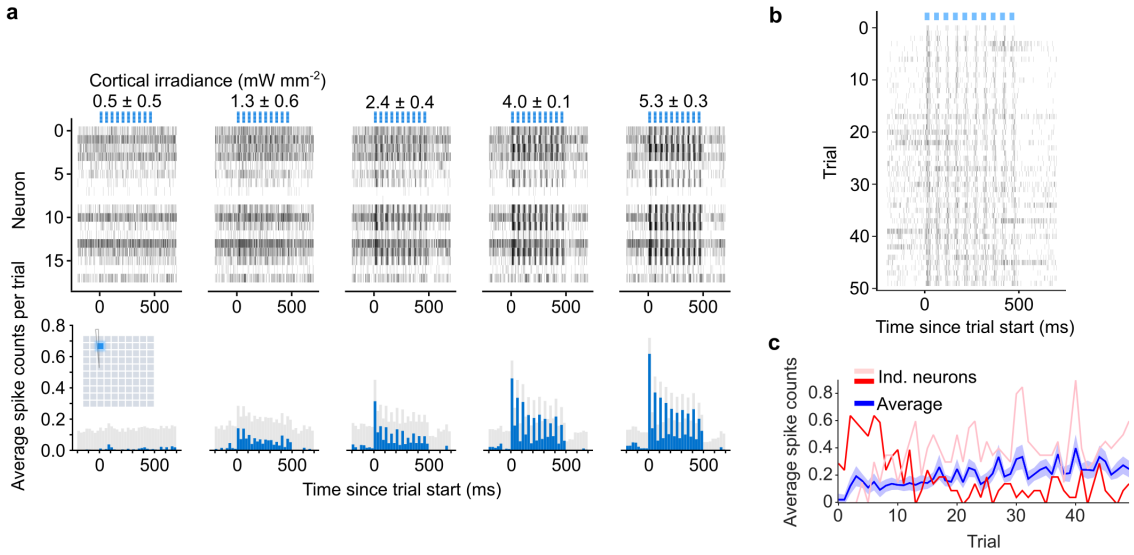


Figure 5.6: **Raster and peri-stimulus time histogram (PSTH) plots for Experiment 1. (a)** Raster plot of sorted spike times under stimulation from µLED P3N2, accumulated across 50 trials (top). Peri-stimulus time histogram (PSTH) plot showing binned average spike counts per trial with mean of background subtracted, averaged over all neurons. **(b)** Raster plot for highest tested irradiance (5.3 ± 0.3) mW/mm^2 over duration of experimental trials, accumulated across all neurons. **(c)** Analysis of individual unit responses (average spike counts per trial) over the course of the experiment; same set of trials as shown for (b). Two individual neuron responses are highlighted (red and pink lines).

Both raster plots and PSTH plots show neuronal responses time-locked to the µLED stimulus appearing at higher cortical surface irradiance levels, usually 4-5 mW/mm^2. Fig. 5.6b further shows that neuronal response was consistent across the duration of the experiment. To compare directly across mice, a dose response was plotted for each experiment, only considering "time-locked" neurons. Neurons were identified as time-locked if their average spike counts per trial during the 500 ms pulsed stimulation time were greater than 2σ from the average background spike counts; again, the background was taken from the mean spike

counts per trial during the 200 ms periods before and after the 500 ms stimulation time. Fig. 5.6b shows the equivalent raster plot, in terms of trials, accumulated across all neurons. This confirms neuronal responses were consistent throughout the duration of the experiment. Fig. 5.7 shows the dose response for each ChR2+ experiment, including ChR2-.

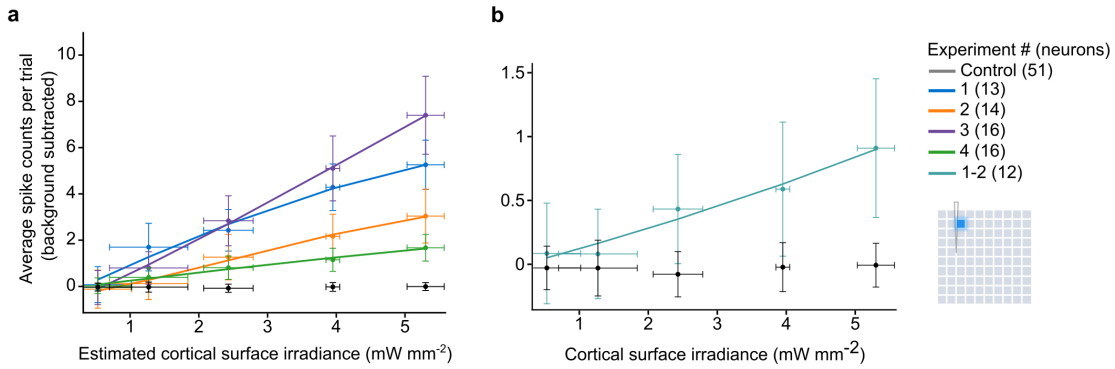


Figure 5.7: **Dose response curves. (a)** Dose response for four mice plus control, showing average spike counts per trial with background subtracted (ChR2+: n = 13, n = 14, n = 16, n = 16 time-locked neurons; ChR2-: n = 51 neurons). **(b)** Dose response for Experiment 1-2 which was omitted from main analysis due to poor recording quality (n = 12 time-locked single units for optogenetic (ChR2+) experiment; n = 51 single units for control (ChR2-) experiment).

The dose responses (Fig. 5.7) show pooled mean, with error bars as pooled SEM of average spike counts per trial (background subtracted); ChR2+ (optogenetic): n = 13, 14, 16, 16, 12 time-locked neurons; ChR2- (control): n = 51 neurons.

Sigmoid curves were fitted to each set of points to aid visualisation. For this analysis, we defined threshold activation as a 50% success rate, i.e. 1 spike per 2 pulses on average per neuron. This is equivalent to 1 spike per 50 ms time window, which remains consistent with the temporal kinetics of ChR2/H134R [11]. These results confirm a stimulation threshold between 4.0 ± 0.1 and 5.3 ± 0.3 mW/mm^2 corresponding to approximately 5 mA drive current, consistent with low thermal effects as discussed in Chapter 4.

The average spike counts per trial for different illuminated µLEDs were plotted to examine effects on the stimulation threshold from both µLEDs distant and nearby to the recording electrode array (Fig. 5.8). For ChR2+ mice, mean spike counts per trial (after background subtraction) were significantly higher for µLEDs located nearby the electrode array, and when compared to the ChR2- experiment.

The boxplots in Fig. 5.8 show error bars as pooled SEM of standard deviations of average spike counts per trial (background subtracted); significance levels were determined using the independent two-sample, single-tailed Student's t-test with alt. hypothesis: spike counts (ChR2+) > spike counts (ChR2-); *p < 0.05, n.s. means alt. hypothesis rejected at p < 0.05); ChR2+ (optogenetic): n = 18, n = 45, n = 16, n = 53, ChR2- (control): n = 51 neurons.

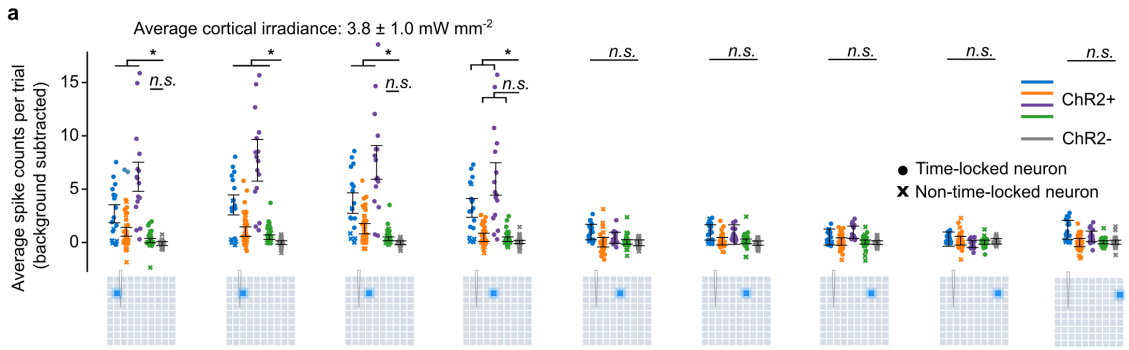


Figure 5.8: **Box plots showing average spike counts per trial with background subtracted**. Stimulation from different µLEDs across the array (ChR2+: n = 18, n = 45, n = 16, n = 53, ChR2-: n = 51 neurons); time-locked neurons are shown as circles, non-time-locked as x's; cortical irradiance given as pooled mean ± pooled standard deviation.

Electrophysiology experiments confirm robust activation thresholds at modest cortical surface irradiance levels (approximately 4 – 5 mW/mm^2) corresponding to low µLED drive currents (<5 mA), well within their operating regime and the specifications of the aforementioned electronic driver system. µLEDs located nearby the recording electrode array elicited significant neuronal responses above background activity across all ChR2+ mice, which reduced with increasing µLED-electrode distance. This suggests spatial selectivity capabilities which is explored in more detail in the next section. Ideally, the devices should stimulate the same ensembles of neurons reliably between trials and over the duration of an experiment, such that consistent perceptions are generated in the auditory cortex. While, in general, bulk neuronal activity seems consistent over the duration of the experiment, albeit with progressively increasing background activity possibly due to tissue damage or immune response (Fig. 5.6b), the consistency of individual unit responses to the same stimulus is important. This shows across all neurons, the response is broadly consistent on average, but that individual units appear increase or reduce their response over the course of the experiment and on a trial-by-trial basis (Fig. 5.6c). It is difficult to discern the exact reason for this, which could be due to the effect of electrode drift or the non-ideal placement of the acute devices on the cortical window. Although not validated though electrophysiology, it is expected that neuronal responses will be significantly more consistent with the chronically implanted devices, which eliminate both the aforementioned effects. Lastly, large variation in neuronal response between animals could be due to the relative positioning of the µLED stimulation site to the recording electrode array (both laterally and with depth). This could be exacerbated due to some difficulties during experiments with placing the acute device on the cortical window, meaning the device may not have been perfectly flat in all experiments. This could further arise from the overall electrode array insertion quality and surrounding tissue health. In the following sections, we explore

chronic implantation of the devices directly on the cortical surface, thereby eliminating the cortical window to mitigate these effects.

5.1.4 Spatial resolution

Spatial resolution can be inferred by analysing single unit responses to optogenetic stimulation of individual µLEDs across the array. As described previously, our experimental protocol cycled µLEDs across the array in a pre-defined pseudo-random sequence allowing spatial maps to be plotted; these show average spike counts per trial with background subtracted (average over all time-locked neurons) (Fig. 5.9).

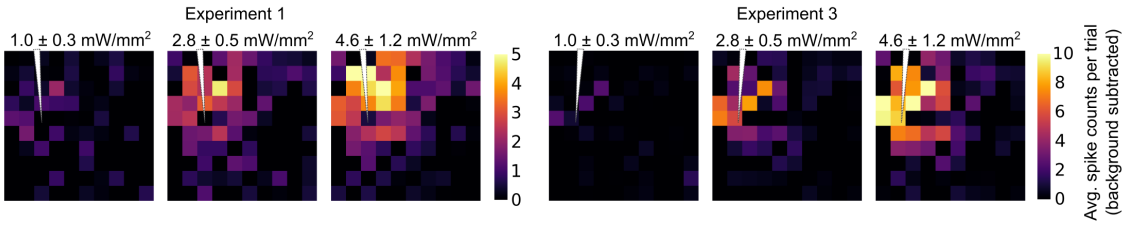


Figure 5.9: **Spatial maps of neuronal response**. Average spike counts per trial with background subtracted, under optogenetic stimulation from µLEDs individually across the array.

To analyse spatial resolution, we focused on the two experiments with the strongest response (Experiment 1 and Experiment 3). Weighted Gaussian curves were fitted to rows N3 and N4 for the highest tested cortical surface irradiance (non-tested µLEDs were excluded from analysis).

For the spatial resolution plots in Fig. 5.10, Gaussian curves were fitted to each row using data points, x, weighted by the error bars.

$$y = y_0 + A \cdot \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \tag{5.1}$$

Initial values for (y_0, μ, A, σ) were (0, -500, 50, 200) with bounds as ($[0, 2]$, $[-900, 0]$, $[0, 200]$, $[100, 1000]$). The x axis is in µm. The standard error of the fitted parameters defined the shaded region. Reduced χ^2 values were 0.56 and 1.40 for rows 3 and 4 of Experiment 1, and 1.68 and 4.43 for rows 3 and 4 of Experiment 3.

Assuming the full-width at half-maximum (FWHM) as the cutoff for neuronal response, this analysis places the estimated spatial resolution between 563 ± 132 µm and 890 ± 381 µm (averaging across both experiments). These estimates are significantly wider than the spacing of the µLEDs on the array (200 µm). This would be expected due to the Lambertian emission projecting a uniform illuminated spot of ∼270 µm diameter and the scattering in tissue further dominating the spatial resolution. This was discussed alongside optical modelling along with

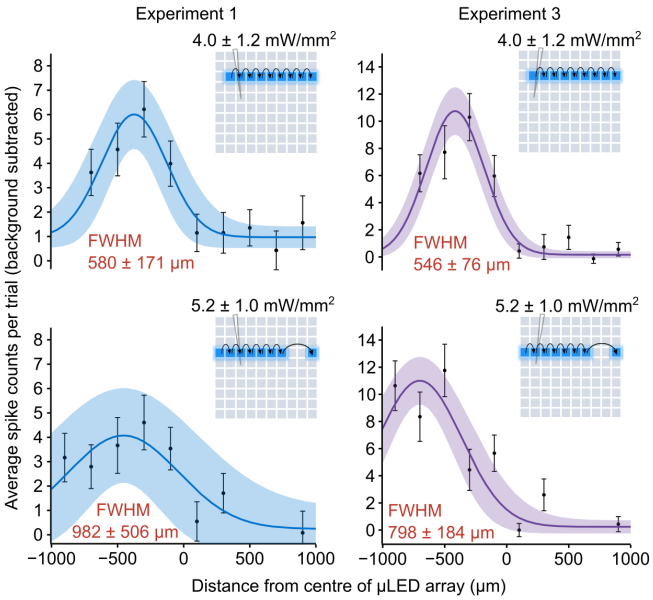


Figure 5.10: **Weighted Gaussian curves**. These are fitted to rows 3 of 4 of the spatial maps. Error bars are shown as pooled SEM of standard deviations of average spike counts per trial (background subtracted); n = 13, 16 time-locked neurons.

strategies to improve the spatial resolution, in Chapter 4. Here, we also use the highest tested optical power/cortical irradiance, however responses can still be seen at lower power and hence reduced spatial width of activation. In this section, we wanted to use the spatial resolution estimates from electrophysiology analysis to validate the optical models.

An optical model was developed using Zemax OpticStudio 2021 in non-sequential mode, incorporating the optical properties of brain matter described in Chapter 4 and the glass cranial window. Fig. 5.11 shows the emission profile from the µLED through the sapphire substrate and glass cranial window and through tissue. Contours correspond to the 1 mW/mm^2 level assuming tested cortical irradiance levels of 1.0 ± 0.3, 2.9 ± 0.5, 5.3 ± 0.7 mW/ mm^2; error regions correspond to the uncertainty in cortical surface irradiance due to aforementioned device degradation. Optical power levels (optical power delivered to the brain) are quoted instead of cortical surface irradiance to provide a consistent benchmark with following analyses (and previous analyses from Chapter 4) which alter the illuminated spot size on the cortical surface; hence using the same cortical surface irradiance across models/analyses would provide an invalid comparison. The optical power delivered to the brain and µLED drive current must be consistent between analyses.

Fig. 5.11b shows the emission profile given the device directly implanted on the cortical surface, in the case of chronic implantation. Larger patterns of simultaneously illuminated µLEDs were modelled by summing the modelled profile results from the single µLED models

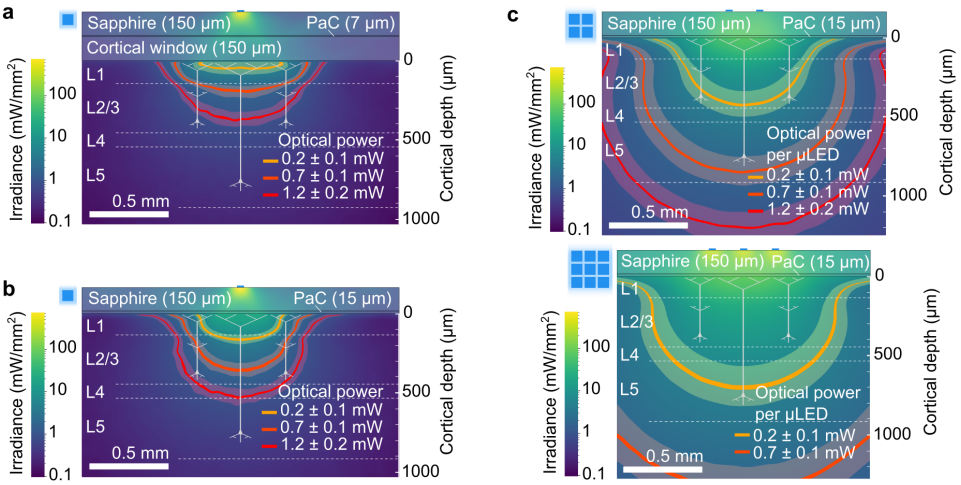


Figure 5.11: **Modelled optical profiles of *in-vivo* experiments. (a)** Optical profile corresponding to electrophysiology experimental setup with device positioned above chronic cortical window. **(b)** Equivalent to (a) but device sapphire surface positioned directly on brain. **(c)** Equivalent to (b) showing central plane of cross-section from several modelled profiles summed together. In all cases the optical power delivered to the brain is the same (for 2x2 and 3x3 cases, this is per µLED), although the cortical irradiance differs between the with/without cortical window cases as the illuminated area is different.

offset by 200 µm in three dimensions (Fig. 5.11c). The activated volume significantly increases in both cases, up to the scale of an entire cortical region.

For the following analyses, the estimated spatial extent of activation (informing spatial resolution) is compared between the neuronal responses given in Fig. 5.10 and the width of the 1 mW/mm^2 contour corresponding to the equivalent cortical surface irradiance/optical power used in the experiment (Fig. 5.11a). The width of the 1 mW/mm^2 contour is taken at a depth of 150 µm in the modelled cortical tissue, consistent with the layer 2/3 boundary where we were targeting optogenetic activation. The 1 mW/mm^2 level is the approximate threshold for ChR2 [12] providing an educated comparison between these.

Spatial resolution estimates from neuronal responses and optical modelling provide reasonable agreement (Fig. 5.12a) hence validating our optical models and the analyses given in Chapter 4 relating to improving the spatial resolution through a collimating optical interposer. However, this comparison is an approximation which ignores dendritic contributions and second and further-order network effects.

To improve spatial resolution, the most obvious approach is to remove the cranial window and implant the device directly on the cortical surface as shown in Fig. 5.11b. In both models (Fig. 5.11a with the window and Fig. 5.11b without) the same optical power was delivered to the brain, hence the same µLED drive current would be needed; although the cortical surface irradiance is higher for direct implantation due to the reduced illuminated spot size.

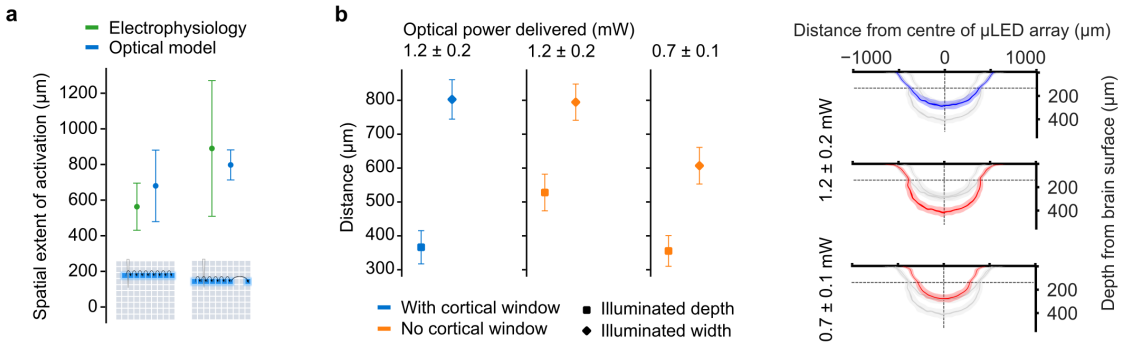


Figure 5.12: **Optical model validation. (a)** Comparison of spatial extent of neuronal response taken as average of FWHM from Weighted Gaussian fits for each experiment in Fig. 5.10, and width of 1 mW/mm^2 contour at depth of 150 µm from optical model delivering 1.2 ± 0.2 mW to brain. **(b)** Comparison of width (at depth of 150 µm) and depth (at centre) of 1 mW/mm^2 contour from modelled optical profile under different experimental conditions (left). Graphical representation of data in left panel comparing the 1 mW/mm^2 contour across cases (right).

Both the depth and width of the modelled illumination region (again corresponding to the 1 mW/mm^2 contour) were compared between the models with and without a cranial window (Fig. 5.12b). For equivalent optical power (1.2 mW), removal of the cortical window allows deeper penetration of light with similar lateral width while reducing the optical power to 0.7 mW reduces the lateral width while maintaining the same depth of light penetration. Direct implantation of the device on the cortical surface is therefore the best way to utilise these devices in *in-vivo* experiments, with the dual-benefit of improved spatial resolution while reducing drive current.

5.1.5 Multi-unit analysis

The key results from the previous sections – activation thresholds and spatial resolution – were determined using single unit activity identified through spike sorting. This short section explores an alternative technique using multi-unit analysis (MUA) [13]. In contrast to spike sorting, which "throws away" activity aggregated from multiple neurons spiking responses in favour of clearly identifiable single units, MUA considers this activity providing an insight into broader network effects. Similar filtering parameters are used as with spike sorting – in our case this is an FIR bandpass filter with cutoffs [300 Hz, 5.5 kHz] – and we carry out common average referencing and stimulation artefact blanking as described previously.

MUA can be carried out using thresholding on the waveform or by calculating the envelope – envelope MUA or eMUA [14]. This section uses the former approach by computing a rolling average of 900 ms sections of the waveform (500 ms stimulation time with 200 ms before and after). Signal present above a 5σ threshold from this rolling average is counted as a "spike" –

individual spikes are only counted if the signal returns below and back above the threshold. A Python script was used to carry out this analysis over the entire recorded waveform and for all 64 recording channels. Fig. 5.13 shows an example of the recorded waveform for a single µLED increasing in irradiance, after pre-processing, with the 5σ threshold lines and several spikes highlighted. The elevated signal during the third set of pulses is the result of a movement artefact; this tended to be consistent across animals, occurring every few seconds during the recording.

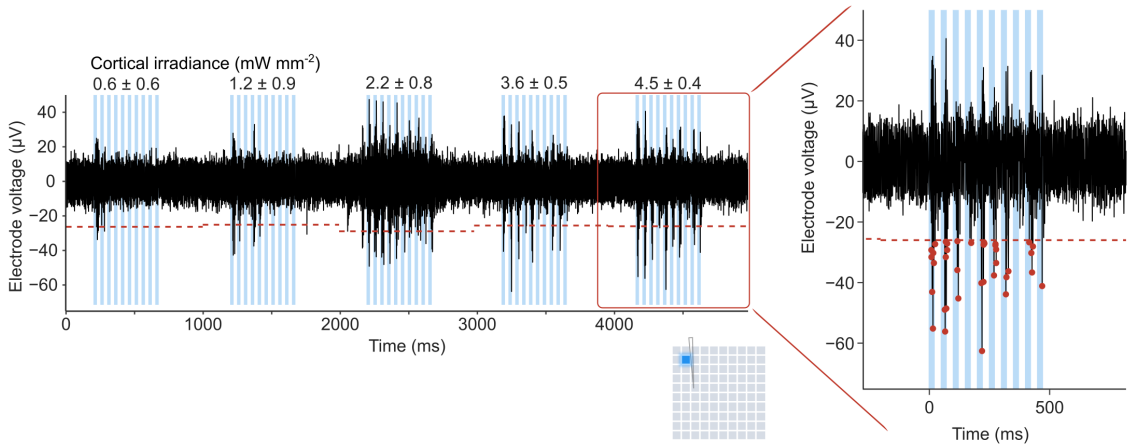


Figure 5.13: **Example waveform recorded at 20 kHz from auditory cortex (AC) under pulsed stimulation from single µLED**. Pre-processed using FIR bandpass filter (300 Hz, 5.5 kHz), common median reference subtraction and artefact blanking using rolling average. Spikes counted through MUA thresholding indicated (right).

To quantify this response, the spikes were plotted as a raster plot for this same µLED, accumulated across all 50 trials; this was done for all 64 recording channels from base (cortical surface) and tip (depth of 680 µm) (Fig. 5.14a top). A schematic illustration of the probe insertion is given in Fig. 5.14b. As before, this response was binned into 25 ms bins and averaged across all recording channels, to produce a peri-stimulus time histogram (PSTH) plot clearly showing time-locked responses at comparable thresholds to those obtained through spike sorting analysis (Fig. 5.14a bottom).

To provide a quantification of neuronal response as a function of µLED, cortical irradiance and/or recording channel, the spike counts were summed across the bins within the 500 ms stimulation time (after background subtraction).

Fig. 5.15a shows the average spike counts per trial for all recording channels as the position of the illuminated µLED is changed. This shows a stronger response at the base of the probe nearby the illuminated µLED. This response becomes weaker and isolated to the tip for µLEDs illuminated in the middle of the array where the recording electrodes are further from the surface (within layer 5). Fig. 5.15b shows individual recording channel responses to each µLED

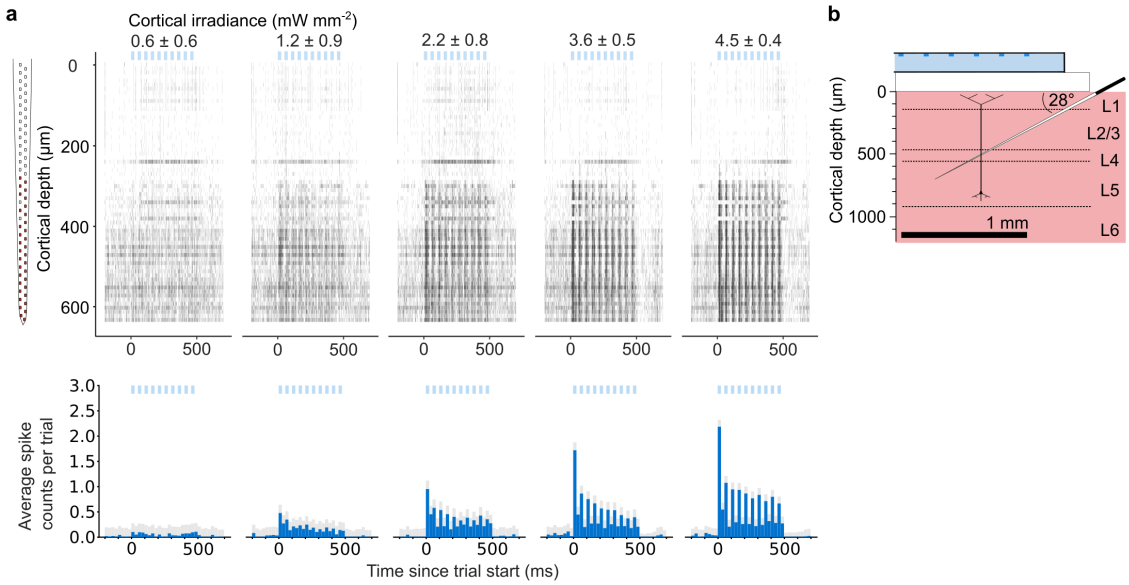


Figure 5.14: **Raster and peri-stimulus time histogram (PSTH) plots from MUA. (a)** Raster plot of thresholded spike times under stimulation from same µLED as Fig. 5.13, accumulated across 50 trials (top). Peri-stimulus time histogram (PSTH) plot showing binned average spike counts per trial with mean of background subtracted, averaged over all recording channels demonstrating a time-locked response (bottom). **(b)** Schematic of probe insertion under µLED array.

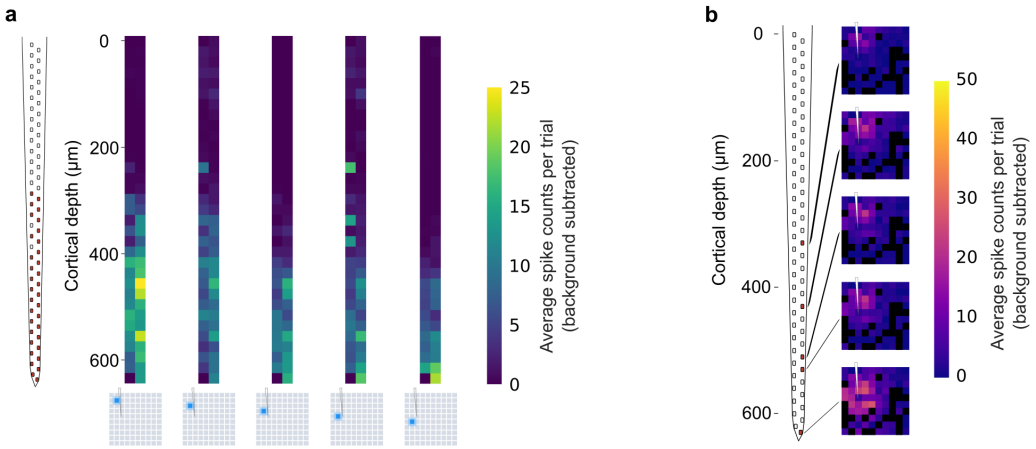


Figure 5.15: **Electrode array plots. (a)** Average spike counts per trial (with background subtracted) for each electrode, as illuminated µLED position moves away from the probe. **(b)** Heat maps showing average spike counts per trial (with background subtracted) across all µLEDs individually illuminated, for selected electrode positions down the probe shank.

being illuminated across the entire array. Interestingly, this suggests a stronger response at the electrode array tip, and with a much broader spatial extent of illuminated regions driving responses. This strongly supports the hypothesis that dendritic or second-order activation is

occurring.

This section has explored MUA as an alternative to spike sorting to analyse recorded electrophysiology data. Analysing neuronal responses isolated to individual recording channels has provided and could provide further insight into the optogenetic activation regime with our µLED arrays. Although this section has provided an alternative analysis of the recorded electrophysiology data through multi-unit analysis, it is by no means exhaustive. Local field potential (LFP) activity is evident in our recorded waveforms (Fig. 5.3) indicating broad network effects within the auditory cortex being driven through optogenetic stimulation with the µLED array. Therefore, it would be possible to analyse this data further to provide additional insights.

5.2 Chronic behavioural experiments

Experimentation in behaving animals allows us to study the circuit-level dynamics involved in a range of cognitive and perceptual processes [15]. Optogenetics is an advantageous technique here due to cell-type specificity and the potential for improved spatial resolution compared with electrical stimulation [16]. The approach carried out here utilises µLED arrays chronically implanted using the flexible device packaging described in Chapter 3, and removal of the aforementioned cranial window previously used for electrophysiology, such that the devices are in direct contact with the cortical tissue. Optogenetic stimulation is delivered to distinct sub-regions of the auditory cortex while observing behavioural responses to a spatial discrimination task. Two sets of behavioural experiments were carried out. Firstly, a head-fixed task where the mouse was also implanted with a medial forebrain bundle (MFB) stimulator. This motivated the mouse to learn the task by reliably activating a dopamine reward circuit in the brain; the animal cannot become sated with this technique as with water or food rewards [17]. Later, we carried out an experiment in freely-behaving mice with the same chronic implantation of µLED array but no MFB stimulator. In this case, the mice were motivated by receiving a water reward delivered via a lick spout within a behavioural chamber. The experimental paradigm is summarised in Fig. 5.16.

The purpose of these experiments was to corroborate the findings from the electrophysiology experiments and confirm activation thresholds, while utilising the improved device packaging capable of facilitating chronic implantation. Ten individual devices were separately implanted and tested across both sets of experiments and, in every case, they successfully elicited reliable behavioural responses (for ChR2+ mice) with minimal device degradation (Table 5.2).

Occasionally, mice were able to chew on the connectors of devices implanted on other mice, damaging the device beyond repair. Therefore, mice were housed separately over the course of these experiments, however, they were able to remain non-encumbered due to the

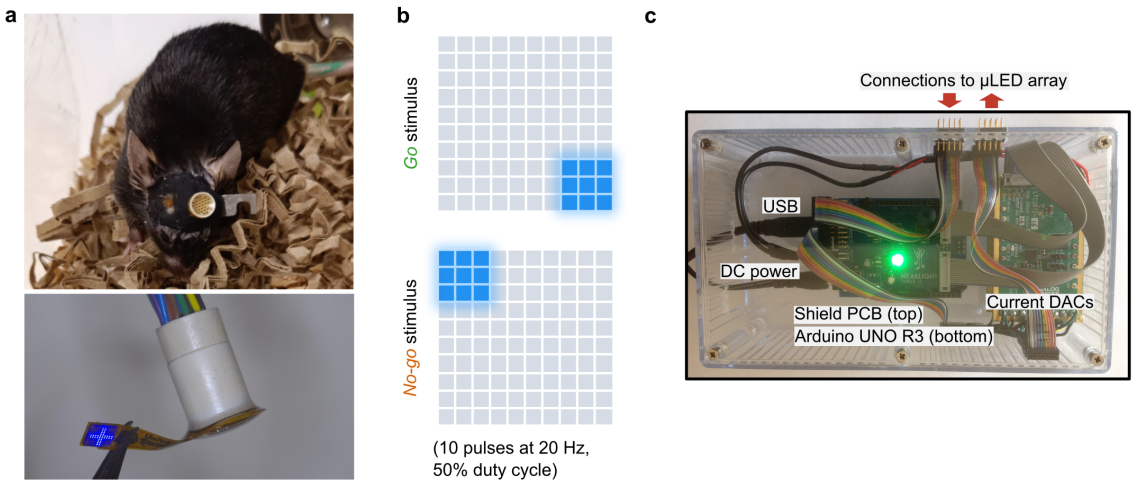


Figure 5.16: **Summary of chronic behavioural experimental paradigm. (a)** Photograph of mouse chronically implanted with µLED array with flexible packaging (top). Photograph of device before implantation (bottom). **(b)** Example of patterned optogenetic stimulation delivered by µLED array. Mouse should carry out an action depending on trial type to receive reward and progress task. **(c)** Photograph of control system used for behavioural experiments; the same as that used for electrophysiology experiments.

Experiment type	**Experiment #**	**Device type**	**Encapsulation**	**Cortical window used**
Electrophysiology	1, 2, 1-2, 3, 4, 5	Acute	PaC (7 µm)	Yes
Behaviour (head-fixed)	1	Chronic	PaC (15 µm)	No
	2	Chronic	PaC (15 µm)	No
	3	Chronic	PaC (15 µm)	No
	4	Chronic	PaC (15 µm)	No
	5	Chronic	PaC (15 µm)	No
	6	Chronic	PaC (15 µm)	No
Behaviour (freely-moving)	1	Chronic	PaC (15 µm)	No
	2	Chronic	PaC (30 µm) / silicone	No
	3	Chronic	PaC (15 µm)	No
	4	Chronic	PaC (15 µm)	No

Table 5.2: **Summary of devices and encapsulation schemes used in experiments**. Each row represents a different device.

extremely lightweight (<300 milligrams) and minimally-invasive device form factor.

All surgeries and behavioural experimentation was conducted within the Bathellier laboratory. Data analysis for the head-fixed behavioural experiments was conducted by the author and is detailed fully here. Freely-behaving experimental data analysis was not carried out by the author and is therefore only briefly summarised here, although all available data has been

presented.

5.2.1 Behavioural experiments in head-fixed mice

For the head-fixed behavioural experiments, six mice were chronically implanted with μLED arrays above one auditory cortex (AC) and each had an MFB stimulator electrode implanted (Fig. 5.17a). Four mice were opsin-expressing (ChR2+) and two non-opsin-expressing (ChR2-) similar to before to provide a control. The total number of mice was chosen to be n = 6; this was to minimise the number of animals while maintaining good statistics, and to mitigate unforeseen difficulties that could reduce the n affecting the outcome of the experiments. Ideally a statistical power analysis should be used to select an appropriate number of mice (n) for a particular study. This ensures that n is not too small such that the effect being studied is missed during statistical analysis, while simultaneously making sure n is not too large resulting in wasted resources [18]. Post-surgery recovery lasted 5 days before mice began behavioural experimentation. The experimental setup comprised a contention tube where the mouse was placed with a lick spout within reach. This was connected to the analog port of a National Instruments DAQ operated via a Matlab programme (developed within the Bathellier laboratory). This also provided the control of the MFB stimulator electrode to provide the mouse with a reward to motivate task learning. This Matlab programme provided the Arduino-based electronic controller system (Fig. 5.16c) with a trigger signal to start optogenetic stimulation of the μLED arrays at the appropriate time. The experimental setup is shown with a schematic in Fig. 5.17b and photograph of the implanted mouse within the contention tube (Fig. 5.17c).

The behavioural experiment consisted of several phases. Firstly, without optogenetic stimulus the mouse was conditioned to being head-fixed within the experimental setup; licking the sensor and learning that the reward of MFB stimulation is available. For behavioural experiments, a simple *go*/*no-go* experimental paradigm was employed utilising 3x3 patterns of illuminated μLEDs on opposing corners of the array (Fig. 5.16b), although exact positioning of the pattern was dependent on the particular devices' performance. The same temporal protocol was used as with electrophysiology experiments (20 Hz pulse repetition rate at 50% duty cycle) and cortical surface irradiance (per μLED) varied between mice and experimental session (see Appendix A.4 for full experimental details).

For each mouse, overall experimental time lasted several days. Each day is referred to as an experimental session consisting of several hundred trials. Each trial provides a 10-pulse burst of optogenetic stimulation lasting 500 ms followed by a response window allowing the mouse to make a decision to lick the sensor based on the stimulation pattern; *go* or *no-go*. If the mouse licks the sensor within the response window after the *go* stimulus, a reward of medial forebrain bundle stimulation is delivered immediately. If the mouse licks in response to

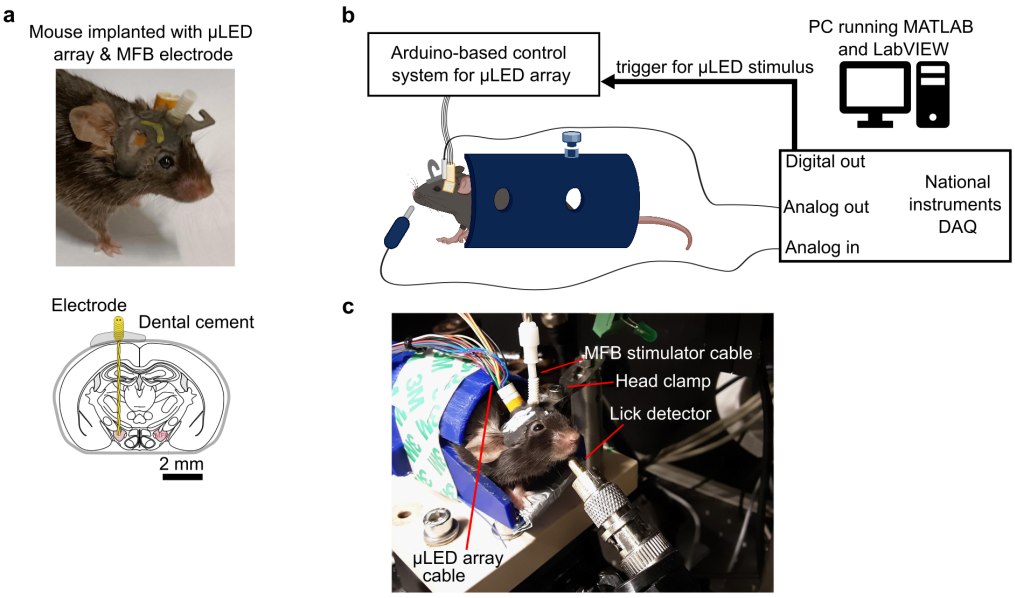


Figure 5.17: **Head-fixed behavioural experiment setup. (a)** Photograph of mouse chronically implanted with µLED array and medial forebrain bundle (MFB) stimulator electrode (top). Coronal schematic of mouse brain showing insertion of MFB stimulator electrode (bottom). **(b)** Schematic of experimental setup including Matlab programme triggering µLED array control system, recoding lick responses and delivering MFB stimulation. **(c)** Photograph of mouse within head-fixed experimental setup.

the *no-go* stimulus, a time penalty is incurred accruing to the default random inter-trial delay, delaying the beginning of the next trial and hence the time the mouse must wait to receive the next reward. Otherwise, when the mouse does not lick, there is no reward or penalty for either trial type. A small number of *blank* trials, providing no optogenetic stimulus are randomly interspersed to act as a control.

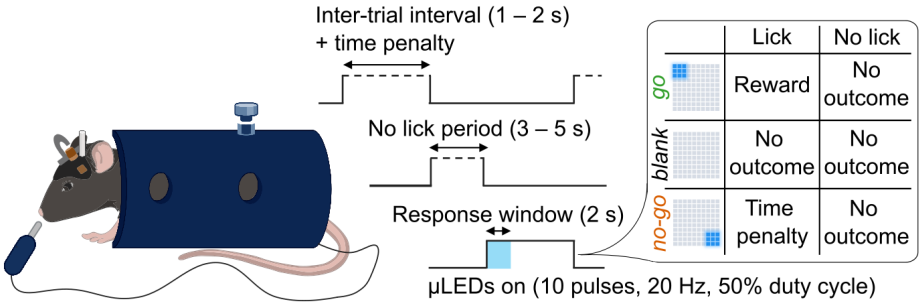


Figure 5.18: **Timing diagram for an experimental trial in the head-fixed behavioural experiment.** After a random inter-trial interval and period where the mouse must not lick, the µLED pattern is activated depending on trial type and the response window opens.

To allow the mice to more easily learn this task, two sets of experiments were carried out; association followed by discrimination. In the association experiment, only the *go* pattern is presented and in the discrimination experiment both *go* and *no-go* patterns are presented. The discrimination experiment follows after the mouse has successfully learned the association experiment. Fig. 5.19 shows lick raster plots for all association experiments, which included the two control mice. These plots show the trial duration on the x axis, beginning from the trial start when the µLED pattern is illuminated for 500 ms, and the trials across all sessions on the y axis (this varies across mice as the individual trial times are performance-dependent, e.g. due to the punishment pause). Individual markers show the start and stop times when the mouse's tongue made and released contact with the sensor, with the mean and standard deviation of the lick times for each trial type shown.

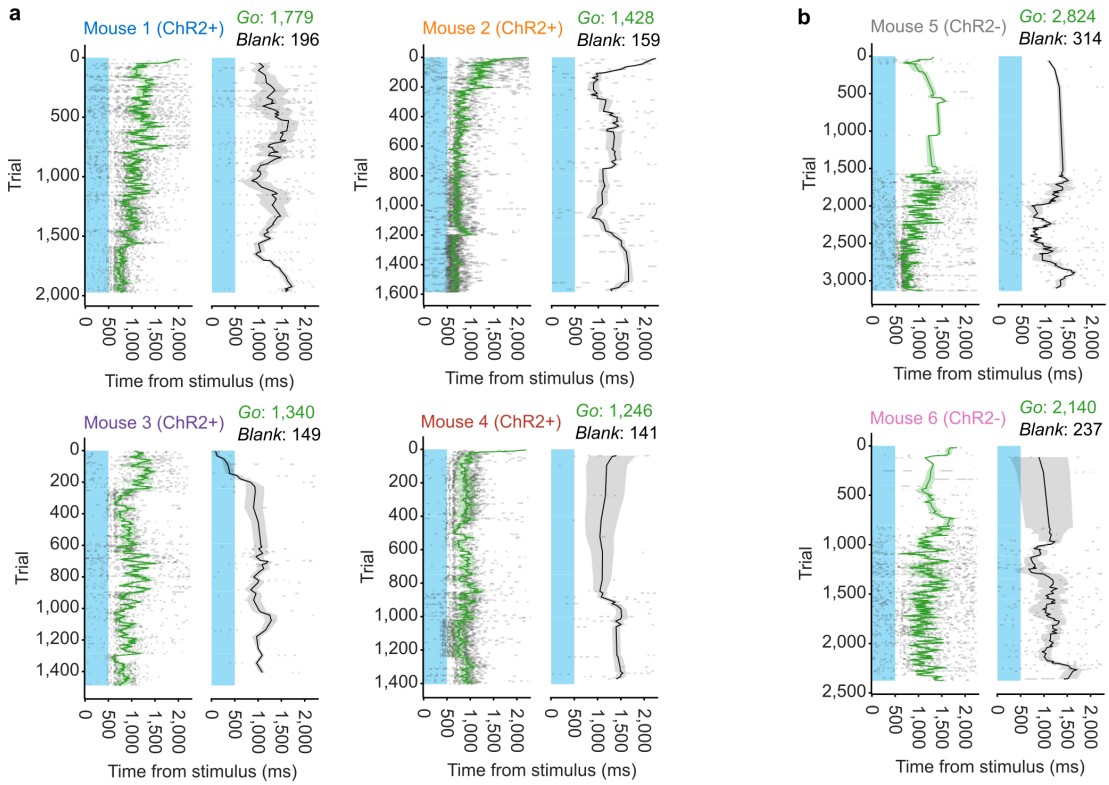


Figure 5.19: **Lick raster plots for association experiment. (a)** Lick raster plots for opsin-expressing mice (ChR2+). **(b)** Lick raster plots for control mice (ChR2-). In all plots, shaded region shows mean and standard deviation of lick times per trial.

All opsin expressing mice appear to learn the task well, with average lick timing appearing closer to the onset of optogenetic stimulation and with reduced variance in timing. There is some inter-session variance, possibly due to the motivation of the mice reducing or increasing on particular days. The control mice did not lick the sensor initially as with

the opsin-expressing mice, although when the MFB stimulation voltage was increased (after approximately 1000 trials), licking did increase, although this was not timed with the onset of optogenetic stimulation. Mouse 5 did exhibit delayed learning near the end of the experiments, although this was not statistically significant. We attribute this to the lack of use of a masking light meaning the mouse could possibly use the light from the µLED array as a visual cue to lick the sensor.

Fig. 5.20 shows the equivalent lick raster plots for the discrimination experiment, with the no-go stimulus now introduced.

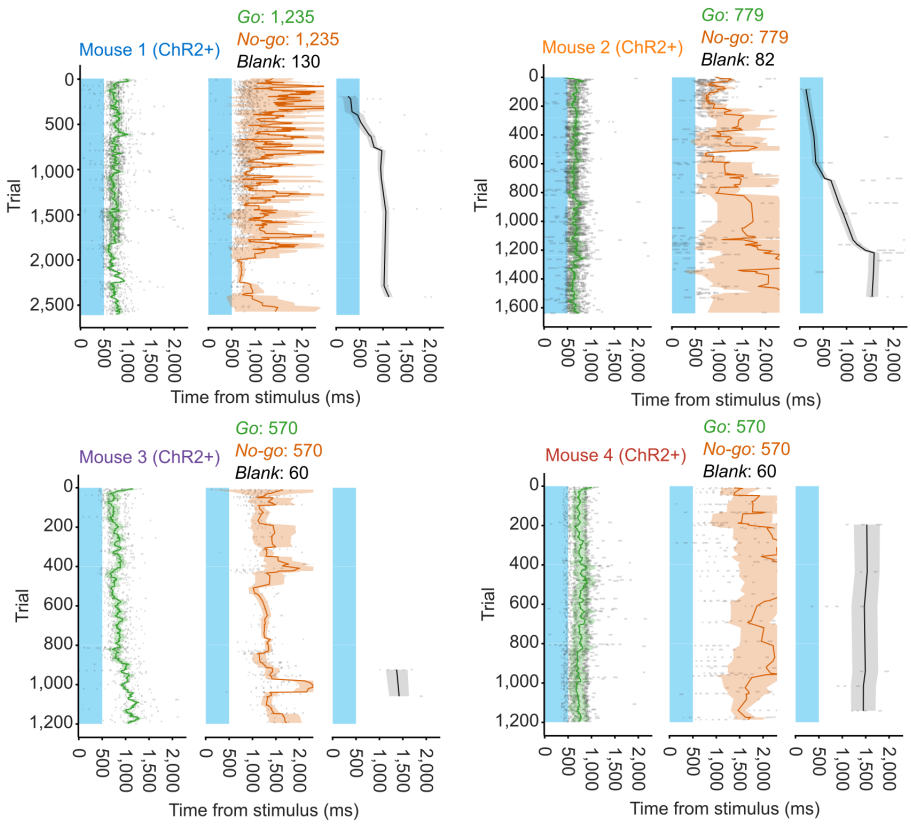


Figure 5.20: **Lick raster plots for discrimination experiment.** In all plots, shaded region shows mean and standard deviation of lick times per trial.

These show the mice were able to detect the *go* stimulus and ignore the *no-go*, indicating different perceptions elicited by optogenetic stimulation with the µLED array.

To evaluate learning performance, the accuracy is computed by the successful trials divided by the total number of trials:

$$\text{Accuracy}_{\text{association}} = \frac{N_{\text{go, success}}}{N_{\text{go, total}}} \tag{5.2}$$

$$\text{Accuracy}_{\text{discrimination}} = \frac{N_{\text{go, success}} + N_{\text{no-go, success}}}{N_{\text{go, total}} + N_{\text{no-go, total}}} \tag{5.3}$$

To qualify a trial as successful, we evaluate a background licking rate from the inter-trial period during an equivalent time frame to the response window. If the number of licks during the response window is greater than 2σ from the average background lick rate over the session then this trial is counted as a trial where the mouse intended to lick. If the licks are above this average background in a session for a *go* trial, it is successful, and if the licks are below this in a *no-go* trial, then it is successful; rewards are only issued for *go* trials (Fig. 5.18). This was necessary because very few trials contain no licks so taking a background lick rate allows us to eliminate random, non-deliberate behavior by the mouse to influence the accuracy. Using a background lick rate is likely to underestimate the accuracy because the licks during successful trials are not superimposed on the background lick rate – the mouse licks more frequently so the number of licks is higher.

Using Equations 2 and 3, learning curves were plotted showing the accuracy over the duration of experimental sessions. Bar plots show the average lick count per trial for each session (green: *go*, red: *no-go*, black: inter-trial period used for determining background lick rate).

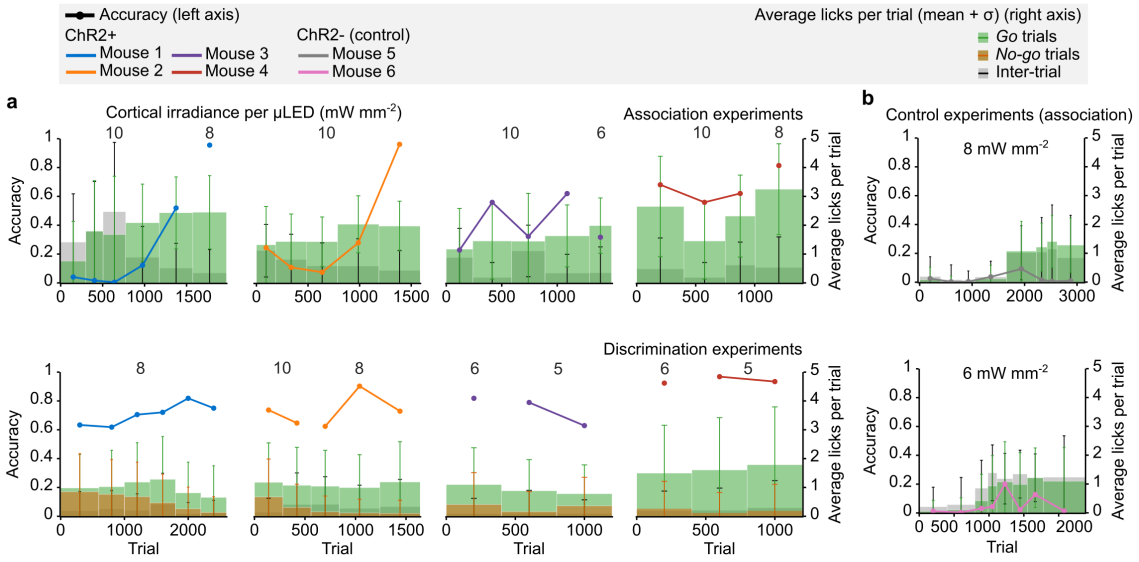


Figure 5.21: **Learning curves. (a)** Learning curves for association (top) and discrimination (bottom) experiments for ChR2+ mice. Cortical surface irradiance per µLED shown above corresponding sessions. **(b)** Same as (a) but for ChR2- mice.

From the learning curves, it can be seen that the mice consistently learn both association and discrimination experiments at comparable cortical surface irradiance levels to those determined as the stimulation threshold from electrophysiology (approx. 5 mW/mm^2). This

is presented over a broader surface area illuminated by a 3x3 pattern of μLEDs in all cases. Control mice do not learn the association task, again confirming no confounding effects such as thermally-induced neuronal firing or perception of light through the back of the retina, influencing behavioural outcomes. Accuracy within the 500 best consecutive trials range again shows that the mice successfully learned both association and discrimination experiments. Although, there is still statistical significance between the *no-go* and *blank* stimulus (0.01<p<0.05) but this is weaker than the *go* versus *blank* (p<0.001).

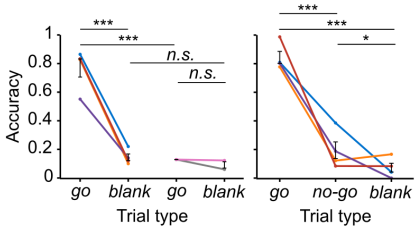


Figure 5.22: **Overall accuracy for 500 best consecutive trials**. Association experiment (left) and discrimination experiment (right).

Error bars show SEM over mice with statistical significance levels determined using the independent two-sample, single-tailed Student's t-test with alt. hypotheses: *go* accuracy > *no-go* or *go* accuracy > *blank* accuracy and/or ChR2+ accuracy > ChR2- accuracy. ***p < 0.001, **p < 0.01 *p < 0.05, n.s. means alt. hypothesis rejected at p<0.05. ChR2+: n = 4 mice, ChR2-: n = 2 mice.

5.2.2 Freely-behaving experiments

Freely-behaving experiments are based on an identical experimental paradigm to the head-fixed, whereby mice are trained to respond differently to optogenetic stimulation of distinct and separated regions of the auditory cortex through 3x3 patterns of illuminated μLEDs. Although in this case, the mice perform the task while being able to move freely, with only the lightweight cable acting as a tether. Here, fewer mice (n = 4) were used as the experimental protocol had largely been validated in the head-fixed experiments, and the control was deemed not necessary for the same reason. Experiments take place in a chamber with an IR sensor and lickport at one side (Fig. 5.23a). To begin a trial, the mouse approaches and inserts its snout into the IR beam, which is detected, beginning the trial and immediately triggering either the *go* or *no-go* stimulus. A one second response window opens, during which the mouse must remain within the IR beam for a cumulative time of >75% of the response window on a *go* trial to receive a reward. For *no-go* trials, the mouse must leave the beam before the 75% cumulative time elapses. Fig. 5.23b illustrates the experimental protocol. Mice are water deprived, motivating them to perform the task to receive water rewards.

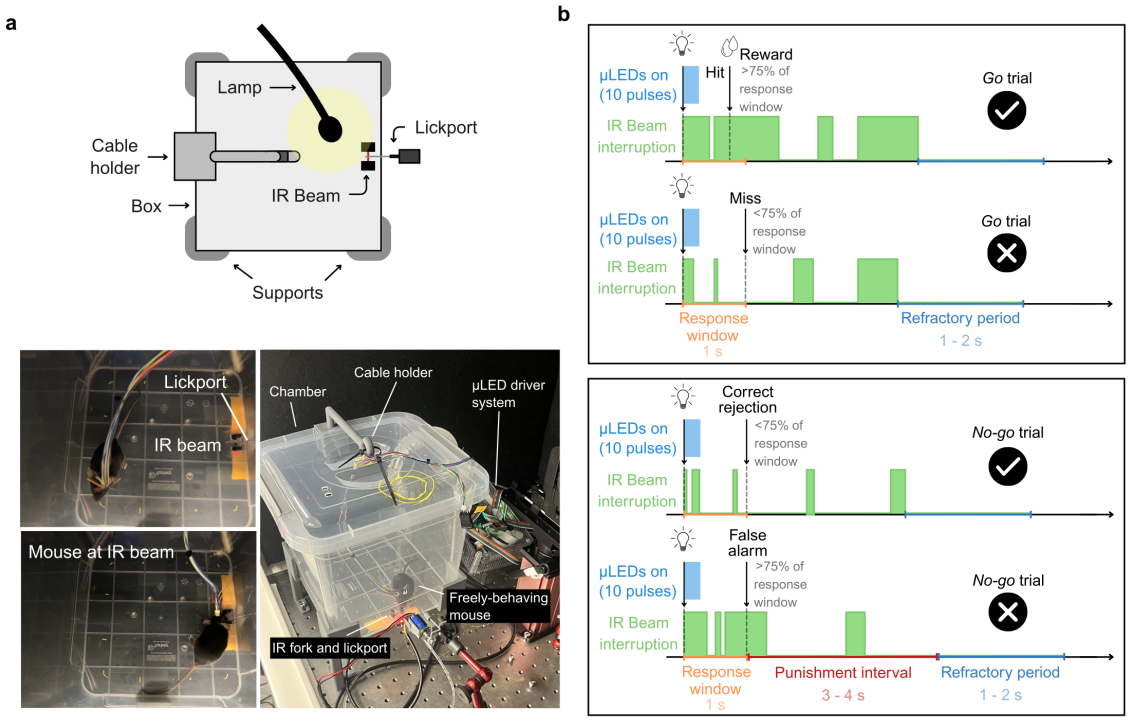


Figure 5.23: **Setup for freely-behaving discrimination experiment. (a)** Schematic of behavioural chamber (top) and photographs of chamber with mouse performing task (bottom). **(b)** Timing diagrams for freely-behaving experiment. The mouse must remain outwith the IR beam for the refractory period before the next trial can start; illustrated by the unlabelled region on the time axis prior to the refractory period.

Unlike head-fixed behaviour and electrophysiology experiments which used transgenic mice expressing ChR2, these mice were instead transfected with AAV vectors encoding ChRmine under the control of the CaMKIIα promoter which were injected to drive expression in pyramidal neurons of the auditory cortex [4]. Since ChRmine has much higher photocurrents than ChR2, blue light illumination is sufficient to elicit robust enough neuronal activation for behaviour.

The following plots are included for completeness, although all data collection and analysis was carried out within the Bathellier lab (Eya Mabrouk; see Acknowledgements).

Fig. 5.24 shows the cumulative time spent in the IR beam for each trial type between an earlier and later session (after the mouse has learned the task) for all four mice.

Similar to before, we plot learning curves along with average performance per trial for each session, computing accuracy in the same way as before (Fig. 5.25).

Finally, IR beam interruption within the trial over the course of the response window is shown (Fig. 5.26). Again, this compares the first session with a later session after the mouse has learned.

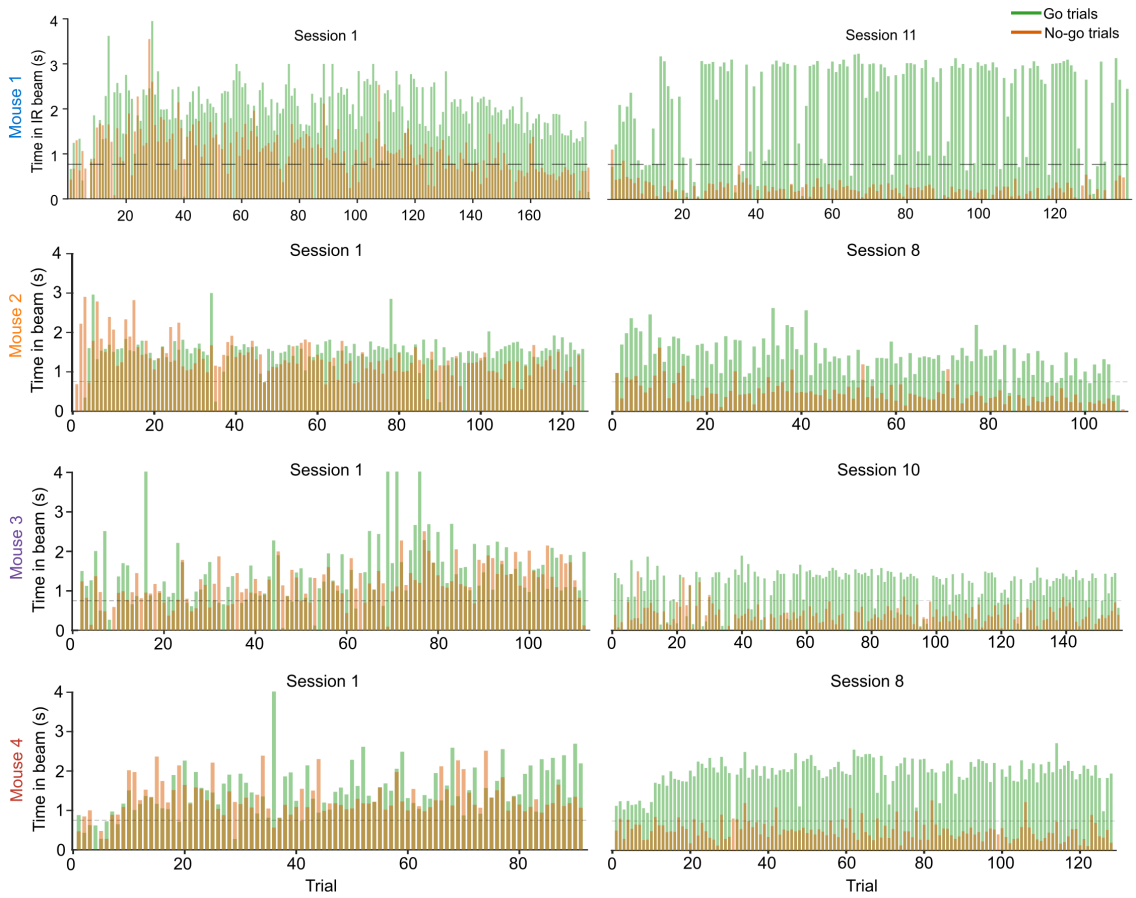


Figure 5.24: **Cumulative time spend in IR beam for each trial type**. First session (left) and session after which mouse has learned the task (right).

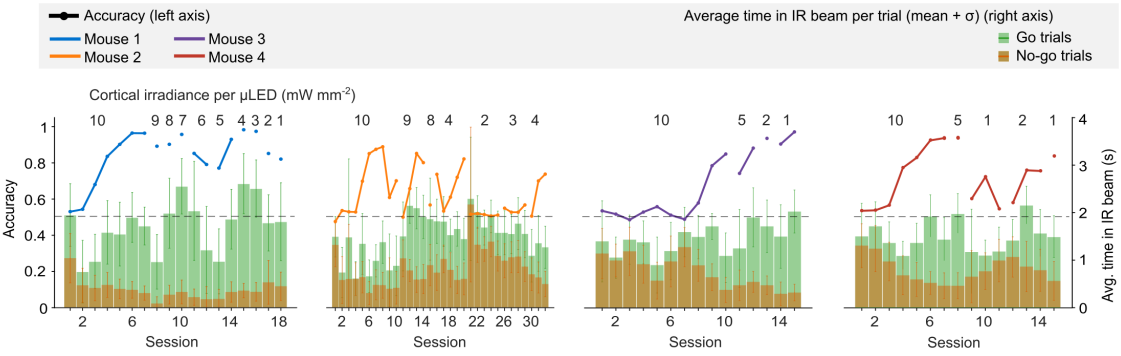


Figure 5.25: **Learning curves from freely-behaving experiments**. These show average time spent in IR beam per trial across sessions (bar plots, right axis) and corresponding accuracy as percentage of trials correct (points/lines, left axis). Cortical irradiance per µLED given above corresponding sessions.

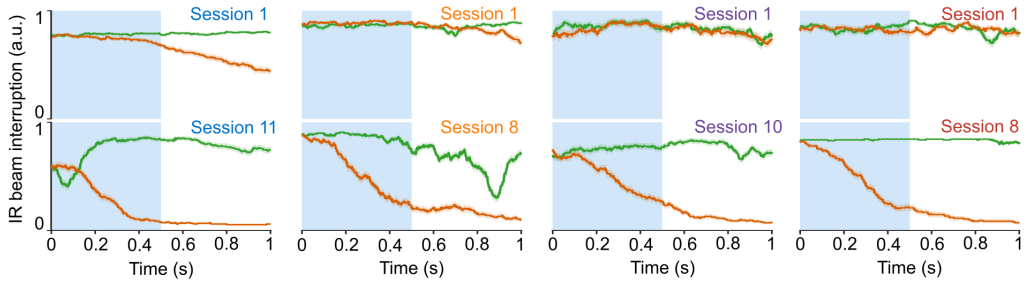


Figure 5.26: **Mean and standard deviation of intra-trial presence in IR beam across trials**. First session (top) and session after which mouse has learned the task (bottom).

All mice were able to learn the freely-behaving task, and maintain accuracy at lower cortical surface irradiances than in the head-fixed case. This aligns with ChRmine's enhanced sensitivity to blue light versus ChR2, despite being a red-shifted opsin [4].

5.3 Histology

Despite the chronic packaging being flexible, the stimulation area of the µLED array remains rigid, constrained by the GaN-on-sapphire fabrication process; the brain locally adapts to this constraint in the same way as it does as for cranial window implantations [19]. Histological analysis demonstrates plastic deformation around the implant site, with the size of the affected region comparable to the geometry of the µLED array (Fig. 5.27). Immune response was then quantified by examining microglial build-up from the surface through the auditory cortex (Fig. 5.27g). This was statistically significant when comparing the Iba-1+ cell response from the implanted side (228 ± 24 per mm^2) to the non-implanted (contralateral) side (181 ± 18 per mm^2). Excluding layer 1 from the analysis shows no significant difference between the implanted side (200 ± 34 per mm^2) and contralateral side (133 ± 11 per mm^2), highlighting a small localised immune response, approximately limited to layer 1 of the mouse cortex. Histology was carried out by our collaborators (Magdalena Pereyra; see Acknowledgements).

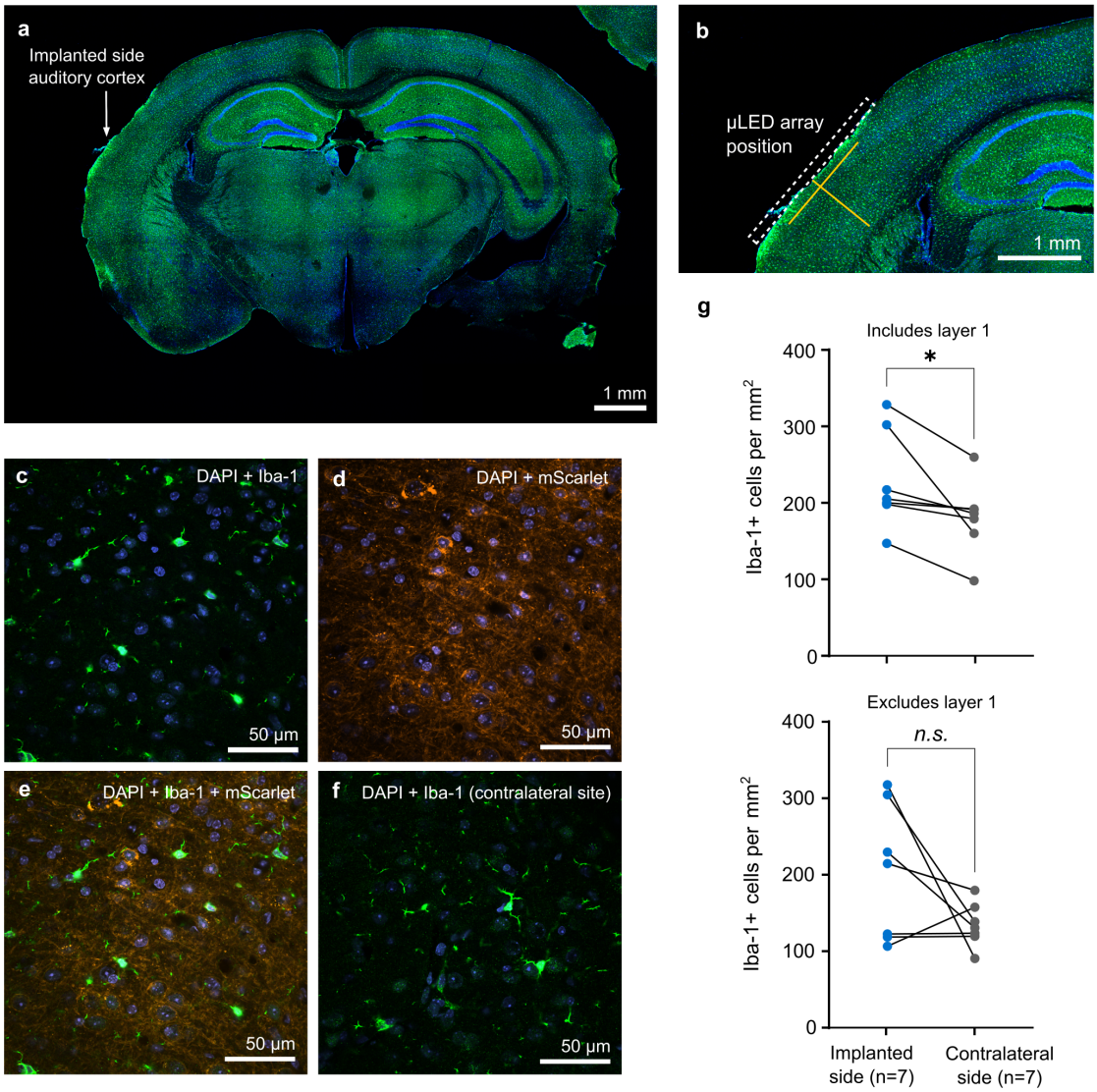


Figure 5.27: **Histological analysis. (a)** Coronal view of brain slice stained with DAPI and Iba-1, with auditory cortex on implanted side indicated. **(b)** Zoomed in view of implanted auditory cortex; µLED array geometry shown as dashed outline; layer 1-2/3 boundary and depth of analysed region indicated by solid yellow lines. **(c)** Zoomed example of implanted region analysed to examine immune response (staining: DAPI + Iba-1). **(d)** Same region as (c) (DAPI + mScarlet). **(e)** Same region as (c) and (d) (DAPI + Iba-1 + mScarlet). **(f)** Similar example region for contralateral side (DAPI + Iba-1). **(g)** Immune response as number of Iba-1+ cells (microglia) per square millimetre of imaged tissue, comparing both implanted and contralateral sides across n = 7 coronal slices for 3 mice; non-parametric Wilcoxon matched-pairs signed rank test. Layer 1 was removed from both sides to exclude surface immune response from the analysis (bottom plot).

5.4 Current and future directions

A potential next step for behavioural experiments using the implanted μLED arrays is to reduce the size of illuminated pattern, as well as observing if there is a spatial limitation in discrimination performance. Preliminary experiments were carried out using the same freely-behaving experimental setup with single-pixel stimulation at 10 mW/mm^2. Over the course of sessions, the *no-go* stimulus was moved closer to the *go* stimulus, and we observed a corresponding reduction in performance (Fig. 5.28).

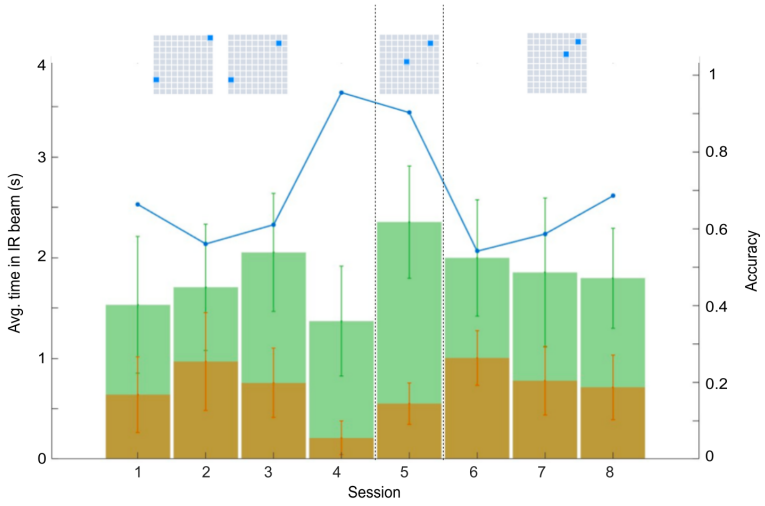


Figure 5.28: **Learning curve showing discrimination performance for a single-pixel task**. Separation between stimulation regions is progressively reduced and mouse's performance measured.

More experiments are needed (n = 1) although this indicates some spatial resolution we need to be aware of when transferring information to the brain optically. This result appears to coincide with the overlapping activation fields observed through the optical modelling, which has also been validated by electrophysiology experiments. Fig. 5.29 shows the equivalent optical model with the 1 mW/mm^2 threshold shown for the *go* and *no-go* experiments in each case.

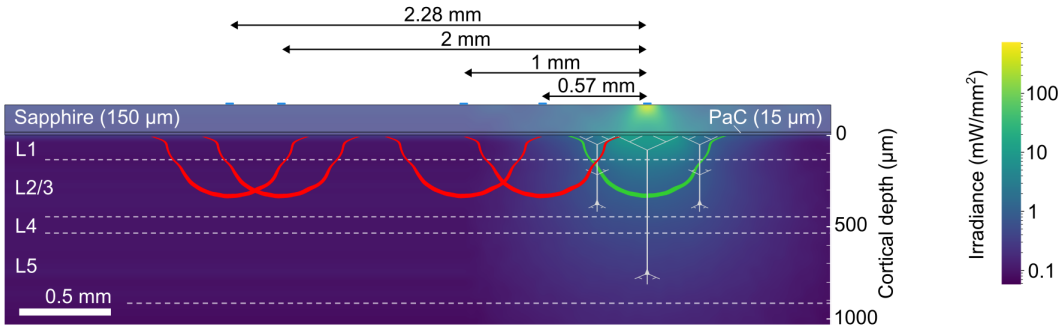


Figure 5.29: **Optical modelling showing overlap of above-threshold irradiance regions for single-pixel discrimination experiment.** Green contour corresponds to *go* µLED and red contour to *no-go* µLED. The model assumes the µLED is in the middle of the sapphire, which is why the sapphire extends to larger than the actual device in the schematic. 1 mW/mm^2 was chosen as an approximation, despite using ChRmine instead of ChR2 in this experiment.

Although, the mechanism for activation is still unclear, as responses may be the result of activation through the dendrites, soma, or other network effects. Therefore, reduction of the cortical irradiance to the behavioural threshold, could help to elucidate this, and further validate the accuracy of the optical modelling. Overall, the *in-vivo* experiments, outlined in this section, have successfully demonstrated stable neuronal and behavioural responses, opening avenues for future experimentation into more sophisticated information transfer methodologies to the cortex.

Bibliography

1. Greer, R. *et al.* Chronically implantable LED arrays for optogenetic cortical surface stimulation in mice. en. *Nature Communications* **17,** 878. ISSN: 2041-1723. doi:10.1038/s41467-025-68191-5. (2026) (Jan. 2026).
2. Gorski, J. A. *et al.* Cortical Excitatory Neurons and Glia, But Not GABAergic Neurons, Are Produced in the Emx1-Expressing Lineage. en. *Journal of Neuroscience* **22,** 6309–6314. ISSN: 0270-6474, 1529-2401. doi:10.1523/JNEUROSCI.22-15-06309.2002. (2026) (Aug. 2002).
3. Madisen, L. *et al.* A toolbox of Cre-dependent optogenetic transgenic mice for light-induced activation and silencing. en. *Nature Neuroscience* **15,** 793–802. ISSN: 1546-1726. doi:10.1038/nn.3078. (2026) (May 2012).
4. Marshel, J. H. *et al.* Cortical layer–specific critical dynamics triggering perception. *Science* **365,** eaaw5202. doi:10.1126/science.aaw5202. (2026) (Aug. 2019).
5. Karimi, A., Odenthal, J., Drawitsch, F., Boergens, K. M. & Helmstaedter, M. Cell-type specific innervation of cortical pyramidal cells at their apical dendrites. *eLife* **9** (eds Mason, C. A. & Marder, E.) e46876. ISSN: 2050-084X. doi:10.7554/eLife.46876. (2025) (Feb. 2020).
6. Park, S. I. *et al.* Soft, stretchable, fully implantable miniaturized optoelectronic systems for wireless optogenetics. eng. *Nature Biotechnology* **33,** 1280–1286. ISSN: 1546-1696. doi:10.1038/nbt.3415 (Dec. 2015).
7. Shin, H. *et al.* Transcranial optogenetic brain modulator for precise bimodal neuromodulation in multiple brain regions. en. *Nature Communications* **15,** 10423. ISSN: 2041-1723. doi:10.1038/s41467-024-54759-0. (2026) (Nov. 2024).
8. Kim, T. *et al.* Thermal effects on neurons during stimulation of the brain. en. *Journal of Neural Engineering* **19,** 056029. ISSN: 1741-2552. doi:10.1088/1741-2552/ac9339. (2025) (Oct. 2022).

9. Owen, S. F., Liu, M. H. & Kreitzer, A. C. Thermal constraints on in vivo optogenetic manipulations. en. *Nature Neuroscience* **22,** 1061–1065. ISSN: 1546-1726. doi:10.1038/s41593-019-0422-3. (2026) (July 2019).
10. Ludwig, K. A. *et al.* Using a Common Average Reference to Improve Cortical Neuron Recordings From Microelectrode Arrays. *Journal of Neurophysiology* **101,** 1679–1689. ISSN: 0022-3077. doi:10.1152/jn.90989.2008. (2025) (Mar. 2009).
11. Lin, J. Y. A user's guide to channelrhodopsin variants: features, limitations and future developments. en. *Experimental Physiology* **96,** 19–25. ISSN: 1469-445X. doi:10.1113/expphysiol.2009.051961. (2026) (2011).
12. Aravanis, A. M. *et al.* An optical neural interface: in vivo control of rodent motor cortex with integrated fiberoptic and optogenetic technology. en. *Journal of Neural Engineering* **4,** S143. ISSN: 1741-2552. doi:10.1088/1741-2560/4/3/S02. (2025) (June 2007).
13. Burns, S. P., Xing, D. & Shapley, R. M. Comparisons of the Dynamics of Local Field Potential and Multiunit Activity Signals in Macaque Visual Cortex. en. *Journal of Neuroscience* **30,** 13739–13749. ISSN: 0270-6474, 1529-2401. doi:10.1523/JNEUROSCI.0743-10.2010. (2025) (Oct. 2010).
14. Choi, Y.-S., Koenig, M. A., Jia, X. & Thakor, N. V. Quantifying Time-Varying Multiunit Neural Activity Using Entropy-Based Measures. *IEEE transactions on bio-medical engineering* **57,** 10.1109/TBME.2010.2049266. ISSN: 0018-9294. doi:10.1109/TBME.2010.2049266. (2025) (Nov. 2010).
15. Wu, M. *et al.* Patterned wireless transcranial optogenetics generates artificial perception. en. *Nature Neuroscience,* 1–12. ISSN: 1546-1726. doi:10.1038/s41593-025-02127-6. (2025) (Dec. 2025).
16. Ceballo, S., Piwkowska, Z., Bourg, J., Daret, A. & Bathellier, B. Targeted Cortical Manipulation of Auditory Perception. eng. *Neuron* **104,** 1168–1179.e5. ISSN: 1097-4199. doi:10.1016/j.neuron.2019.09.043 (Dec. 2019).
17. Verdier, A. *et al.* Enhanced perceptual task performance without deprivation in mice using medial forebrain bundle stimulation. eng. *Cell Reports Methods* **2,** 100355. ISSN: 2667-2375. doi:10.1016/j.crmeth.2022.100355 (Dec. 2022).
18. Charan, J. & Kantharia, N. D. How to calculate sample size in animal studies? *Journal of Pharmacology & Pharmacotherapeutics* **4,** 303–306. doi:10.4103/0976-500X.119726 (2013).
19. Holtmaat, A. *et al.* Long-term, high-resolution imaging in the mouse neocortex through a chronic cranial window. *Nature protocols* **4,** 1128–1144. ISSN: 1754-2189. doi:10.1038/nprot.2009.89. (2025) (2009).

Chapter 6

Advanced systems for closed-loop cortical stimulation

Systems engineering is critically important in the design and execution of *in-vivo* experimentation using novel stimulation technologies, such as those described in previous chapters. Therefore, a key focus of this project has been to develop robust, portable and software-controllable driver systems. This chapter first details the architecture for the microcontroller-based system used throughout all *in-vivo* experiments (v1), capable of operating the µLED arrays with the required spatial and temporal patterns. A newer version of this system (v2) has since been developed which supercedes all the functionality of v1. Hardware designs and software code listings may be found in Appendices B and C, respectively.

The design of system v1 was published in [1].

Table 6.1 summarises the core functionality of both systems.

	System v1	**System v2**
µLED device size	Up to 10×10 array	2 10×10 arrays or up to 20×20 array
Recording capability	No	Yes – Intan RHD2000
Peripherals	Limited	Audio/video etc…
Programming environment	LabVIEW	Python
Embedded machine learning	No	Yes, tensorflow lite etc...

Table 6.1: **Control system comparison and features**

6.1 Microcontroller-based current driver system for µLED cortical implant (v1)

Although this system has since been superceded by v2, a detailed description of the v1 architecture will be provided in this short section, since it has been used extensively throughout

all *in-vivo* experiments. The general requirement was a system with independent current control for the µLEDs and with the capability to produce different spatial and temporal patterns required for *in-vivo* electrophysiology, and later, behavioural experiments. The system should also be reliable and easy to operate through a software environment.

Fig. 6.1a summarises the system architecture that was already briefly introduced in Chapter 5, Fig. 5.1. 10 independently operable current supplies connect to the p tracks of the device, while the 10 n tracks can be switched to ground. This allows any of the 100 µLEDs across the device to be addressed along with adjustable brightness. Due to matrix addressing, arbitrary patterns are limited unless time-division multiplexing (TDM) is employed. A microcontroller operates the described circuit and facilitates communication to a host PC providing system control via a user interface.

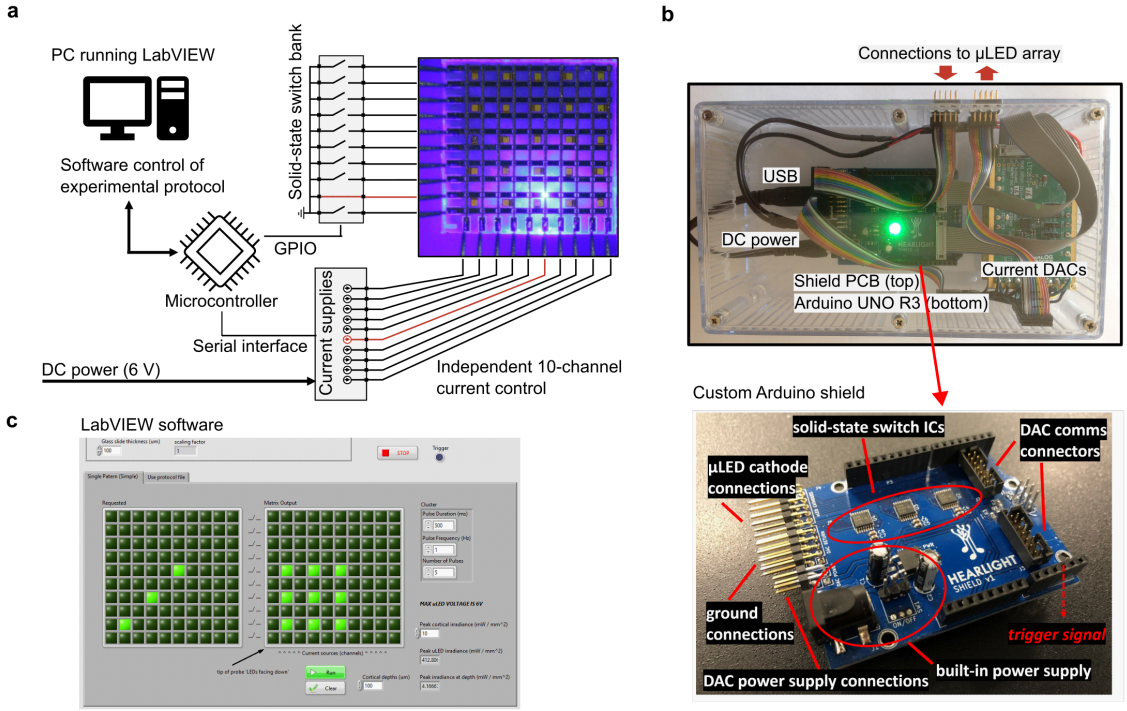


Figure 6.1: **Control system v1 overview. (a)** High-level schematic showing electronic control of µLED arrays. The GPIO and serial interface (SPI) are connected to the Arduino header pins. **(b)** Annotated photograph of boxed control system (top). Annotated photograph of custom Arduino shield PCB housing switch circuitry (bottom). **(c)** Screenshot of LabVIEW user interface.

A photograph of the system is shown in Fig. 6.1b. An Arduino UNO R3 microcontroller was selected as they are easy to program and to operate with LabVIEW via a serial (UART/USB) port. A custom shield PCB (Fig. 6.1b bottom) plugs into the Arduino and contains the switch ICs (TMUX7211) as well as connector ports allowing two LTC2662 5-channel current DAC (digital-analog converter) chips (each mounted on a DC2903a development board), containing

the 10 current supplies to be connected. The current DACs are controlled over an SPI (serial peripheral interface), allowing simple programming with the Arduino. Otherwise, the boxed system contains cables and connectors on the side, allowing the µLED array to be plugged in. A separate ribbon cable (acute device) or flexible cable (chronic device) allows the device to be plugged into the pins on the side of the controller box. A digital pin on the Arduino can act as a trigger such that the onset and offset of µLED stimulation can be correlated with the recorded electrophysiology signal. The system can receive a trigger to drive a stimulation, such as that used in behavioural experiments.

As discussed, the system uses LabVIEW, allowing straightforward programming of pre-defined routines (used in the electrophysiology experiment) or to specify a go/no-go pattern (for behavioural experiments) (Fig. 6.1c). The software interprets the requested pattern and current for the DACs (usually calibrated from requested cortical irradiance as described in Chapter 4) as well as the required pulse timing (number of pulses, duty cycle and refresh rate). A data packet is assembled representing this single stimulation protocol, and transmitted over serial to the Arduino, to be executed immediately or upon detection of a trigger signal. This simple dataflow – a single pattern, intensity and pulse profile at a time – allows the user to easily write custom programmes in LabVIEW without having to understand or modify the microcontroller code. As an example, for our behavioural experiments, a single trial may be a *go* pattern; 10 pulses of a 3x3 pattern of µLEDs at 5 mW/mm^2. This would be a single stimulation protocol or data packet to be sent from LabVIEW to the microcontroller.

The first iteration of the controller system served well in the laboratory over several years of *in-vivo* experimentation. However, it has since been deprecated in favour of the system described next (v2), which has improved features and functionality, although in principle operates in a similar way. The new system does not incorporate separate development boards for the current DAC chips as with v1; initially this was done due to time constraints and occasionally posed some reliability issues due to the many interconnections.

6.2 FPGA-enabled system for closed-loop optogenetic control and integration of electrophysiology recording

The current version of the control system (v2) is similar to v1 in terms of its ability to operate the µLED arrays through independently controllable current supplies (for the p tracks) and a switch matrix to ground (for the n tracks). However, this capability is extended to 20 current supplies and a 20-channel switch matrix, allowing operation of two 10x10 µLED arrays simultaneously, or one larger µLED matrix like the Utah optrode array (UOA) [2]. Furthermore, this system has the capability to record neuronal data from Intan Technologies RHD2000 headstages [3], which are ubiquitous in neuroscience laboratories. Since the system incorporates both neuronal

recording and optogenetic stimulation circuitry, closed-loop experimentation is possible [4]. This extended functionality, as well as a host of other useful features, is enabled by the PYNQ (Python on Zynq) framework [5], and specifically the PYNQ-Z2 development board utilised in the design.

The PYNQ-Z2 development board incorporates a Zynq system-on-chip (SoC) from AMD (formerly Xilinx). The Zynq SoC contains an FPGA (field-programmable gate array, equivalent to an AMD Artix-7 fabric) commonly referred to as the programmable logic (PL), and a dual-core ARM A9 CPU, commonly referred to as the processing system (PS). These are fabricated on a single die, thereby offering the flexibility of software programming with the signal processing and performance capabilities of hardware. In PYNQ, the PS of the Zynq can be programmed with Python (hence the name) through a Jupyter Notebook interface, hosted by the board, but accessed through the browser on the connected PC (i.e. Google Chrome). This makes the barrier to entry very low for programming the PS as Python/Jupyter are highly user-friendly and no additional software needs to be installed, therefore avoiding expensive software licences. Programming the FPGA/PL is done using traditional hardware description language (HDL), high-level synthesis or block-based approaches (ultimately compiling to HDL) before carrying out synthesis and implementation flows through tools like AMD Vivado Design Suite. Fortunately, the PYNQ framework comes with a "base" overlay (overlay = bitstream/FPGA configuration file). This base overlay allows the PS to access all the features of the PYNQ-Z2 board connected to the PL through an Advanced eXtensible Interface (AXI-4) [6] and user-friendly Python APIs. For example, once the base overlay is loaded to the PL, one can access the Arduino pin-header pins and operate them as GPIOs (general purpose input-outputs) through simple Python calls, making the board easy to interface with the LED driver shield without needing to develop FPGA code, which is often time-consuming and complex.

Fig. 6.2a shows a high-level overview of the hardware. A photograph of the PYNQ-Z2 board with new shield PCB is shown (Fig. 6.2b). Fig. 6.2c shows a more detailed view of which aspects are implemented in the PL and PS of the Zynq, which will be expanded in the next section.

As discussed, the LED driver circuitry is largely the same as before although this time four LTC2662 5-channel DAC chips permit control of 20 p-channels. The necessary low-noise 5 V analog supply was also implemented on the shield using an LDO (low-dropout regulator). V+ for the current supplies is provided through the Arduino shield pin and is connected to the DC power adaptor which dictates voltage compliance for the LEDs; typically this will be 7.5 V for our devices. A 20-channel switch matrix to ground was implemented using the same high-power solid-state switch ICs as before (TMUX7211) controlled through a shift register to reduce GPIO overhead. All of the LED driver circuitry is connected through the Arduino

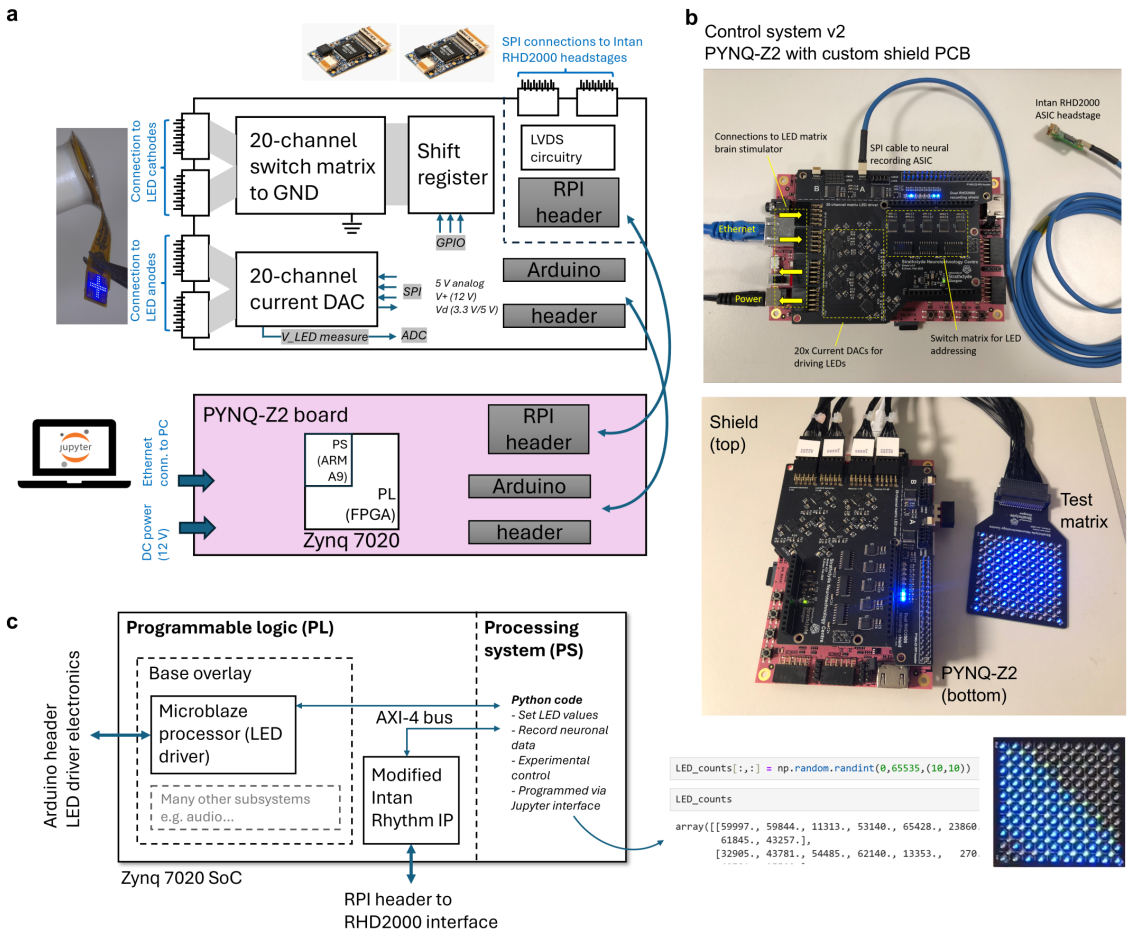


Figure 6.2: **Control system v2 overview. (a)** High-level schematic of control system hardware; shield (top), PYNQ-Z2 board (bottom). **(b)** Photographs of control system with custom shield (black). Intan RHD2000 headstage connected (top), custom UOA LED test matrix being driven (bottom). **(c)** High-level schematic of system-on-chip (SoC) design showing key FPGA blocks and processes. Example of Python code operating LED matrix (right).

header on the PYNQ-Z2 board. The board also contains two 12-pin Omnetics connectors compatible with the SPI cables for Intan RHD2000 neural recording ASIC (application-specific integrated circuit) headstages. These headers connect directly into the PL fabric where the SPI data streams are decoded in real-time (electrophysiology design shown in following section). Low-voltage differential signalling (LVDS) can be optionally enabled via a jumper, meaning LVDS and/or non-LVDS headstages can be used with this system. This design was based on Intan's LVDS breakout board.

The following sections describe the software and FPGA designs for the control system. Full code listings may be found in Appendix C, as well as Jupyter Notebook demonstrations showcasing the functionality and features of the system.

6.2.1 LED driver software

As discussed, the system utilises Python to operate the processing system (PS). Due to the base overlay, control of the LED driver part of the shield is possible without writing FPGA code. Instead, Microblaze processors – soft RISC-V processor cores implemented in the PL fabric – are programmed in C similarly to the Arduino programming done for system v1. Fig. 6.3 gives a high-level overview of the operation of the LED driver.

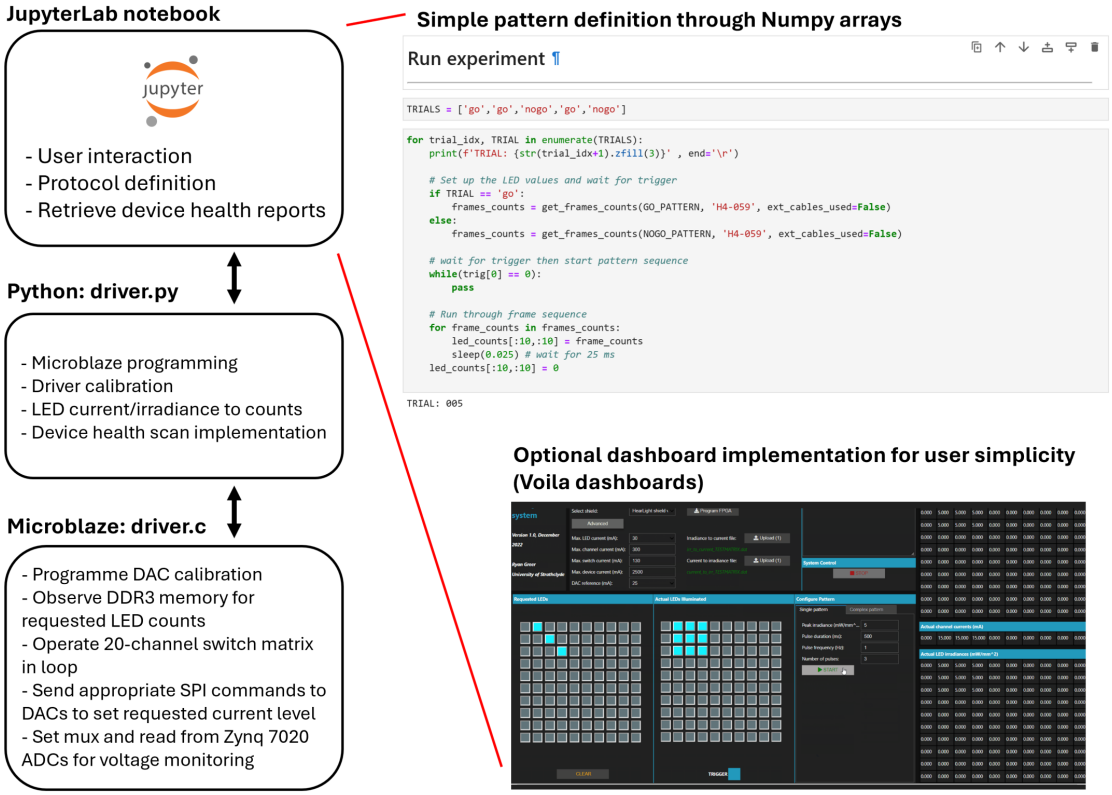


Figure 6.3: **High-level overview of operation of LED driver**. The elements implemented in PS backend and PL are shown, along with user interaction through Jupyter Notebooks.

The C driver operates continuously and monitors memory allocated in DRAM by the PS (512 MB off-chip memory). This 100-element memory represents the "counts" values for the DAC for each µLED location. The driver then cycles through each row (n track) closing the respective switch to GND (Fig. 6.1) and assigns the DAC counts value. The entire matrix is cycled in less than 10 ms giving arbitrary pattern generation, albeit at a 10% duty cycle. The python part of the driver (.py file) contains helper functions to programme the Microblaze and convert cortical irradiance to current, and current to DAC counts. The user interacts with the driver through the Jupyter Notebook and simply allocates and assigns values to the 10x10 matrix. These can be read from a file and changed in a loop with any arbitrary timing. This

allowed outputs from the Braincodec algorithm (programmed in PyTorch, a Python-based deep neural network tool) to be rapidly adapted into *in-vivo* experiments (described in more detail in Chapter 7).

Furthermore, the PYNQ-Z2 board contains other peripherals which can be used alongside the LED driver; namely, the audio subsystem (3.5 mm audio jack and codec IC). A demonstrator was put together which can read audio snippets and perform a discrete Fourier Transform (FFT algorithm), assigning power to frequency bins and to a template tonotopic map [7]. This then translates to brightness in the LED matrix. A graphical user interface was developed in Jupyter Notebooks to demonstrate this (Fig. 6.4). The resulting demonstration showed the LED matrix lighting in the regions corresponding to the tonotpic map (bottom-right of Fig. 6.4) depending on the frequency of the tone input to the audio port on the board, in real-time. This demonstration was programmed entirely in the PS (Python) utilising the base overlay to avoid complex FPGA workflows, highlighting the power of this platform to enable more sophisticated optogenetic *in-vivo* experiments.



Figure 6.4: **Example graphical user interface for audio recording closed-loop demonstration.** The interface is constructed from panels using Jupyter widgets (a Python library for developing HTML-like interfaces); these include configuration (top), audio configuration (bottom-left) and plotting of time/frequency series data in real-time (bottom-right).

6.2.2 Intan RHD interface FPGA design

The developed controller system offers two key modules; the LED driver described previously, and an interface to connect Intan RHD neural recording headstages for real-time electrophysiology recording. The former utilises the base overlay to enable control of the Arduino

pin-header which connects all of the current driver and switching circuitry, entirely through C and Python libraries. However, interacting with Intan headstages through the PYNQ environment requires dedicated FPGA programming for the PL, namely adapting Intan's design to use AXI-4 interfaces instead of USB3. The Verilog hardware description language and AMD Vivado Design Suite were used.

RHD headstages are typically operated via an Intan recording controller. This contains an FPGA (Artix-7 for their latest generation USB3 recording controllers) which decodes the SPI datastreams from multiple headstages in parallel, so hundreds of neural recording channels can be read simultaneously. The recording operates at a set sample rate and records a sample as a 16-bit number; for example we may record from 256 channels (4 64 channel headstages) at 30 kHz which generates a large quantity of data ($\sim$123 Mbits/s), motivating the use of an FPGA and high-speed USB3 interface. The Intan recording controller board connects to a PC running software, typically allowing the datastream to be saved to a file and analysed later, although tools exist for online analysis. Intan's FPGA design is called "Rhythm3" and is provided as open-source Verilog code [3]. In this project, the Rhythm3 code has been modified and integrated with the PYNQ base overlay such that both electrophysiology recording and the LED driver can be operated using the same bitstream and therefore simultaneously through the Python/Jupyter Notebook interface. The overall design was motivated by the desire to have closed-loop electrophysiology recording and optogenetic control.

Fig. 6.5 summarises the high-level framework of the FPGA design.

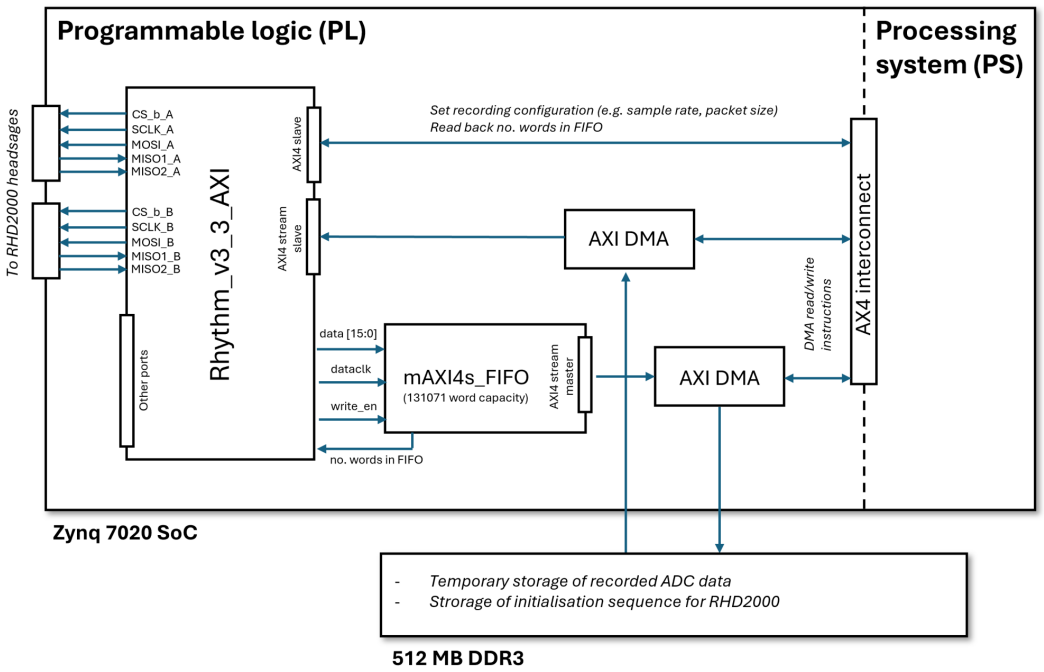


Figure 6.5: **High-level schematic of FPGA implementation of Intan Rhythm3 design modified for Advanced eXtensible Interface (AXI-4).** The Rhythm3 block connects directly to the SPI ports on the board, and generates command sequences and reads back responses from the RHD2000. This data is continually presented to a first-in-first-out (FIFO) buffer which must be read using direct memory access (DMA) from the PS, and is stored in off-chip DRAM. Interfacing between the PL and PS is done through the Advanced eXtensible Interface (AXI-4) protocol. In the diagram, blocks represent Verilog modules, and arrows represent digital signals within the FPGA.

Intan's Rhythm3 code was modified to remove the USB3 controller (for the Opal Kelly board) and replace this with an AXI-4 interface capable of communication with the PS in the Zynq 7020 chip and direct memory access (DMA) to the on-board DDR3 memory. Three interfaces were added:

- AXI-4 lite slave to allow writing to Rhythm registers (to modify the operation of the FPGA for example setting the state machine clock i.e. the rate of sampling from the ADCs).
- AXI-4 streaming slave which receives a list of commands to be written to the headstages, for example setting the initialisation sequence.
- Direct 16-bit digital output (FIFO data) - the actual recorded data samples, as well as timestamps and other optional inputs to the board.

A 16-bit digital signal outputs samples, which are then fed into a custom-designed circular first-in first-out (FIFO) buffer. The other side contains an AXI-4 streaming master port which

connects via DMA allowing the datastream to be written directly to off-chip DRAM. This DRAM is effectively acting as a large FIFO, where the data can then be accessed at the PS side and operated on in real-time and/or saved to the SD card of the PYNQ-Z2 for offline analysis. The FIFO has a capacity of 131071 16-bit words, limited by the FPGA block RAM (BRAM) that implements it. Table 6.2 summarises the terminology used to describe this system; these terms will be referred to in this chapter.

Term	**Description**	**Value range / default value**
f_s	Sample rate	Typically 1–30 kHz
Sample	Single sample from all channels	0–65535 per word
FIFO_capacity	Maximum number of 16-bit data words in FIFO	131071
no_words_in_FIFO	Number of 16-bit data words currently in FIFO	0–131071
N_data_streams_enabled	Number of MISO lines enabled (i.e. RHD2132 or RHD2164 chips connected)	1–32
block_size	Set of 16-bit data words that make up a single sample from all channels, including additional data such as magic number and timestamp. Calculated as $35 \cdot N\_data\_streams\_enabled + 16 + (N\_data\_streams\_enabled \bmod 4)$	i.e. for 2 data streams $\Rightarrow$ 88
N_samples_packet	Number of samples per packet. A packet represents a single AXI4-streaming transfer from the FIFO. For the fastest data readout, this should be set to 1. Otherwise, it can be set higher to free PS resources for other operations.	≥ 1
packet_size	Number of 16-bit data values per packet	$N\_samples\_packet \cdot block\_size$

Table 6.2: **Nomenclature for real-time electrophysiology recording system.**

To verify the functionality of the system, and that it has sufficient performance, i.e. no/minimal lost data packets, an experiment was set up to record from electrodes placed in conductive solution (phosphate buffered saline, PBS) (Fig. 6.6a). 32 electrodes (using the "hybrid" device described in Chapter 3) were connected via an RHD2164 headstage, operating

in LVDS mode, which was connected to the controller system via SPI cable. The electrode device and beaker of PBS were placed in a Faraday cage. The Faraday cage was used to mitigate electrical interference on the recordings to make the analysis easier, although clean electrophysiology-grade recordings were not strictly required for the purposes of these experiments. A National Instruments DAQ was connected to a platinum electrode placed nearby the recording electrodes to provide a test signal (250 Hz sine wave). Fig. 6.6b shows example recorded waveforms at different sample rates, captured within the Jupyter Notebook, which show the 250 Hz signal.

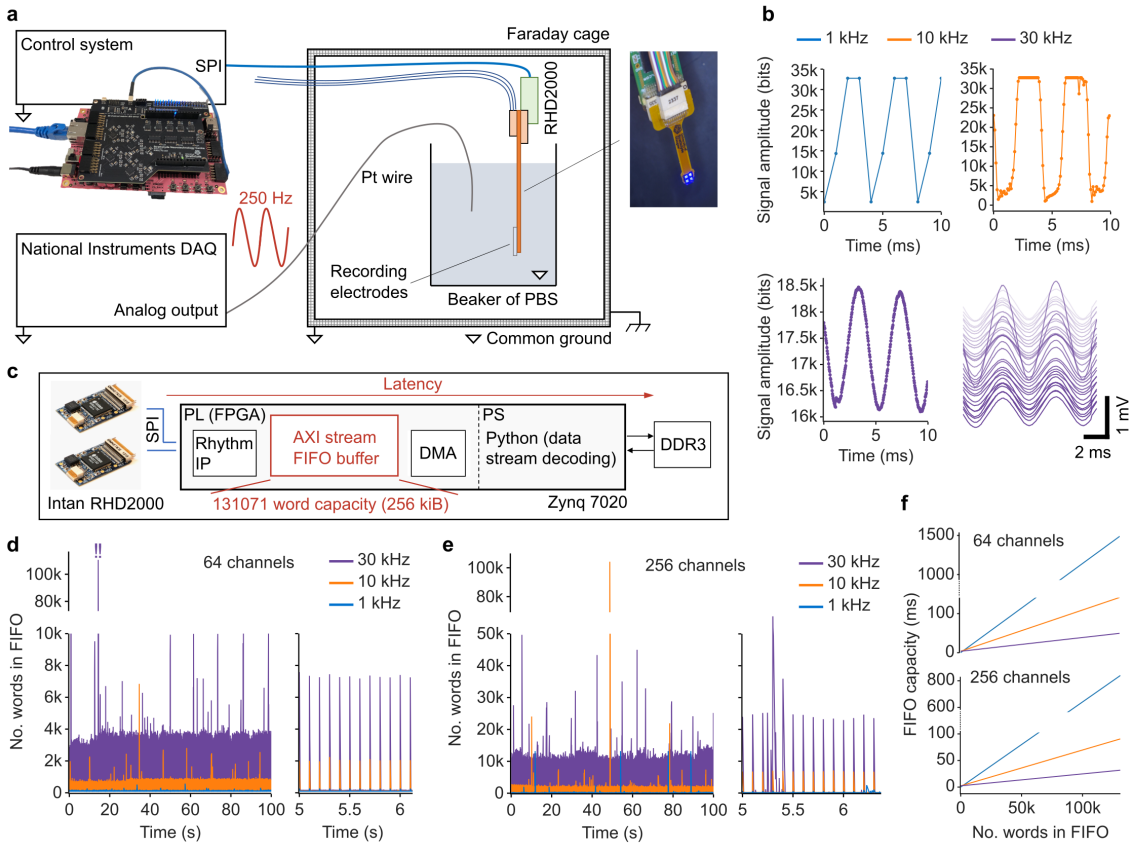


Figure 6.6: **Validation of electrophysiology recording system FPGA design (a)** Schematic of experimental setup. **(b)** Example waveforms from single channel showing 250 Hz signal, at different sample rates. For two of the recordings (1 kHz and 10 kHz), poor grounding resulted in saturated amplifiers, however the system was still able to record the digitised signal. For the 30 kHz recording, all channels are shown (bottom-right). **(c)** Block diagram of data transmission system, highlighting AXI-4 streaming master FIFO being stress-tested. **(d)** Number of words in FIFO over duration of experiment at different sample rates (64 channels enabled). The two exclamation marks indicate two timepoints where the buffer reached full capacity and had to reset, each corresponding to ∼50 ms of lost data. **(e)** Same as (d) for 256 channels enabled. **(f)** Amount of data stored in FIFO in milliseconds, as a function of the number of 16-bit words stored. This depends on the sample rate and number of channels enabled.

With any recording system, if the data is not read fast enough there will be data loss due to buffer overflow. In this case, and with Intan's recording controller, there is a buffer which cannot be allowed to fill otherwise data will be lost. The AXI4-master FIFO developed here can report the number of words currently stored through the register map, available in real-time through a Python function call. The number of words was monitored at 1 ms intervals (packet size dependent on sample rate, but always corresponding to 1 ms) while data were recorded at different sample rates (1, 10, 30) kHz and for both 64 and 256 recording channels. Here, we also save data to the SD card of the PYNQ-Z2 every 100 ms, to ensure the system can record data for offline analysis while not affecting the data flow through the FIFO. Fig. 6.6d and 6.6e show the number of words over the course of each experiment (100 seconds). This fluctuates due to other processor tasks interrupting the Python loop which runs to retrieve the data packets and makes saves. Since the maximum FIFO capacity is 131071 words, the number of words must remain below this to avoid overflow and loss of data. As can be seen, this is the case over the majority of the duration of the experiment, although occasionally the buffer fills when the processor stalls at making the data request. For the six experiments, only two packets were lost, shown by the exclamation marks above the spike for the 64-channel, 30 kHz recording. Otherwise, the number of words in the FIFO remained at virtually zero except when the data was saved to the SD card each 100 ms (resulting in a delay in the Python code accessing the next buffer) shown by the zoomed insets. Overall, this represents a data loss of 2978 samples (100 ms) over the course of the 100 seconds for one of the 64-channel tests. The 256-channel tests did not demonstrate any data loss. The reason for this counter-intuitive result is due to random variations in the processor tasks causing delays in the Python processing loop. In general, this level of data loss would not be acceptable in a real neuroscience experiment, however it is expected that relatively easy-to-implement optimisations in the Python code (e.g. threading/multi-processing of the data acquisition loop) could make this more robust. Furthermore, if the FIFO starts filling up, the processor is able to rapidly empty the FIFO contents to avoid a propagating delay (shown by the zoomed insets). Except when the FIFO fills completely; in this case, the buffer resets and its entire contents are lost. Fig. 6.6f shows the amount of data (in milliseconds) stored in the FIFO which varies depending on sample rate and number of channels. The highest value for each line gives the FIFO capacity in milliseconds.

This experiment confirmed successful electrophysiology recording using the controller system with custom-modified FPGA firmware replicating the functionality of Intan RHD recording controllers. Real-time availability of the recorded samples through the Python/Jupyter interface enables an extended level of experimental control with relatively easy programmability. As described, the system can operate LED arrays (or any kind of LED device) so a real-time, closed-loop experiment can be designed in Python, all on the board and with no additional

software required (see next section). Due to the use of an embedded device it is easy to interact with and control other peripheral devices, enriching neuroscience experiments involving electrophysiology recordings.

6.2.3 Closed-loop electrophysiology recording and optogenetic stimulation

In this section, the control system's capabilities are fully explored through a closed-loop benchtop experiment, where events in the electrophysiology recording were used to drive a µLED device in real-time. As described in Chapter 2, there is a clear motivation for closed-loop technology, for example monitoring and suppressing (through optogenetic stimulation) patterns of neuronal activity associated with the onset of epileptic seizures [4]. Fig. 6.7a summarises the experimental setup; it is similar to before where a Pt wire delivers a charge into PBS where the recording electrodes are submerged. This time a step function is used to provide a "cue" that should be detected in real-time to drive an optical stimulation. Here we use the "hybrid" device described in Chapter 3 which integrates 32 recording electrodes over a µLED array. An Intan RHD2000 headstage is connected to the electrodes, and to the system through an SPI cable, while a 20-channel cable connects to the LED drivers on the board.

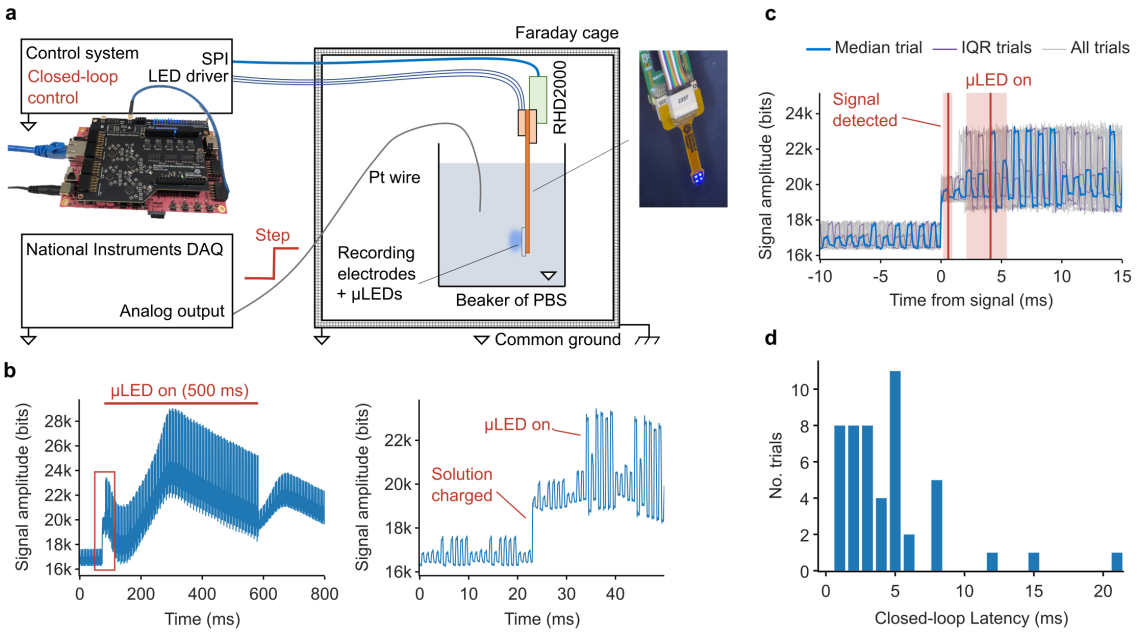


Figure 6.7: **Closed-loop electrophysiology and optical stimulation experiment. (a)** Schematic of experimental setup. **(b)** Recorded waveform showing when µLED array is switched on in response to recorded event. Zoomed inset shows when injected step function is detected and µLED array switches on (right). **(c)** All experimental trials showing average (median ± inter-quartile range, IQR) of signal detection and µLED array switching on in response. **(d)** Closed-loop latency of all 50 experimental trials represented as a histogram.

Firstly, a Python programme was developed which reads data packets (at 30 kHz, from 64-channels, 1 ms packets). If the signal passes above a pre-defined threshold, the µLED array is activated for 500 ms. Fig. 6.7b shows the recorded signal from one of the electrodes before and after the step function is delivered to the solution, and when the µLEDs are switched on. The noise in the signal is due to the switching electronics of the LED driver as it is operating in duty cycle mode, and is as high as ∼1.2 mV when the µLED is switched on. Clearly, this would be unacceptable for a practical electrophysiology experiment, and would need filtered out, the grounding improved, or improvements made to the electronics or device design. For the purposes of this experiment, the noise was ignored since the purpose of this experiment is to validate the closed-loop capabilities of the control system, and the signal of interest (the solution charging) can be easily discerned from the waveform (Fig. 6.7b right). This experiment was repeated over 50 trials, and the signal detection and µLED illumination time were calculated. The signal passing threshold is always detected in the recording between 0-1 ms after charge injection due to the packet size being set to 1 ms, although if the FIFO buffer fills, then it could be some time later before this is registered by the Python code. The µLED illumination time can be determined directly from the recorded waveform due to capacitive (electric field) coupling or the Becquerel effect (optical coupling through conductive solution). This gives a definitive report of the closed-loop latency, shown as median and inter-quartile range (IQR) in Fig. 6.7c, and as a histogram in Fig. 6.7d. The latency of approx. 5 ms is comparable to similar technologies for closed-loop optogenetic stimulation with recorded electrophysiology [8]. Large outliers in this measurement are likely due to the FIFO buffer filling up, due to a stall in the Python loop which also saves the recorded waveform every 100 ms. In this case, after the PS has recovered, the buffer is rapidly emptied and the signal change is detected and the µLEDs driven. Overall, this experiment could be optimised and removing the noise would give better visualisation. There remains room for further work in optimising the code to reduce the latency, and provide more advanced closed-loop control, beyond a simple threshold detection. However, the FPGA design is robust, so this would only involve Python programming. Overall, this experiment successfully demonstrated the closed-loop capabilities of the system, highlighting its potential use in sophisticated neuroscience studies.

6.3 Discussion

This chapter has detailed the design of an electronic control system to facilitate advanced *in-vivo* experiments using the µLED array technology developed here, and, more broadly, other µLED-based devices for optogenetics. These systems were a crucial component in the success of the *in-vivo* experiments described in Chapter 5, demonstrating their portability and robustness. Then, the functionality was expanded upon, providing a platform to introduce closed-loop

control through audio (or other peripheral) inputs, and using real-time electrophysiology recordings to drive optogenetic stimulation. In the latter case, the Intan Rhythm3 framework was adapted for the PYNQ-Z2 board, providing a highly-capable system and low barrier-for-entry for non-programmers. The closed-loop experiment carried out highlighted how a sophisticated experimental paradigm could be rapidly developed using this technology. As discussed in Chapter 2, there has been a recent move to incorporate closed-loop control in neural interfacing systems, positioning this system in potential future neuroscience studies.

Bibliography

1. Greer, R. *et al.* Chronically implantable LED arrays for optogenetic cortical surface stimulation in mice. en. *Nature Communications* **17,** 878. ISSN: 2041-1723. doi:10.1038/s41467-025-68191-5. (2026) (Jan. 2026).
2. Clark, A. M. *et al.* An optrode array for spatiotemporally-precise large-scale optogenetic stimulation of deep cortical layers in non-human primates. en. *Communications Biology* **7,** 329. ISSN: 2399-3642. doi:10.1038/s42003-024-05984-2. (2026) (Mar. 2024).
3. *Intan Technologies* en-US. Accessed on 25/01/2026. https://www.intantech.com.
4. Luo, J. *et al.* The Neural Engine: A Reprogrammable Low Power Platform for Closed-Loop Optogenetics. eng. *IEEE transactions on bio-medical engineering* **67,** 3004–3015. ISSN: 1558-2531. doi:10.1109/TBME.2020.2973934 (Nov. 2020).
5. *PYNQ* en-US. Accessed on 25/01/2026. http://www.pynq.io/.
6. *Learn the architecture - An introduction to AMBA AXI* Accessed on 25/01/2026. https://developer.arm.com/documentation/102202/0300/AXI-protocol-overview.
7. Ceballo, S., Piwkowska, Z., Bourg, J., Daret, A. & Bathellier, B. Targeted Cortical Manipulation of Auditory Perception. eng. *Neuron* **104,** 1168–1179.e5. ISSN: 1097-4199. doi:10.1016/j.neuron.2019.09.043 (Dec. 2019).
8. Abbasi, A. *et al.* Brain-machine interface learning is facilitated by specific patterning of distributed cortical feedback. *Science Advances* **9,** eadh1328. doi:10.1126/sciadv.adh1328. (2026) (Sept. 2023).

Chapter 7

Conclusions and future work

7.1 Discussion

This thesis has detailed the work undertaken to develop and characterise chronically implantable optogenetic cortical surface implants based on µLED arrays fabricated within the group. The work undertaken in this thesis alongside our collaborators, has been published, highlighting the potential impact of novel devices, such as the one detailed here, for use in future neuroscience studies [1].

The key result chapters outlined the development of a micro-assembly-based integration approach to achieve robust, chronically implantable systems for use in behavioural experiments in the mouse model (Fig. 3.5). Accelerated testing with the devices in a simulated biological environment verified the devices could survive over the duration months-long active *in-vivo* studies (Fig. 3.11). The packaging procedure facilitates rapid customisability utilising commercial polyimide, flexible printed circuit boards (PCBs) (Fig. 3.9). New designs could therefore allow chronic optogenetic experiments across multiple cortical regions in a single implantation, or in animals with larger cortices like non-human primates (NHPs) [2].

Electrical characterisation of the packaged devices in the laboratory allowed optical and thermal models to be developed, providing an operational framework to inform *in-vivo* experiments (Fig. 4.1, 4.5, 4.16). Then, a series of *in-vivo* electrophysiology and behavioural experiments were carried out, in collaboration with the Bathellier laboratory. This yielded data that were used to establish a stimulation threshold of approximately 5 mW/mm^2 (5 mA drive current) to achieve robust neuronal activations (Fig. 5.7), leading to consistent behavioural outcomes. Throughout *in-vivo* experiments, control mice which did not express the opsin were used to verify observed responses were the result of optogenetic stimulation, and not the result of thermally-induced neuronal firing or other recording artefacts.

The goal of the µLED array devices was to provide high-density and high-spatial resolution optical stimulation coverage across a cortical region. Ideally stimulation site spacing would be

of the order of 100 µm [3]. *In-vivo* electrophysiology experiments demonstrated a lateral width of activation of approximately 800 µm; this is with a cortical window implanted (Fig. 5.12). The obvious choice to improve spatial resolution was to remove the cortical window and implant the device in direct contact with cortical tissue, as is done for our chronic implantations. This reduces the width of activation by $\sim$ 200 µm to $\sim$ 600 µm (Fig. 5.12). To improve this further, using the experimentally validated optical models, an investigation was carried out into light shaping, specifically the use of an optical interposer; a device which can collimate light using small holes etched into an otherwise opaque material like silicon. These models showed the illuminated width could be reduced to $\sim$ 500 µm but requiring a significantly higher drive current (4-5 times) to deliver an equivalent optical power (Fig. 4.9). Given the high efficiency of the devices presented here, this is achievable, and developing the interposer would require a relatively straightforward fabrication and integration process. Comparing the interposer models to models of perfectly collimated light sources shows that achieving equivalent performance to a digital light projector (DLP) system is possible (Fig. 4.10). Light scattering in the brain tissue is then the dominant factor limiting spatial resolution. The only way to improve this further would be to reduce the 150 µm standoff distance between source and tissue, due to the sapphire substrate which could be achieved through laser lift-off or selective chemical etching processes [4]. These techniques allow devices to be transfer printed onto other substrates to achieve optogenetic cortical surface stimulation with similar stimulation site spacing to ours, but significantly reduced source-tissue promximity [5, 6]. These devices, however, have not been characterised to the same extent as here, and further *in-vivo* experiments are needed to confirm they can deliver sufficient power to achieve robust neuronal responses leading to behavioural outcomes, while remaining stable over the duration of long-term implantations.

A series of behavioural experiments were carried out, both in head-fixed and freely-behaving mice, using the chronically implantable µLED arrays with flexible packaging. Over the course of ten experiments, all non-control mice achieved robust discrimination performance in a task which optogenetically stimulated large, distinct sub-regions of the auditory cortex (Fig. 5.16) (Table 5.2). Reducing the size of the illuminated region and moving the stimulation points closer together demonstrates a spatial resolution, limiting the perception of stimulation with the devices (Fig. 5.28). A greater number of mice are needed to draw conclusive results, and ideally at a range of cortical irradiances. Additionally, this result further corroborates the optical modelling (Fig. 5.29).

Open questions remain on the exact mechanism of the neuronal and behavioural responses recorded as a result of optogenetic stimulation with the µLED array. That is, both optogenetic constructs employed in the experiments make the neuron sensitive to stimulation at the cell body and dendrites, and we cannot say whether responses were from somatic activation,

summed dendritic activation, or other second-order network effects. To elucidate this further, a multi-unit analysis (MUA) was carried out using the electrophysiology data. This looked at bulk neuronal responses as a function of recording depth. Responses were seen at the deepest electrodes (>600 μm deep) (Fig. 5.15); since the light did not penetrate this deep (Fig. 5.11), this strongly suggests the contribution of a second-order effect like the activation of layer 5 pyramidal neurons through stimulation of their dendrites in layer 2/3. An interesting future study could utilise soma-expressing opsin variants such as ST-ChroME [7] to decouple these results and gain a clearer understanding of the stimulation regime. Throughout all *in-vivo* experiments, the same temporal stimulation pattern has been utilised (10 pulses, 20 Hz, 50% duty cycle). This could also be adapted in future studies, for example to use faster electrical stimulation, known to recruit inhibitory networks [8], in tandem with optogenetic stimulation to gain a clearer understanding of auditory cortical circuit dynamics.

Finally, an electronic control system was designed, capable of driving up to a 20-channel matrix LED device, with simultaneous electrophysiological recording from Intan RHD2000 ASIC headstages [9]. The use of a system-on-chip incorporating a field-programmable gate array (FPGA) opens the possibility of real-time, embedded and closed-loop optogenetic experiments.

7.2 Future work

Here, I will outline several avenues for potential future work that could be carried out based on the results of this thesis, in order of importance.

7.2.1 Braincodec

Capitalising on the behavioural experiments carried out in Chapter 5, advancing the discrimination task to involve meaningful information transfer to cortical circuits is an exciting direction to realise the impact of the work carried out here. An encoding algorithm has been developed within the Bathellier laboratory called Braincodec. Providing a detailed description of this algorithm is outwith the scope of this thesis, however Braincodec uses a deep neural network architecture based on Meta's Encodec algorithm [10] trained with biological constraints, to encode an audio signal into a two-dimensional pattern to be projected onto the surface of the auditory cortex. This could serve as a sophisticated coding strategy for an eventual auditory cortical implant, which could benefit deaf patients unable to receive a cochlear implant due to a missing or damaged acoustic nerve. Furthermore, one of the key limitations of cochlear implants, low perceived spectral resolution, particularly in noisy environments, could be overcame by the expanded number of stimulation sites possible over the larger cortical structure.

The Python-based electronic control system has a straightforward interface for controlling

the µLED array implants described here. Therefore, trained Braincodec model outputs can be readily applied to behavioural discrimination experiments. So far, *go* and *no-go* patterns – representing pure tone frequencies at opposite ends of the spectrum – along with intermediate patterns in between these have been tested. These have produced highly promising results, demonstrating that animals perceive these optogenetic stimulation patterns similarly to normal hearing (Fig. 7.1).

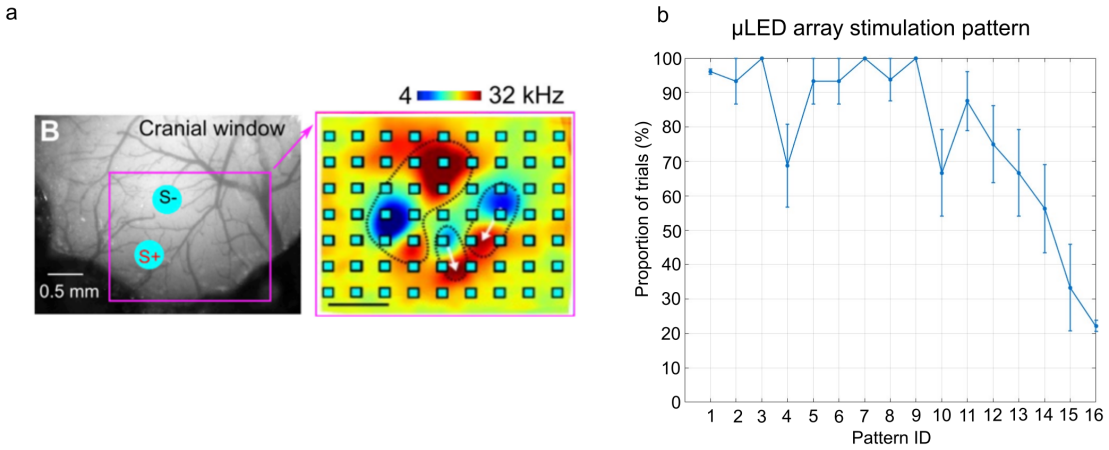


Figure 7.1: **Braincodec**. **(a)** Tonotopy of auditory cortex with stimulation matrix overlaid [3]. **(b)** Preliminary results comparing discrimination task performance between stimulation patterns projected on the auditory cortex. The µLED stimulation pattern is the output from the Braincodec model given an input auditory signal at frequencies ranging from 1 kHz to 16 kHz. The pattern ID '1' is treated as the *go* stimulus with all other patterns treated as the *no-go*. The y axis shows the proportion of trials that the mouse licked the sensor for (i.e. when expecting a *go* stimulus).

These results are preliminary due to the small number of animals (n = 2) however are extremely promising, opening the possibility of testing different encoding schemes.

The electronic control system detailed in Chapter 6 is capable of real-time neural network inference (Fig. 7.2). Two approaches could be considered for a practical embedded implementation. The first and simplest approach is using the dual-core ARM A9 processing system and importing the trained model in Python. This is best done with Tensorflow-lite (a deep neural network design tool optimised for embedded implementations) [11] as opposed to manually implementing the network, which would be too inefficient and lead to high latency. Initial tests suggest the current model with a 25 ms audio signal can be calculated in <13 ms, well within the requirement of 25 ms. Further optimisations may need to consider the second approach; utilising the FPGA on the Zynq as a coprocessor to offload the computationally-intensive convolution operations. This would involve more work, but would result in a lower latency implementation. Considering the control system detailed here includes a 3.5 mm audio jack and built-in audio codec IC on-board, along with simple Python APIs for accessing the sampled

audio data, makes real-time experiments with Braincodec an exciting prospect, particularly on deaf animals.

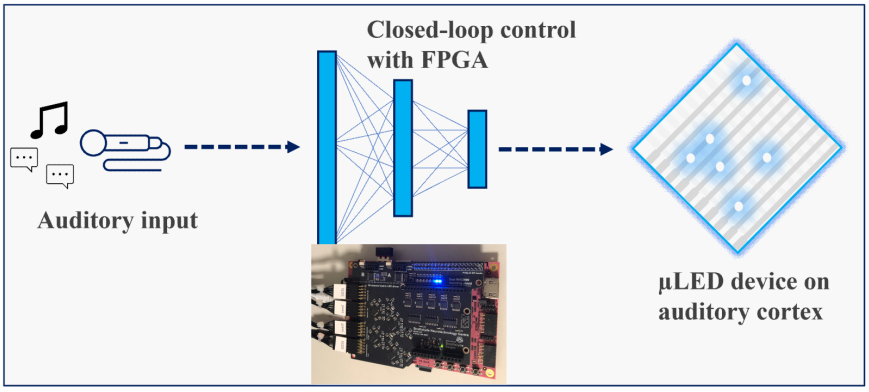


Figure 7.2: **Real-time control of µLED array with auditory input in real-time.**

7.2.2 Novel *in-vivo* experiments

The outcome of this project represents a technology platform for neuroscience studies involving spatiotemporal optogenetic stimulation of the cortical surface in mice. Chronic stability enables stimulation of the same ensembles of neurons over weeks or months-long periods. Furthermore, stimulation parameters that have been robustly established through laboratory charaterisation, *in-vivo* electrophysiology and behavioural experiments, as well as an advanced but straightforward-to-use electronic control system and software, make this platform readily translatable to new *in-vivo* studies.

A priority should therefore be to capitalise on the development here and translate the devices to high-impact neuroscience studies. Batches of devices should be assembled, characterised and provided, alongside the associated electronic control system and cables, to neuroscience laboratories. This has been realised with funding from the EU BOOST programme [12].

7.2.3 Closed-loop electrophysiology

As described in Chapter 6, the control system was designed with an interface to Intan RHD2000 electrophysiology recording headstages. This system, although developed and characterised, has not been translated to a neuroscience experiment. Currently, the system is able to record electrophysiology data and control the LED driver within a single FPGA configuration, and entirely through the Python/Jupyter Notebook interface. To realise a practical experiment and potential publication, the system requires both neuroscience input (experimental design), and engineering support. The design could also be improved by using a better development board,

for example the ZCU104 with a 64-bit quad-core ARM processor, and larger FPGA with greater block RAM capacity [13]. This could better deal with the high data rate coming from the AXI-streaming buffer and still run the PYNQ framework. Although the PYNQ-Z2 still works well for a lower sample rate and simple closed-loop experimental paradigm. Alternatively, and as described in Section 6.2.2, the Python implementation remains open to further optimisations to improve the robustness of the data acquisition loop.

7.2.4 Novel system geometries

Due to the packaging process developed for the µLED arrays, alternate designs potentially scaling to multiple µLED arrays are readily possible. This is due to the use of commercial PCB designs (PCBway) which are low-cost and easy to design. Chapter 3 already demonstrated a design with 4 µLED arrays, matrix addressed as a 20x20 layout. Custom designs could allow chronic experiments with high-density optogenetic stimulation in multiple cortical regions. Furthermore, as the packaging is flexible, devices could be scaled to non-human primate models which have significantly larger cortical regions than the mouse model, but which are still approximately flat at the scale of the µLED arrays (2 mm). Custom packaging designs could represent an opportunity to readily push these devices to other neuroscience laboratories with the potential for commercialisation.

7.2.5 Multi-colour µLED arrays

Incorporation of red µLEDs onto the 10x10 matrix represents a possibility to fully explore the cell-type specificity offered by optogenetics, although modifying the design to facilitate the transfer printing of red membranes would be challenging, with limited scalability compared with the blue-only devices shown here. These devices could allow the selective activation of excitatory or inhibitory cortical circuits, providing further insight into how best to stimulate the cortex, auditory or otherwise, in future cortical prostheses.

7.3 Final remark

The work carried out in this thesis represents the groundwork for novel future neuroscience studies into cortical and sensory function, with the potential to inform the clinical translation of such devices.

Bibliography

1. Greer, R. *et al.* Chronically implantable LED arrays for optogenetic cortical surface stimulation in mice. en. *Nature Communications* **17,** 878. ISSN: 2041-1723. doi:10.1038/s41467-025-68191-5. (2026) (Jan. 2026).
2. Clark, A. M. *et al.* An optrode array for spatiotemporally-precise large-scale optogenetic stimulation of deep cortical layers in non-human primates. en. *Communications Biology* **7,** 329. ISSN: 2399-3642. doi:10.1038/s42003-024-05984-2. (2026) (Mar. 2024).
3. Ceballo, S., Piwkowska, Z., Bourg, J., Daret, A. & Bathellier, B. Targeted Cortical Manipulation of Auditory Perception. eng. *Neuron* **104,** 1168–1179.e5. ISSN: 1097-4199. doi:10.1016/j.neuron.2019.09.043 (Dec. 2019).
4. Klein, E., Gossler, C., Paul, O., Schwarz, U. T. & Ruther, P. High-yield indium-based wafer bonding for large-area multi-pixel optoelectronic probes for neuroscientific research. en. *Journal of Micromechanics and Microengineering* **29,** 095006. ISSN: 0960-1317. doi:10.1088/1361-6439/ab2a53. (2026) (July 2019).
5. Sekiguchi, H. *et al.* Adhesionable flexible GaN-based microLED array film to brain surface for in vivo optogenetic stimulation. en. *Applied Physics Express* **15,** 046501. ISSN: 1882-0786. doi:10.35848/1882-0786/ac5ba3. (2026) (Mar. 2022).
6. Gu, W., Wang, L., Wang, X., Zhao, C. & Guan, S. Large-Scale, High-Density MicroLED Array-Based Optogenetic Device for Neural Stimulation and Recording. *Nano Letters* **24,** 15252–15259. ISSN: 1530-6984. doi:10.1021/acs.nanolett.4c03645. (2026) (Dec. 2024).
7. Mardinly, A. R. *et al.* Precise multimodal optical control of neural ensemble activity. en. *Nature Neuroscience* **21,** 881–893. ISSN: 1546-1726. doi:10.1038/s41593-018-0139-8. (2026) (June 2018).
8. Xie, Y. *et al.* High-frequency electrical stimulation suppresses cholinergic accumbens interneurons in acute rat brain slices through GABAB receptors. en. *European Journal of Neuroscience* **40,** 3653–3662. ISSN: 1460-9568. doi:10.1111/ejn.12736. (2026) (2014).

9. *Intan Technologies* en-US. Accessed on 25/01/2026. https://www.intantech.com.

10. Défossez, A., Copet, J., Synnaeve, G. & Adi, Y. *High Fidelity Neural Audio Compression* arXiv:2210.13438 [eess]. Oct. 2022. doi:10.48550/arXiv.2210.13438. (2025).

11. David, R. *et al. TensorFlow Lite Micro: Embedded Machine Learning on TinyML Systems* arXiv:2010.08678 [cs]. Mar. 2021. doi:10.48550/arXiv.2010.08678. (2026).

12. *Boost* en-US. Accessed on 08/02/2026. https://boost-tech.eu/.

13. *AMD Zynq™ UltraScale+™ MPSoC ZCU104 Evaluation Kit* en. Accessed on 08/02/2026. https://www.amd.com/en/products/adaptive-socs-and-fpgas/evaluation-boards/zcu104.html.

Appendix A

Supplementary information

A.1 µLED array wafer history and failure mechanisms

Although the µLED array fabrication and process debugging is outwith the scope of this thesis, a small summary of the wafer history is provided here for reference and context.

µLED arrays are fabricated at wafer-scale (220 die per 2" wafer). In this report, they are labelled as H[wafer number]-[die number].

- **H1**: Initial fabrication run. Issues with the passivation layer causing shorts. Devices not used.
- **H3**: Second fabrication run. Devices had very high brightness (representative LI curves based on H3 µLEDs) however suffered from passivation issues causing shorts. Single device (H3-019) used for electrophysiology experiments only.
- **H2**: Third fabrication run. Devices had lower brightness than H3 but passivation issue resolved, therefore very few devices with shorts. These devices were used to generate all the behavioural results presented in this thesis (10 mice implanted and responded successfully).
- **H4**: Fourth fabrication run. Devices have similar (slightly lower) brightness than H2 but slightly better I-V characteristic. Less stable than H2; shorts develop during packaging process and *in-vivo* use at a faster rate than H2.
- **H5**: Fifth fabrication run. Significantly greater efficiency than H2 and H4. Testing in progress and stability to be verified as of the writing of this report.

A.2 Comparison of previous chronically implantable μLED/LED array cortical surface stimulators

Reference	Pixel size	No. pixels	Pixel pitch	Stimulation area	Power density dynamic range on brain surface (mW mm^{-2})	Animal model
Pollmann et al. 2024	0.2 x 0.1 mm^2	48	0.6 – 1.25 mm	6.4 x 7.8 mm^2	70	NHP
Rajalingham et al. 2021	0.5 x 0.5 mm^2	24	1 mm	5 x 5 mm^2	56	NHP
Shin et al. 2024	1.6 x 1.6 mm^2	4	>1 mm	>4 mm^2	-	Mouse (transcranial)
Komatsu et al. 2017	0.33 x 0.33 mm^2	8	4 mm	8 x 16 mm^2	-	NHP
Ohta et al. 2021	0.35 x 0.28 mm^2	48	1 mm	9 x 6.5 mm^2	>10	NHP
Lee et al. 2018	0.05 x 0.05 mm^2	12	0.5 mm	2 x 1.5 mm^2	>10	Rat
Wu et al. 2025	0.3 x 0.3 mm^2	Up to 64	1.5 mm	4.5 x 4 mm^2	70	Mouse
This work	0.04 x 0.04 mm^2	100	0.2 mm	2 x 2 mm^2	>50	Mouse

A.3 p values from statistical tests

µLED	Experiment 1	Experiment 2	Experiment 3	Experiment 4	Experiment 1-2
P2N3	0.0010	0.0075	0.00001	0.1523	0.1262
P3N3	0.0002	0.0154	0.0001	0.0295	0.3638
P4N3	0.0001	0.0062	0.00001	0.0744	0.3786
P5N3	0.0003	0.1491	0.0001	0.1730	0.3812
P6N3	0.0993	0.4576	0.2169	0.4049	0.6426
P7N3	0.0927	0.3274	0.1998	0.2821	0.5188
P8N3	0.2462	0.3968	0.0575	0.4149	0.5037
P9N3	0.3854	0.4491	0.7084	0.6005	0.7247
P10N3	0.1009	0.5214	0.1564	0.5102	0.5311

p values for electrophysiology statistical tests. Independent two-sample, single-tailed Student's t-test with alt. hypothesis: spike counts (ChR2+) > spike counts (ChR2-); ChR2+ (optogenetic): Exp. 1 n = 18, Exp. 2 n = 45, Exp. 3 n = 16, Exp. 4 n = 53, Exp. 1-2 n = 23; ChR2- (control): n = 51 neurons.

Association				Discrimination		
go + / blank +	go + / blank -	blank + / blank -	go - / blank -	go / no-go	go / blank	no-go / blank
0.00004	0.00027	0.08264	0.08264	0.00005	0.000003	0.05651

p values for behavioural statistical tests. Independent two-sample, single-tailed Student's t-test with alt. hypothesis: go accuracy > no-go or go accuracy > blank accuracy and/or ChR2+ accuracy > ChR2- accuracy. ChR2+: n = 4 mice, ChR2-: n = 2 mice.

A.4 Behavioural experimental parameters

	Session	Session time (hr:min)	No. trials	Accum. trials	*Go* pattern	µLED irradiance (mW mm^{-2})	MFB voltage (V)	MFB duration (ms)
Mouse 1	1	1:26	300	300	P8N8 P9N9 P10N10	10	3.0	0.01
	2	1:18	205	505	P8N8 P9N9 P10N10	10	3.0	0.01
	3	1:23	233	738	P8N8 P9N9 P10N10	10	3.0	0.01
	4	1:39	400	1138	P1N1 P2N2 P3N3	10	2.4	0.01
	5	1:19	400	1538	P1N1 P2N2 P3N3	10	2.4	0.01
	6	0:44	400	1938	P1N4 P2N5 P3N6	8	2.4	0.01
Mouse 2	1	1:29	187	287	P1N8 P2N9 P3N10	10	1.8	0.01
	2	1:43	300	487	P1N8 P2N9 P3N10	10	1.8	0.01
	3	1:02	300	787	P1N8 P2N9 P3N10	10	1.7	0.01
	4	1:20	400	1187	P1N8 P2N9 P3N10	10	1.7	0.01
	5	1:12	400	1587	P1N8 P2N9 P3N10	10	1.7	0.01
Mouse 3	1	1:15	239	239	P1N1 P2N2 P3N3	10	2.0	0.001
	2	0:56	350	589	P1N1 P2N2 P3N3	10	2.0	0.001
	3	1:59	300	889	P1N1 P2N2 P3N3	10	2.0	0.001
	4	1:26	400	1289	P1N1 P2N2 P3N3	10	2.0	0.001
	5	0:34	200	1489	P3N2 P4N3 P5N4	6	2.0	0.001
Mouse 4	1	1:11	400	400	P1N1 P2N2 P3N3	10	2.6	0.01
	2	0:56	350	750	P1N1 P2N2 P3N3	10	2.6	0.01
	3	1:06	255	1005	P1N1 P2N2 P3N3	10	3.0	0.01
	4	1:16	400	1405	P2N3 P3N4 P4N5	8	2.6	0.01
Mouse 5	1	1:07	400	400	P8N1 P9N2 P10N3	8	2.2	0.001
	2	1:07	400	800	P8N1 P9N2 P10N3	8	2.2	0.001
	3	0:39	256	1056	P8N1 P9N2 P10N3	8	2.2	0.001
	4	1:53	600	1656	P8N1 P9N2 P10N3	8	2.6	0.001
	5	2:55	565	2221	P8N1 P9N2 P10N3	8	2.2	0.001
	6	1:30	210	2431	P8N1 P9N2 P10N3	8	2.0	0.001
	7	1:30	186	2617	P8N1 P9N2 P10N3	8	1.9	0.001
	8	1:11	521	3138	P8N1 P9N2 P10N3	8	1.7	0.001
Mouse 6	1	1:09	400	400	P1N8 P2N9 P3N10	6	1.8	0.01
	2	1:10	400	800	P1N8 P2N9 P3N10	6	1.8	0.01
	3	0:56	174	974	P1N8 P2N9 P3N10	6	1.8	0.01
	4	1:46	130	1104	P1N8 P2N9 P3N10	6	1.8	0.01
	5	4:18	225	1329	P1N8 P2N9 P3N10	6	1.9	0.01
	6	2:55	200	1529	P1N8 P2N9 P3N10	6	1.6	0.01
	7	2:26	228	1757	P1N8 P2N9 P3N10	6	1.2	0.01
	8	3:47	600	2357	P1N8 P2N9 P3N10	6	1.5	0.01

Summary of behavioural association experiment parameters.

Appendix A. Supplementary information

	Session	Session time (hr:min)	No. trials	Accum. trials	*Go* pattern	*No-go* pattern	µLED irradiance (mW mm^{-2})	MFB voltage (V)	MFB duration (ms)
Mouse 1	1	1:26	600	600	P1N4 P2N5 P3N6	P8N8 P9N9 P10N10	8	2.4	0.01
	2	1:17	400	1000	P1N4 P2N5 P3N6	P8N8 P9N9 P10N10	8	2.4	0.01
	3	1:17	400	1400	P1N4 P2N5 P3N6	P8N8 P9N9 P10N10	8	2.3	0.01
	4	1:19	400	1800	P1N4 P2N5 P3N6	P8N8 P9N9 P10N10	8	2.3	0.01
	5	1:07	400	2200	P1N4 P2N5 P3N6	P8N8 P9N9 P10N10	8	2.2	0.01
	6	1:04	400	2600	P1N4 P2N5 P3N6	P8N8 P9N9 P10N10	8	2.2	0.01
Mouse 2	1	0:53	280	280	P1N8 P2N9 P3N10	P5N1 P6N2 P7N3	10	1.7	0.01
	2	0:50	280	560	P1N8 P2N9 P3N10	P5N1 P6N2 P7N3	10	1.7	0.01
	3	0:49	280	840	P1N8 P2N9 P3N10	P5N1 P6N2 P7N3	8	1.7	0.01
	4	0:43	400	1240	P1N8 P2N9 P3N10	P5N1 P6N2 P7N3	8	1.7	0.01
	5	0:43	400	1640	P1N8 P2N9 P3N10	P5N1 P6N2 P7N3	8	1.7	0.01
Mouse 3	1	1:11	400	400	P3N2 P4N3 P5N4	P8N8 P9N9 P10N10	6	2.0	0.001
	2	1:06	400	800	P3N2 P4N3 P5N4	P8N8 P9N9 P10N10	5	1.8	0.001
	3	1:09	400	1200	P3N2 P4N3 P5N4	P8N8 P9N9 P10N10	5	1.8	0.001
Mouse 4	1	1:11	400	400	P2N3 P3N4 P4N5	P8N8 P9N9 P10N10	6	2.6	0.01
	2	1:10	400	800	P2N3 P3N4 P4N5	P8N8 P9N9 P10N10	5	2.6	0.01
	3	1:05	400	1200	P2N3 P3N4 P4N5	P8N8 P9N9 P10N10	5	2.6	0.01

Summary of behavioural discrimination experiment parameters.

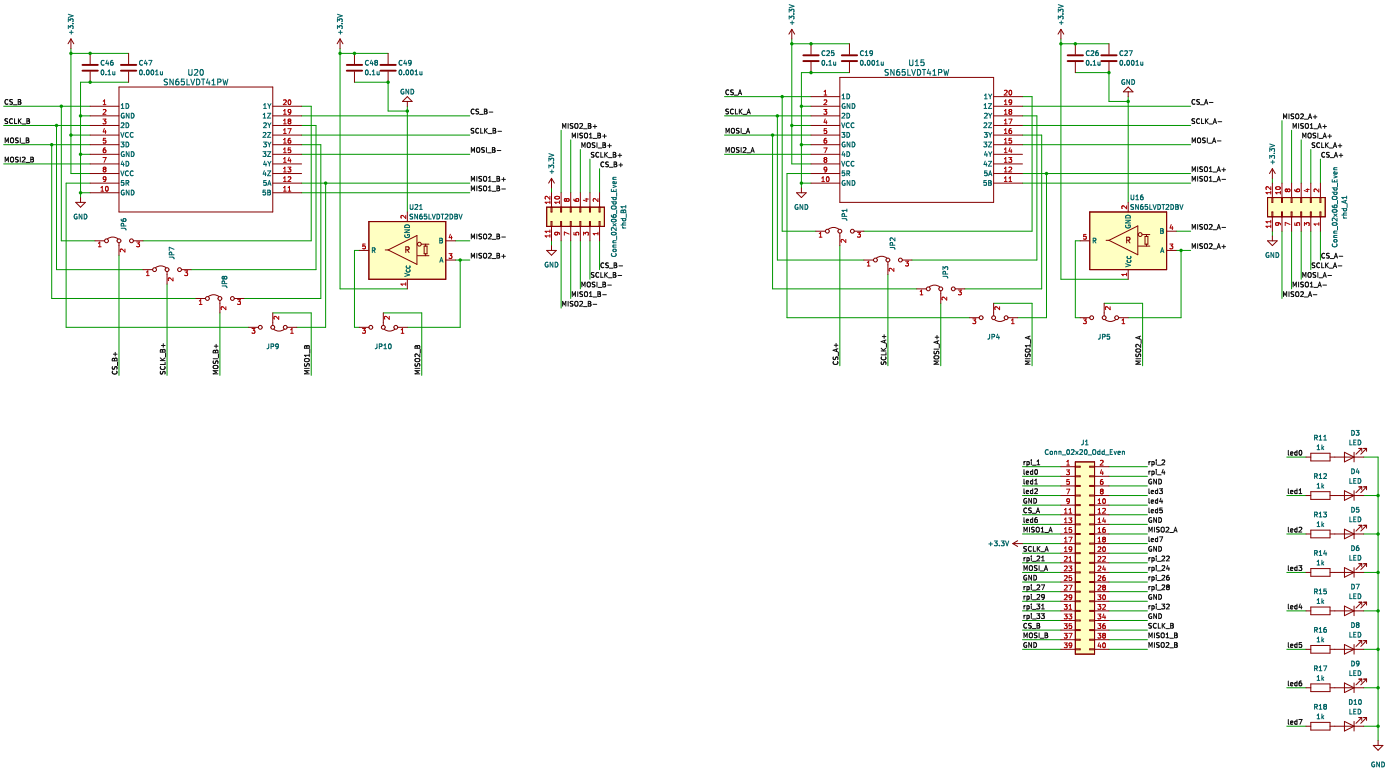
Appendix B

Hardware designs

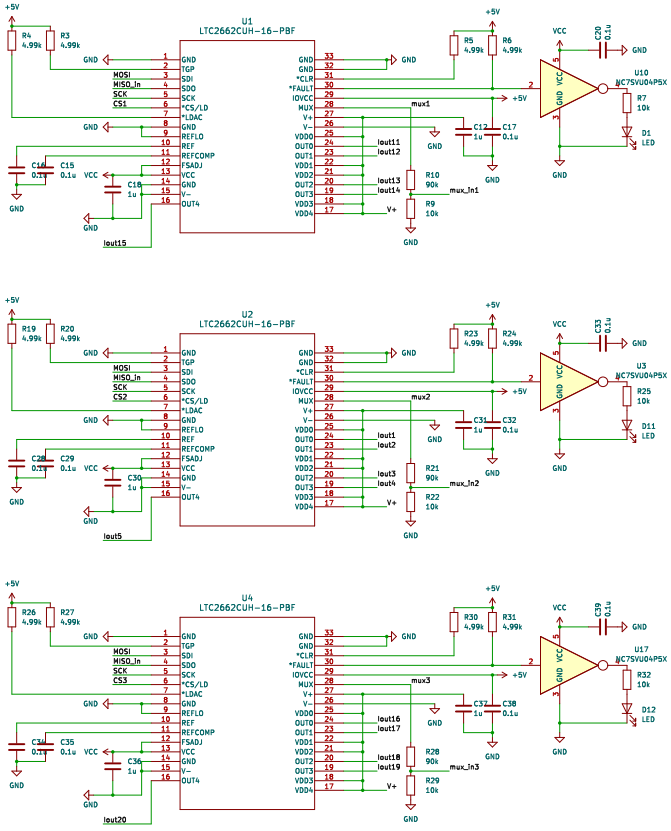
B.1 PCB schematics

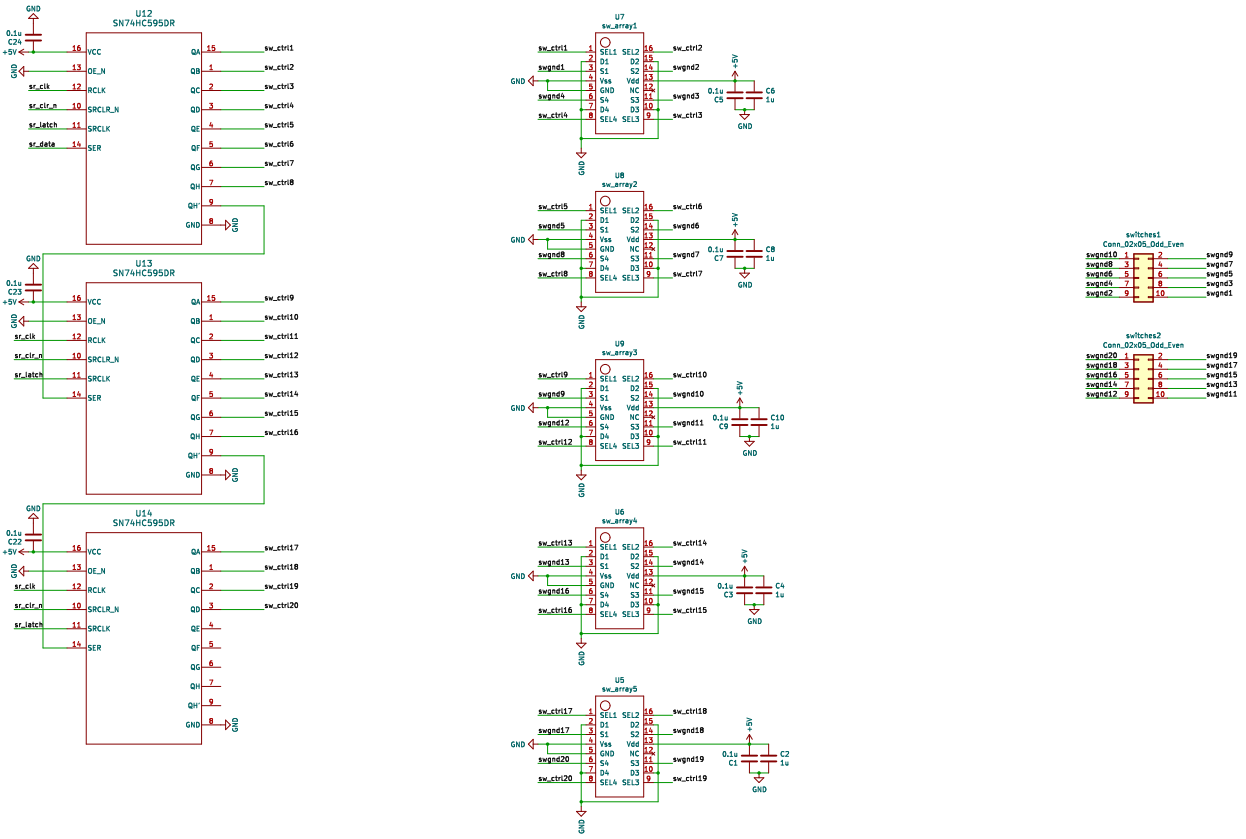
Note: PCB schematics and board designs are provided in vector format. For full resolution, please view on a PC.

B.1.1 RHD2000 interface



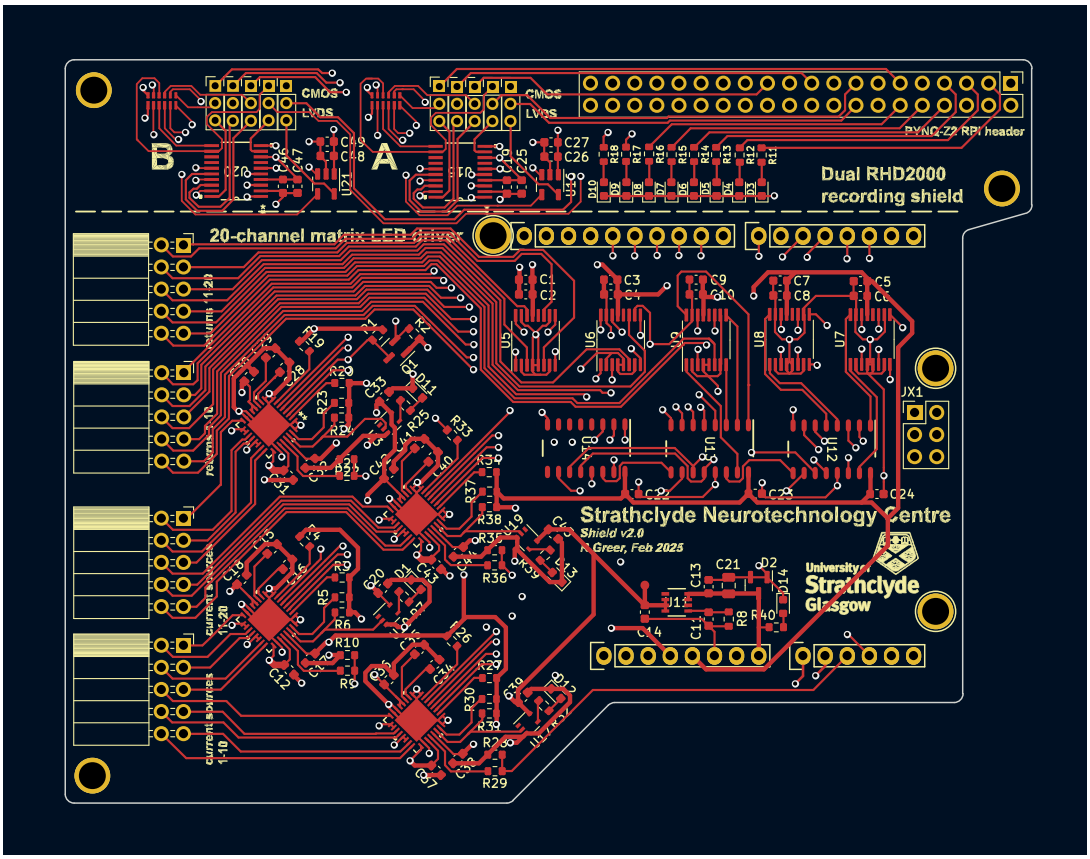
B.1.2 LED driver



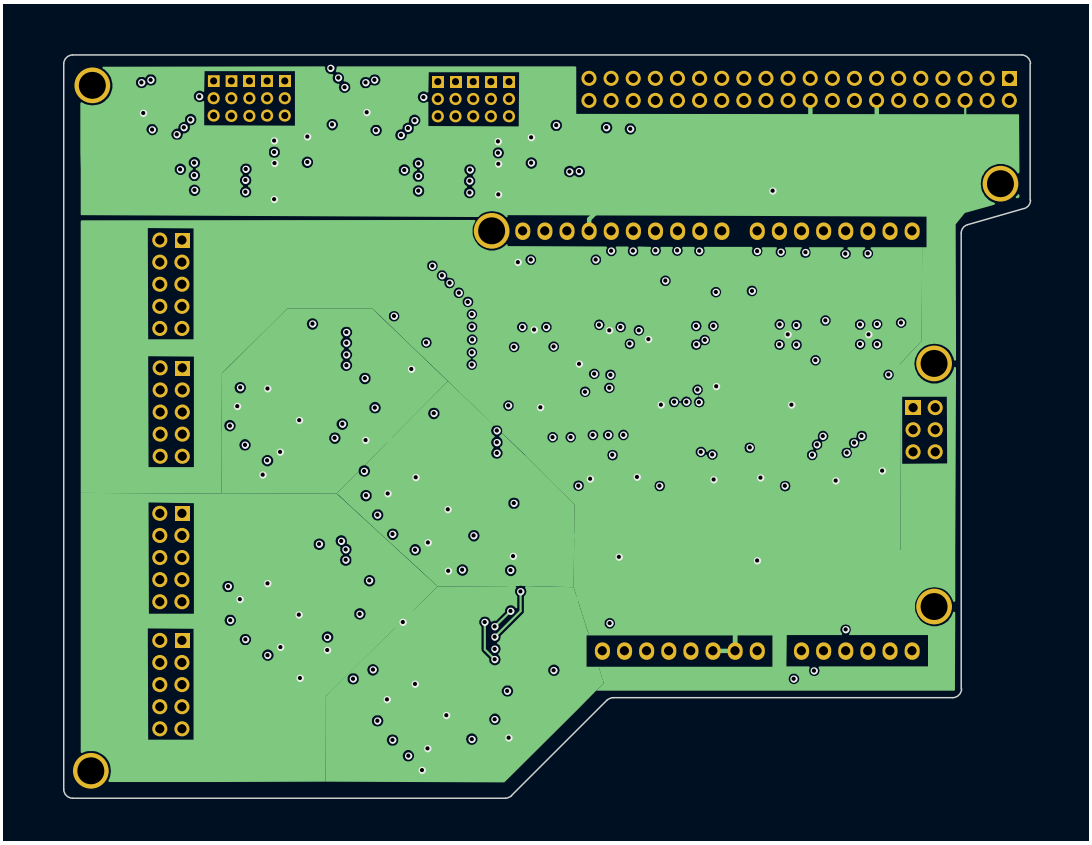

U12
SN74HC595DR
U13
SN74HC595DR
U14
SN74HC595DR
U7
sw_array1
U8
sw_array2
U9
sw_array3
U6
sw_array4
U5
sw_array5
switches1
Conn_02x05_Odd_Even
switches2
Conn_02x05_Odd_Even
GND
+5V
0.1u
1u
sr_clk
sr_clr_n
sr_latch
sr_data

B.2 PCB board design

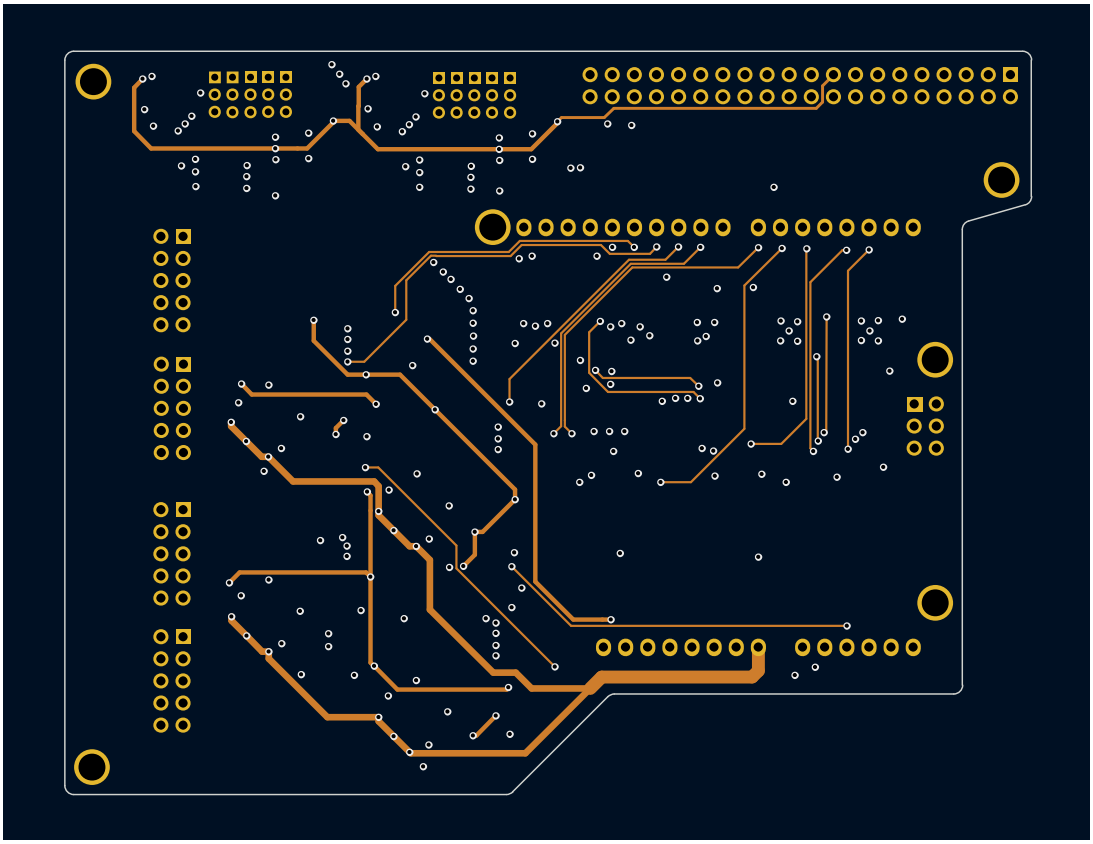
B.2.1 Front layer



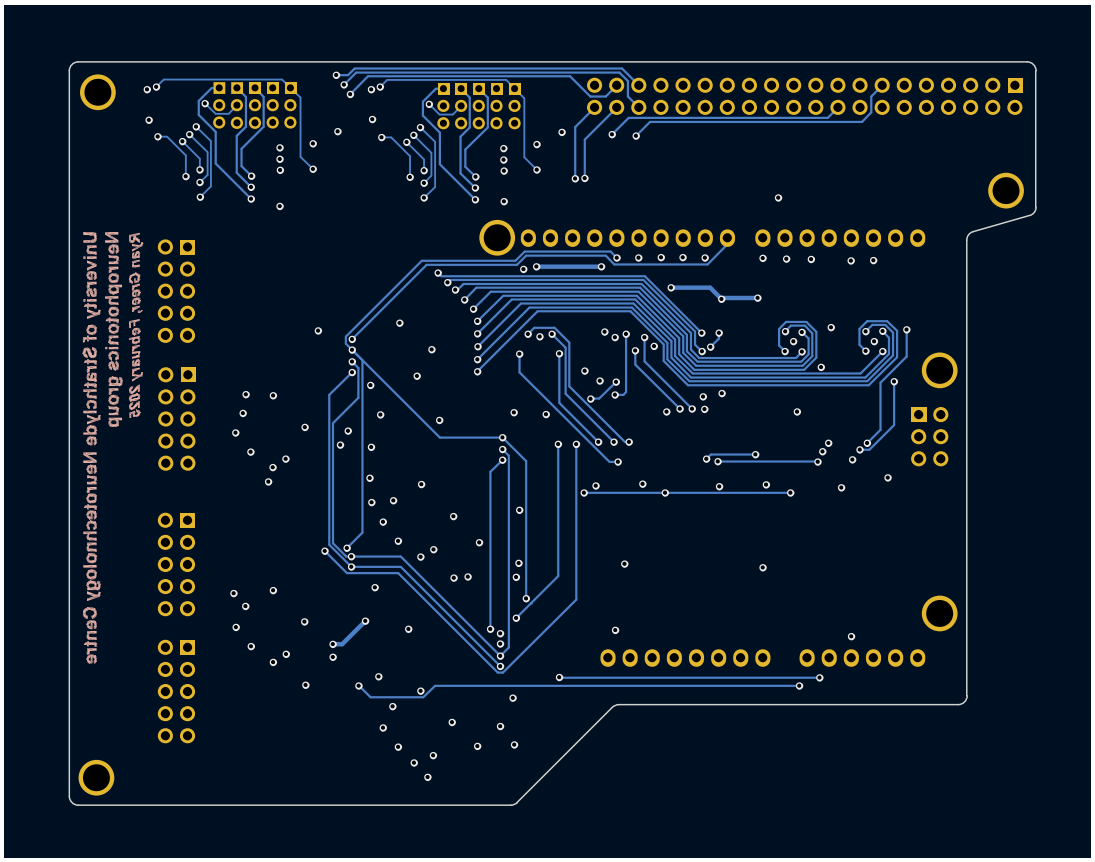
B.2.2 Inner layer 1



B.2.3 Inner layer 2



B.2.4 Bottom layer



Appendix C

Code listings

C.1 FPGA Verilog code listings

Rhythm_v3_3_AXI

main.v

Adapted from https://intantech.com/downloads.html?tabSelect=Source&yPos=0; only modified parts included.

```
1   //////////////////////////////////////////////////////////////////////////////////
2   // Company:           Intan Technologies, LLC
3   //                 Copyright (c) 2023 Intan Technologies LLC
4   //
5   // Design Name:    RHD2000 7310 Rhythm USB3 Interface
6   // Module Name:    main, command_selector
7   // Project Name:   RHDInterfaceXEM7310
8   // Target Devices:
9   // Tool versions:
10  // Description:    Uses Opal Kelly XEM7310 USB/FPGA board to interface multiple
11  //                 Intan Technologies RHD2000-series chips to a computer via a
12  //                 USB 3.0 connection.
13  //
14  //                 This software is provided 'as-is', without any express or implied
15  //                 warranty.  In no event will the authors be held liable for any
16  //                 damages arising from the use of this software.
17  //
18  //                 Permission is granted to anyone to use this software for any
19  //                 applications that use Intan Technologies integrated circuits, and
20  //                 to alter it and redistribute it freely.
21  //
22  //                 See http://www.intantech.com for documentation and product information.
23  //
24  // Dependencies:
25  //
26  // Revision:          3.0 (10 January 2023)
27  // Additional Comments:
28  //
29  //////////////////////////////////////////////////////////////////////////////////
30
31  // Company: University of Strathclyde
32  // Engineer: Ryan Greer
33
34  `timescale 1ns/1ps
35
36  module main
37      (
38      output wire mAXIS_FIFO_write_en,
39      output wire [15:0] mAXIS_FIFO_data_out,
40      output wire dataclk_out,
41      input wire [16:0] num_words_in_FIFO,
42      output wire [16:0] packet_size,
43
44      // parameters for loading auxiliary commands //
45      input wire  S_AXIS_ACLK,
46      input wire  S_AXIS_ARESETN,
47      output wire  S_AXIS_TREADY,
48      input wire [15 : 0] S_AXIS_TDATA,
49      input wire  S_AXIS_TLAST,
50      input wire  S_AXIS_TVALID,
51
52      // parameters for AXI4-lite slave (register map)
53  //    input wire trig_in, // temporary - change once I have register map set up
54      input wire S_AXI_ACLK,
55      input wire S_AXI_ARESETN,
56      input wire S_AXI_AWVALID,
```

```
57      output wire S_AXI_AWREADY,
58      input wire [9-1:0] S_AXI_AWADDR,
59      input wire [2:0] S_AXI_AWPROT,
60      input wire S_AXI_WVALID,
61      output wire S_AXI_WREADY,
62      input wire [32-1:0] S_AXI_WDATA,
63      input wire [32/8-1:0] S_AXI_WSTRB,
64      output wire S_AXI_BVALID,
65      input wire S_AXI_BREADY,
66      output wire [1:0] S_AXI_BRESP,
67       input wire S_AXI_ARVALID,
68       output wire      S_AXI_ARREADY,
69       input wire [9-1:0] S_AXI_ARADDR,
70       input wire [2:0] S_AXI_ARPROT,
71      output wire      S_AXI_RVALID,
72      input wire S_AXI_RREADY,
73      output wire      [32-1:0] S_AXI_RDATA,
74      output wire      [1:0] S_AXI_RRESP,
75
76
77  //    input  wire         sys_clk_p,
78  //    input  wire         sys_clk_n,
79      input wire sys_clk,
80
81      // SPI
82      output wire                             CS_b_A,
83      output wire                             SCLK_A,
84      output wire                             MOSI1_A,
85        output wire                                                                 MOSI2_A,
86      input  wire                             MISO1_A,
87      input  wire                             MISO2_A,
88
```

...

```
427
428     // connection to mAXIS FIFO for transferring data through DMA to PS DDR3
429     assign mAXIS_FIFO_write_en = FIFO_write_to;
430     assign mAXIS_FIFO_data_out = FIFO_data_in;
431
```

...

```
692
693      assign packet_size = ep15wirein[16:0];
694
695     // USB WireOut outputs
696     assign ep20wireout =                                {15'b0, num_words_in_FIFO}; // number of words in mAXIS FIFO
(capacity of 131072 currently)
```

...

```
760
761     // AXI-lite slave interface for parameters
762     sAXI4L_interface #(.C_AXI_ADDR_WIDTH(9),
763                        .C_AXI_DATA_WIDTH(32))
764     sAXI4L_interface_inst(.S_AXI_ACLK(S_AXI_ACLK),
765                           .S_AXI_ARESETN(S_AXI_ARESETN),
766                           .S_AXI_AWVALID(S_AXI_AWVALID),
767                           .S_AXI_AWREADY(S_AXI_AWREADY),
768                           .S_AXI_AWADDR(S_AXI_AWADDR),
769                           .S_AXI_AWPROT(S_AXI_AWPROT),
770                           .S_AXI_WVALID(S_AXI_WVALID),
771                           .S_AXI_WREADY(S_AXI_WREADY),
772                           .S_AXI_WDATA(S_AXI_WDATA),
773                           .S_AXI_WSTRB(S_AXI_WSTRB),
774                           .S_AXI_BVALID(S_AXI_BVALID),
775                           .S_AXI_BREADY(S_AXI_BREADY),
776                           .S_AXI_BRESP(S_AXI_BRESP),
777                           .S_AXI_ARVALID(S_AXI_ARVALID),
778                           .S_AXI_ARREADY(S_AXI_ARREADY),
779                           .S_AXI_ARADDR(S_AXI_ARADDR),
780                           .S_AXI_ARPROT(S_AXI_ARPROT),
781                           .S_AXI_RVALID(S_AXI_RVALID),
782                           .S_AXI_RREADY(S_AXI_RREADY),
```

```
783                             .S_AXI_RDATA(S_AXI_RDATA),
784                             .S_AXI_RRESP(S_AXI_RRESP),
785
786                             .dataclk(dataclk),
787
788                             .ep00wirein(ep00wirein), .ep01wirein(ep01wirein), .ep02wirein(ep02wirein),
.ep03wirein(ep03wirein), .ep04wirein(ep04wirein), .ep05wirein(ep05wirein), .ep06wirein(ep06wirein), .ep07wirein(ep07wirein),
789                             .ep08wirein(ep08wirein), .ep09wirein(ep09wirein), .ep0awirein(ep0awirein),
.ep0bwirein(ep0bwirein), .ep0cwirein(ep0cwirein), .ep0dwirein(ep0dwirein), .ep0ewirein(ep0ewirein), .ep0fwirein(ep0fwirein),
790                             .ep10wirein(ep10wirein), .ep11wirein(ep11wirein), .ep12wirein(ep12wirein),
.ep13wirein(ep13wirein), .ep14wirein(ep14wirein), .ep15wirein(ep15wirein), .ep16wirein(ep16wirein), .ep17wirein(ep17wirein),
791                             .ep18wirein(ep18wirein), .ep19wirein(ep19wirein), .ep1awirein(ep1awirein),
.ep1bwirein(ep1bwirein), .ep1cwirein(ep1cwirein), .ep1dwirein(ep1dwirein), .ep1ewirein(ep1ewirein), .ep1fwirein(ep1fwirein),
792
793                             .ep20wireout(ep20wireout), .ep21wireout(ep21wireout), .ep22wireout(ep22wireout),
.ep23wireout(ep23wireout), .ep24wireout(ep24wireout), .ep25wireout(ep25wireout), .ep26wireout(ep26wireout),
.ep27wireout(ep27wireout),
794                             .ep28wireout(ep28wireout), .ep29wireout(ep29wireout), .ep2awireout(ep2awireout),
.ep2bwireout(ep2bwireout), .ep2cwireout(ep2cwireout), .ep2dwireout(ep2dwireout), .ep2ewireout(ep2ewireout),
.ep2fwireout(ep2fwireout),
795                             .ep30wireout(ep30wireout), .ep31wireout(ep31wireout), .ep32wireout(ep32wireout),
.ep33wireout(ep33wireout), .ep34wireout(ep34wireout), .ep35wireout(ep35wireout), .ep36wireout(ep36wireout),
.ep37wireout(ep37wireout),
796                             .ep38wireout(ep38wireout), .ep39wireout(ep39wireout), .ep3awireout(ep3awireout),
.ep3bwireout(ep3bwireout), .ep3cwireout(ep3cwireout), .ep3dwireout(ep3dwireout), .ep3ewireout(ep3ewireout),
.ep3fwireout(ep3fwireout),
797
798                             .ep40trigin(ep40trigin), .ep41trigin(ep41trigin), .ep42trigin(ep42trigin)
799     );
800
801     // MOSI auxiliary commands sequence from AXI4-stream (custom, Ryan Greer)
802     wire [15:0] cmd_stream_data_1;
803     wire [15:0] cmd_stream_data_2;
804     wire [15:0] cmd_stream_data_3;
805     reg read_en;
806
807     aux_cmd_buffer # (.C_S_AXIS_TDATA_WIDTH(16),
808                       .FIFO_DATA_WIDTH(16),
809                       .FIFO_DEPTH(64))
810     aux_cmd_buffer_inst (
811         .S_AXIS_ACLK(S_AXIS_ACLK),
812         .S_AXIS_ARESETN(S_AXIS_ARESETN),
813         .S_AXIS_TREADY(S_AXIS_TREADY),
814         .S_AXIS_TDATA(S_AXIS_TDATA),
815         .S_AXIS_TLAST(S_AXIS_TLAST),
816         .S_AXIS_TVALID(S_AXIS_TVALID),
817         .cmd_stream_data_1(cmd_stream_data_1),
818         .cmd_stream_data_2(cmd_stream_data_2),
819         .cmd_stream_data_3(cmd_stream_data_3),
820         .dataclk(dataclk),
821         .read_en(read_en),
822         .num_words_in_FIFO(num_words_in_FIFO_aux)
823         );
824

...

1055
1056     always @(posedge dataclk) begin
1057             if (reset) begin

...

1072                     FIFO_data_in <= 16'b0;
1073                     FIFO_write_to <= 1'b0;
1074             end else begin
1075                     CS_b <= 1'b0;
1076                     SCLK <= 1'b0;
1077                     FIFO_data_in <= 16'b0;
1078                     FIFO_write_to <= 1'b0;
1079
1080                     case (main_state)
1081

...
```

```
1292
1293                            ms_clk1_d: begin
1294                                    if (channel == 31) begin
1295//                                          aux_cmd_A <= RAM_data_out_1;
1296                            aux_cmd_A <= cmd_stream_data_1;
1297                            read_en <= 1'b0;
1298                                    end else if (channel == 32) begin
1299//                                          aux_cmd_A <= RAM_data_out_2;
1300                            aux_cmd_A <= cmd_stream_data_2;
1301                            read_en <= 1'b0;
1302                                    end else if (channel == 33) begin
1303//                                          aux_cmd_A <= RAM_data_out_3;
1304                            aux_cmd_A <= cmd_stream_data_3;
1305                            read_en <= 1'b0;
1306                                    end
1307
1308                                    if (channel == 0) begin
1309                                            FIFO_data_in <= header_magic_number[47:32];
1310                                            FIFO_write_to <= 1'b1;
1311                                    end
1312
1313                                    SCLK <= 1'b1;
1314                                    in4x_A1[1] <= MISO_A1; in4x_A2[1] <= MISO_A2;
1315                                    in4x_B1[1] <= MISO_B1; in4x_B2[1] <= MISO_B2;
1316                                    in4x_C1[1] <= MISO_C1; in4x_C2[1] <= MISO_C2;
1317                                    in4x_D1[1] <= MISO_D1; in4x_D2[1] <= MISO_D2;
1318                                    in4x_E1[1] <= MISO_E1; in4x_E2[1] <= MISO_E2;
1319                                    in4x_F1[1] <= MISO_F1; in4x_F2[1] <= MISO_F2;
1320                                    in4x_G1[1] <= MISO_G1; in4x_G2[1] <= MISO_G2;
1321                                    in4x_H1[1] <= MISO_H1; in4x_H2[1] <= MISO_H2;
1322                                    main_state <= ms_clk2_a;
1323                            end
1324
```

sAXI4L_interface.v

```
1  `timescale 1ns / 1ps
2  // Ryan Greer
3  // University of Strathclyde, Glasgow
4  // Adapted from https://github.com/ZipCPU/wb2axip/blob/master/rtl/easyaxil.v#L66
5
6  module sAXI4L_interface #(
7      parameter C_AXI_ADDR_WIDTH = 9,
8      parameter C_AXI_DATA_WIDTH = 32
9      )
10     (
11     input wire S_AXI_ACLK,
12     input wire S_AXI_ARESETN,
13
14     input wire S_AXI_AWVALID,
15     output wire S_AXI_AWREADY,
16     input wire [C_AXI_ADDR_WIDTH-1:0] S_AXI_AWADDR,
17     input wire [2:0] S_AXI_AWPROT,
18
19     input wire S_AXI_WVALID,
20     output wire S_AXI_WREADY,
21     input wire [C_AXI_DATA_WIDTH-1:0] S_AXI_WDATA,
22     input wire [C_AXI_DATA_WIDTH/8-1:0] S_AXI_WSTRB,
23
24     output wire S_AXI_BVALID,
25     input wire S_AXI_BREADY,
26     output wire [1:0] S_AXI_BRESP,
27
28       input wire S_AXI_ARVALID,
29       output wire       S_AXI_ARREADY,
30       input wire [C_AXI_ADDR_WIDTH-1:0] S_AXI_ARADDR,
31       input wire [2:0] S_AXI_ARPROT,
32
33     output wire        S_AXI_RVALID,
34     input wire S_AXI_RREADY,
35     output wire        [C_AXI_DATA_WIDTH-1:0] S_AXI_RDATA,
36     output wire        [1:0] S_AXI_RRESP,
```

```
37
38      input wire dataclk,
39
40      output reg[31:0] ep00wirein, ep01wirein, ep02wirein, ep03wirein, ep04wirein, ep05wirein, ep06wirein, ep07wirein,
41      ep08wirein, ep09wirein, ep0awirein, ep0bwirein, ep0cwirein, ep0dwirein, ep0ewirein, ep0fwirein,
42      ep10wirein, ep11wirein, ep12wirein, ep13wirein, ep14wirein, ep15wirein, ep16wirein, ep17wirein,
43      ep18wirein, ep19wirein, ep1awirein, ep1bwirein, ep1cwirein, ep1dwirein, ep1ewirein, ep1fwirein,
44
45      input wire[31:0] ep20wireout, ep21wireout, ep22wireout, ep23wireout, ep24wireout, ep25wireout, ep26wireout,
ep27wireout,
46      ep28wireout, ep29wireout, ep2awireout, ep2bwireout, ep2cwireout, ep2dwireout, ep2ewireout, ep2fwireout,
47      ep30wireout, ep31wireout, ep32wireout, ep33wireout, ep34wireout, ep35wireout, ep36wireout, ep37wireout,
48      ep38wireout, ep39wireout, ep3awireout, ep3bwireout, ep3cwireout, ep3dwireout, ep3ewireout, ep3fwireout,
49
50      output reg[31:0] ep40trigin, ep41trigin, ep42trigin
51      );
52
53      // Register/wire signal declarations
54       localparam       ADDRLSB = $clog2(C_AXI_DATA_WIDTH)-3;
55
56       wire i_reset = !S_AXI_ARESETN;
57
58       wire axil_write_ready;
59       wire [C_AXI_ADDR_WIDTH-ADDRLSB-1:0]        awskd_addr;
60       //
61       wire [C_AXI_DATA_WIDTH-1:0]       wskd_data;
62       wire [C_AXI_DATA_WIDTH/8-1:0] wskd_strb;
63       reg axil_bvalid;
64       //
65       wire axil_read_ready;
66       wire [C_AXI_ADDR_WIDTH-ADDRLSB-1:0]        arskd_addr;
67       reg      [C_AXI_DATA_WIDTH-1:0] axil_read_data;
68       reg axil_read_valid;
69
70      reg [31:0] epwirein [31:0];
71      reg [31:0] epwireout [31:0];
72      reg [31:0] eptrigin [2:0];
73
74      wire [31:0] wskd_epwirein [31:0];
75      wire [31:0] wskd_epwireout [31:0];
76      wire [31:0] wskd_eptrigin [2:0];
77
78      // Simple writes, no skid buffer
79      reg       axil_awready;
80
81      initial   axil_awready = 1'b0;
82      always @(posedge S_AXI_ACLK)
83      if (!S_AXI_ARESETN)
84          axil_awready <= 1'b0;
85      else
86          axil_awready <= !axil_awready
87              && (S_AXI_AWVALID && S_AXI_WVALID)
88              && (!S_AXI_BVALID || S_AXI_BREADY);
89
90      assign    S_AXI_AWREADY = axil_awready;
91      assign    S_AXI_WREADY  = axil_awready;
92
93      assign    awskd_addr = S_AXI_AWADDR[C_AXI_ADDR_WIDTH-1:ADDRLSB];
94      assign    wskd_data  = S_AXI_WDATA;
95      assign    wskd_strb  = S_AXI_WSTRB;
96
97      assign    axil_write_ready = axil_awready;
98
99      // assign ports to register values
100     always @(posedge S_AXI_ACLK) begin
101         ep00wirein = epwirein[0];
102         ep01wirein = epwirein[1];
103         ep02wirein = epwirein[2];
104         ep03wirein = epwirein[3];
105         ep04wirein = epwirein[4];
106         ep05wirein = epwirein[5];
107         ep06wirein = epwirein[6];
108         ep07wirein = epwirein[7];
109         ep08wirein = epwirein[8];
110         ep09wirein = epwirein[9];
```

```
111         ep0awirein = epwirein[10];
112         ep0bwirein = epwirein[11];
113         ep0cwirein = epwirein[12];
114         ep0dwirein = epwirein[13];
115         ep0ewirein = epwirein[14];
116         ep0fwirein = epwirein[15];
117         ep10wirein = epwirein[16];
118         ep11wirein = epwirein[17];
119         ep12wirein = epwirein[18];
120         ep13wirein = epwirein[19];
121         ep14wirein = epwirein[20];
122         ep15wirein = epwirein[21];
123         ep16wirein = epwirein[22];
124         ep17wirein = epwirein[23];
125         ep18wirein = epwirein[24];
126         ep19wirein = epwirein[25];
127         ep1awirein = epwirein[26];
128         ep1bwirein = epwirein[27];
129         ep1cwirein = epwirein[28];
130         ep1dwirein = epwirein[29];
131         ep1ewirein = epwirein[30];
132         ep1fwirein = epwirein[31];
133
134         ep40trigin = eptrigin[0];
135         ep41trigin = eptrigin[1];
136         ep42trigin = eptrigin[2];
137     end
138
139     initial   axil_bvalid = 0;
140       always @(posedge S_AXI_ACLK)
141       if (i_reset)
142                 axil_bvalid <= 0;
143       else if (axil_write_ready)
144                 axil_bvalid <= 1;
145       else if (S_AXI_BREADY)
146                 axil_bvalid <= 0;
147
148       assign   S_AXI_BVALID = axil_bvalid;
149       assign   S_AXI_BRESP = 2'b00;
150
151
152     reg       axil_aready;
153
154     always @(*)
155         axil_aready = !S_AXI_RVALID;
156
157     assign    arskd_addr = S_AXI_ARADDR[C_AXI_ADDR_WIDTH-1:ADDRLSB];
158     assign    S_AXI_ARREADY = axil_aready;
159     assign    axil_read_ready = (S_AXI_ARVALID && S_AXI_ARREADY);
160
161
162       initial  axil_read_valid = 1'b0;
163       always @(posedge S_AXI_ACLK)
164       if (i_reset)
165                 axil_read_valid <= 1'b0;
166       else if (axil_read_ready)
167                 axil_read_valid <= 1'b1;
168       else if (S_AXI_RREADY)
169                 axil_read_valid <= 1'b0;
170
171       assign   S_AXI_RVALID = axil_read_valid;
172       assign   S_AXI_RDATA  = axil_read_data;
173       assign   S_AXI_RRESP = 2'b00;
174
175       // AXI-lite register logic
176     genvar i;
177     generate
178     for(i=0; i<32; i=i+1) begin
179         assign wskd_epwirein[i] = apply_wstrb(epwirein[i], wskd_data, wskd_strb);
180     end
181     endgenerate
182     generate
183     for(i=0; i<32; i=i+1) begin
184         assign wskd_epwireout[i] = apply_wstrb(epwireout[i], wskd_data, wskd_strb);
185     end
```

```
186     endgenerate
187     generate
188     for(i=0; i<3; i=i+1) begin
189         assign wskd_eptrigin[i] = apply_wstrb(eptrigin[i], wskd_data, wskd_strb);
190     end
191     endgenerate
192
193     generate
194     for(i=0; i<32; i=i+1) begin
195         initial epwirein[i] = 0;
196     end
197     endgenerate
198     generate
199     for(i=0; i<32; i=i+1) begin
200         initial epwireout[i] = 0;
201     end
202     endgenerate
203     generate
204     for(i=0; i<3; i=i+1) begin
205         initial eptrigin[i] = 0;
206     end
207     endgenerate
208
209     reg [6:0] j;
210
211     reg [6:0] k;
212
213       always @(posedge S_AXI_ACLK) begin
214         if (i_reset) begin
215             for(j=7'd0; j<7'd32; j=j+1) begin
216                 epwirein[j] <= 0;
217             end
218             for(j=7'd0; j<7'd32; j=j+1) begin
219                 epwireout[j] <= 0;
220             end
221             for(j=7'd0; j<7'd3; j=j+1) begin
222                 eptrigin[j] <= 0;
223             end
224
225         end else if (axil_write_ready)begin
226             for(k=7'd0; k<7'd32; k=k+1) begin
227                 if(awskd_addr == k) begin
228                     epwirein[k] <= wskd_epwirein[k];
229                 end
230             end
231             // these registers cannot be written from AXI-lite interface
232             for(k=7'd0; k<7'd3; k=k+1) begin
233                 if(awskd_addr == (k+64)) begin
234                     eptrigin[k] <= wskd_eptrigin[k];
235                 end
236             end
237
238         end
239         // wire out registers are always set to signal values from top-level
240         epwireout[0] <= ep20wireout;
241         epwireout[1] <= ep21wireout;
242         epwireout[2] <= ep22wireout;
243         epwireout[3] <= ep23wireout;
244         epwireout[4] <= ep24wireout;
245         epwireout[5] <= ep25wireout;
246         epwireout[6] <= ep26wireout;
247         epwireout[7] <= ep27wireout;
248         epwireout[8] <= ep28wireout;
249         epwireout[9] <= ep29wireout;
250         epwireout[10] <= ep2awireout;
251         epwireout[11] <= ep2bwireout;
252         epwireout[12] <= ep2cwireout;
253         epwireout[13] <= ep2dwireout;
254         epwireout[14] <= ep2ewireout;
255         epwireout[15] <= ep2fwireout;
256         epwireout[16] <= ep30wireout;
257         epwireout[17] <= ep31wireout;
258         epwireout[18] <= ep32wireout;
259         epwireout[19] <= ep33wireout;
260         epwireout[20] <= ep34wireout;
```

```
261         epwireout[21] <= ep35wireout;
262         epwireout[22] <= ep36wireout;
263         epwireout[23] <= ep37wireout;
264         epwireout[24] <= ep38wireout;
265         epwireout[25] <= ep39wireout;
266         epwireout[26] <= ep3awireout;
267         epwireout[27] <= ep3bwireout;
268         epwireout[28] <= ep3cwireout;
269         epwireout[29] <= ep3dwireout;
270         epwireout[30] <= ep3ewireout;
271         epwireout[31] <= ep3fwireout;
272     end
273
274       initial  axil_read_data = 0;
275       always @(posedge S_AXI_ACLK)
276     if (!S_AXI_RVALID || S_AXI_RREADY)
277       begin
278         for(k=7'd0; k<7'd32; k=k+1) begin
279             if(arskd_addr == k) begin
280                     axil_read_data <= epwirein[k];
281                 end
282             end
283         for(k=7'd0; k<7'd32; k=k+1) begin
284             if(arskd_addr == (k+32)) begin
285                     axil_read_data <= epwireout[k];
286                 end
287             end
288         for(k=7'd0; k<7'd3; k=k+1) begin
289             if(arskd_addr == (k+64)) begin
290                     axil_read_data <= eptrigin[k];
291                 end
292             end
293     end
294
295       function [C_AXI_DATA_WIDTH-1:0]    apply_wstrb;
296               input    [C_AXI_DATA_WIDTH-1:0]             prior_data;
297               input    [C_AXI_DATA_WIDTH-1:0]             new_data;
298               input    [C_AXI_DATA_WIDTH/8-1:0]  wstrb;
299
300               integer  k;
301               for(k=0; k<C_AXI_DATA_WIDTH/8; k=k+1)
302               begin
303                       apply_wstrb[k*8 +: 8]
304                               = wstrb[k] ? new_data[k*8 +: 8] : prior_data[k*8 +: 8];
305               end
306       endfunction
307
308
309endmodule
310
```

aux_cmd_buffer.v

```
1   `timescale 1ns / 1ps
2   // Ryan Greer
3   // University of Strathclyde, Glasgow
4
5   module aux_cmd_buffer #
6       (
7       parameter integer C_S_AXIS_TDATA_WIDTH      = 16,
8
9       parameter integer FIFO_DATA_WIDTH = 16,
10      parameter integer FIFO_DEPTH = 64               // must be a power of 2 - maximum size of FIFO to hold auxiliary
commands
11      )
12      (
13      input wire  S_AXIS_ACLK,
14      input wire  S_AXIS_ARESETN,
15      output wire  S_AXIS_TREADY,
16      input wire [C_S_AXIS_TDATA_WIDTH-1 : 0] S_AXIS_TDATA,
17      input wire  S_AXIS_TLAST,
18      input wire  S_AXIS_TVALID,
19
20      output reg [FIFO_DATA_WIDTH-1:0] cmd_stream_data_1,
```

```
21      output reg [FIFO_DATA_WIDTH-1:0] cmd_stream_data_2,
22      output reg [FIFO_DATA_WIDTH-1:0] cmd_stream_data_3,
23
24      input wire dataclk,
25      input wire read_en,
26
27      output wire [5:0] num_words_in_FIFO
28      );
29
30        function integer clogb2 (input integer bit_depth);
31           begin
32               bit_depth = bit_depth - 1;
33               for(clogb2=0; bit_depth>0; clogb2=clogb2+1)
34                   bit_depth = bit_depth >> 1;
35           end
36        endfunction
37
38        parameter [1:0] EMPTY = 2'b00,
39                        NORMAL = 2'b01,
40                        FULL = 2'b10;
41
42        // Implementation of FIFO buffer as distributed RAM
43        reg [FIFO_DATA_WIDTH-1:0] fifo[FIFO_DEPTH-1:0];
44
45        reg [1:0] fifo_state;                   // empty, normal, full
46        reg [clogb2(FIFO_DEPTH)-1:0] read_pointer;      // next data to read
47        wire axis_tready;                       // internal S_AXIS TREADY signal
48        reg [1:0] aux_cmd_counter;              // keeps track of which auxiliary command we should be sending FIFO data to
49
50      reg [clogb2(FIFO_DEPTH)-1:0] write_pointer;      // next data to write
51
52        assign num_words_in_FIFO = write_pointer - read_pointer;       // number of words currently in FIFO
53
54      reg [clogb2(FIFO_DEPTH)-1:0] read_pointer_s_axis_aclk;
55      always @(posedge S_AXIS_ACLK) begin
56          read_pointer_s_axis_aclk <= read_pointer;
57      end
58
59      reg [5:0] num_words_in_FIFO_reg;
60      always @(posedge S_AXIS_ACLK) begin
61          num_words_in_FIFO_reg <= write_pointer - read_pointer_s_axis_aclk;
62      end
63
64      assign axis_tready = (num_words_in_FIFO_reg < 6'd63);
65
66      // these signals allow us to buffer a set of commands and stream them in sequence (this sequence of commands is
identified as an AXI stream packet)
67      reg stream_command_sequence;           // If end of AXI-stream packet (TLAST enabled), this is latched high and stays
high until current data in FIFO is read
68      reg command_stream_done;
69      reg [clogb2(FIFO_DEPTH)-1:0] stream_command_length;      // FIFO can be filled as stream of commands being sent
70      reg [clogb2(FIFO_DEPTH)-1:0] stream_command_counter;      // count number of commands sent
71
72      // An AXI stream data transfer will occur on the next clock edge
73      assign transfer_en = S_AXIS_TVALID && S_AXIS_TREADY;
74      assign S_AXIS_TREADY = axis_tready_reg;
75
76      reg axis_tready_reg;
77      always @(posedge S_AXIS_ACLK) begin
78          if(!S_AXIS_ARESETN) begin
79              axis_tready_reg <= 1'b0;
80          end else begin
81              axis_tready_reg <= axis_tready;
82          end
83      end
84
85      reg [FIFO_DATA_WIDTH-1:0] cmd_stream_data;
86
87      reg dataclk_reset;
88      reg dataclk_reset_done;
89
90      // Write to FIFO
91      always @(posedge S_AXIS_ACLK) begin
92          if(dataclk_reset_done == 1'b1) begin
93              dataclk_reset <= 1'b0;
```

```
94          end
95          if(!S_AXIS_ARESETN) begin
96              write_pointer <= 0;
97              dataclk_reset <= 1'b1;
98 //             test_counter <= 6'b0;
99          end else if(transfer_en) begin
100             fifo[write_pointer] <= S_AXIS_TDATA;
101             write_pointer <= write_pointer + 1;
102
103             // if the transfer that executed on this clock edge was the last one in a data packet
104             if(num_words_in_FIFO_reg == 6'd59) begin
105                 stream_command_sequence <= 1;
106                 stream_command_length = 6'd60;
107             end
108         end
109
110         if(command_stream_done) begin
111             stream_command_sequence <= 0;
112         end
113     end
114
115     reg stream_command_sequence_dataclk;
116     reg [clogb2(FIFO_DEPTH)-1:0] stream_command_length_dataclk;
117
118     always @(posedge dataclk) begin
119         stream_command_sequence_dataclk <= stream_command_sequence;
120         stream_command_length_dataclk <= stream_command_length;
121     end
122
123     // Read from FIFO - cmd_stream_data_1, 2 or 3 will be available on the next clock edge
124     always @(posedge dataclk) begin
125         if(dataclk_reset == 1'b1) begin
126             read_pointer <= 0;
127             aux_cmd_counter <= 0;
128             command_stream_done <= 1'b0;
129             stream_command_counter <= 0;
130             dataclk_reset_done <= 1'b1;
131         end else if(read_en) begin
132             dataclk_reset_done <= 1'b0;
133             if(stream_command_sequence_dataclk && ~command_stream_done) begin
134                 if(stream_command_counter != stream_command_length_dataclk) begin
135                     // sending auxiliary command stream
136                     if(aux_cmd_counter == 2'b00) begin
137                         cmd_stream_data_1 <= fifo[read_pointer];
138                     end else if(aux_cmd_counter == 2'b01) begin
139                         cmd_stream_data_2 <= fifo[read_pointer];
140                     end else if(aux_cmd_counter == 2'b10) begin
141                         cmd_stream_data_3 <= fifo[read_pointer];
142                     end
143                     stream_command_counter <= stream_command_counter + 1;
144                     read_pointer <= read_pointer + 1;
145                     command_stream_done <= 1'b0;
146                 end else begin
147                     // end of auxiliary command stream
148                     if(aux_cmd_counter == 2'b00) begin
149                         cmd_stream_data_1 <= 16'b1111111100000000;
150                     end else if(aux_cmd_counter == 2'b01) begin
151                         cmd_stream_data_2 <= 16'b1111111100000000;
152                     end else if(aux_cmd_counter == 2'b10) begin
153                         cmd_stream_data_3 <= 16'b1111111100000000;
154                     end
155                     command_stream_done <= 1'b1;
156                     stream_command_counter <= 0;
157                 end
158             end else begin
159                 if(aux_cmd_counter == 2'b00) begin
160                     cmd_stream_data_1 <= 16'b1111111100000000;
161                 end else if(aux_cmd_counter == 2'b01) begin
162                     cmd_stream_data_2 <= 16'b1111111100000000;
163                 end else if(aux_cmd_counter == 2'b10) begin
164                     cmd_stream_data_3 <= 16'b1111111100000000;
165                 end
166                 command_stream_done <= 1'b0;
167                 stream_command_counter <= 0;
168             end
```

```
169             if(aux_cmd_counter == 2'b10) begin
170                 aux_cmd_counter <= 2'b00;
171             end else begin
172                 aux_cmd_counter <= aux_cmd_counter + 1;
173             end
174         end else begin
175             dataclk_reset_done <= 1'b0;
176         end
177     end
178 
179endmodule
180 
```

mAXIS_FIFO_v1.v

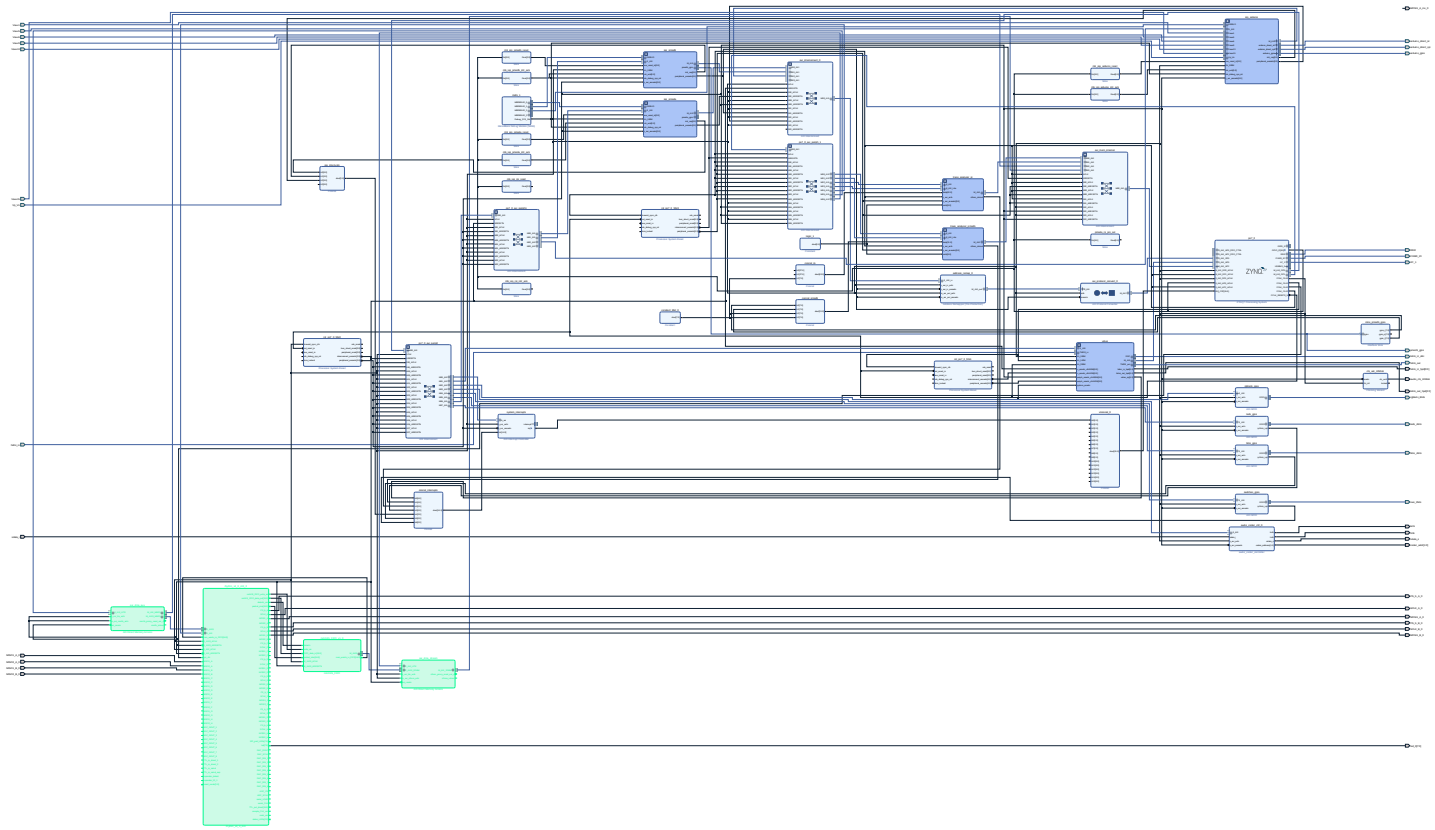
```
1  // Company: University of Strathclyde
2  // Engineer: Ryan Greer
3  
4  `timescale 1 ns / 1 ps
5  
6        module mAXI4S_FIFO_v1 #
7        (
8                parameter integer FIFO_DEPTH = 131072,                // must be a power of 2 as the pointers wrap
9                parameter integer FIFO_DATA_WIDTH = 16,
10 
11               // Do not modify the parameters beyond this line
12               parameter integer C_M_AXIS_TDATA_WIDTH     = 16      // Width of S_AXIS address bus. The slave accepts the
read and write addresses of width C_M_AXIS_TDATA_WIDTH.
13       )
14       (
15           input wire dataclk,
16         input wire write_en,
17         input wire [FIFO_DATA_WIDTH-1:0] FIFO_data_in,
18         input wire [16:0] packet_size,
19          output wire [16:0] num_words_in_FIFO,
20 
21               // Do not modify the ports beyond this line
22               input wire  M_AXIS_ACLK,
23               input wire  M_AXIS_ARESETN,   // syncronous reset is active low
24               output wire  M_AXIS_TVALID, // TVALID indicates that the master is driving a valid transfer, A transfer
takes place when both TVALID and TREADY are asserted.
25               output wire [C_M_AXIS_TDATA_WIDTH-1 : 0] M_AXIS_TDATA, // TDATA is the primary payload that is used to
provide the data that is passing across the interface from the master.
26               output wire  M_AXIS_TLAST,    // TLAST indicates the boundary of a packet.
27               input wire  M_AXIS_TREADY     // TREADY indicates that the slave can accept a transfer in the current
cycle.
28       );
29 
30       function integer clogb2 (input integer bit_depth);
31          begin
32              bit_depth = bit_depth - 1;                                  // this line is not in the Xilinx example
33              for(clogb2=0; bit_depth>0; clogb2=clogb2+1)
34                  bit_depth = bit_depth >> 1;
35          end
36       endfunction
37 
38       parameter [1:0] EMPTY = 2'b00,
39                       NORMAL = 2'b01,
40                       FULL = 2'b10;
41 
42       reg [1:0] fifo_state;                  // empty, normal, full
43       reg [clogb2(FIFO_DEPTH)-1:0] read_pointer;      // Example design FIFO read pointer
44 
45      reg [clogb2(FIFO_DEPTH)-1:0] read_pointer_next;
46 
47 //    integer packet_counter;             // keeps track of how many transfers have taken place and sets TLAST accordingly
48      reg[clogb2(FIFO_DEPTH)-1:0] packet_counter;
49 
50       // AXI Stream internal signals
51      reg     axis_tvalid;
52        reg     axis_tvalid_delay;                  // streaming data valid delayed by one clock cycle
53      reg     axis_tlast;
54        reg     axis_tlast_delay;                   // Last of the streaming data delayed by one clock module
55        reg[C_M_AXIS_TDATA_WIDTH-1:0] axis_tdata;
```

```
56
57        // signals for FIFO write
58      reg [1:0] write_state;
59      reg [clogb2(FIFO_DEPTH)-1:0] write_pointer;
60
61      reg [16:0] num_words_in_FIFO_reg;
62      reg [16:0] num_words_in_FIFO_next_reg;
63
64      reg fifo_empty_flag;
65
66      // Implementation of FIFO buffer as BRAM
67      (* ram_style = "block" *) reg [FIFO_DATA_WIDTH-1:0] fifo[FIFO_DEPTH-1:0];
68
69        // I/O Connections assignments
70      assign M_AXIS_TVALID = axis_tvalid;
71      assign M_AXIS_TLAST = axis_tlast;
72        assign M_AXIS_TSTRB       = {(C_M_AXIS_TDATA_WIDTH/8){1'b1}};
73      assign M_AXIS_TDATA = axis_tdata;
74
75        assign num_words_in_FIFO = write_pointer - read_pointer;
76
77        reg [clogb2(FIFO_DEPTH)-1:0] write_pointer_m_axis;
78        always @(posedge M_AXIS_ACLK) begin
79           write_pointer_m_axis <= write_pointer;
80        end
81
82        always @(posedge M_AXIS_ACLK) begin
83         num_words_in_FIFO_next_reg <= write_pointer_m_axis - read_pointer_next;
84        end
85
86        // Delay the axis_tvalid and axis_tlast signal by one clock cycle to match the latency of M_AXIS_TDATA
87        always @(posedge M_AXIS_ACLK) begin
88          if (!M_AXIS_ARESETN) begin
89              axis_tvalid_delay <= 1'b0;
90              axis_tlast_delay <= 1'b0;
91            end
92          else begin
93              axis_tvalid_delay <= axis_tvalid;
94              axis_tlast_delay <= axis_tlast;
95            end
96        end
97
98        // FIFO read enable generation - an AXI4-stream transfer will occur on the next clock edge if this state is
maintained (TREADY can be de-asserted)
99        assign transfer_en = M_AXIS_TREADY && M_AXIS_TVALID;
100
101     always @(posedge M_AXIS_ACLK) begin
102         // if transfer has actually occured, increment read_pointer
103         if(!M_AXIS_ARESETN) begin
104             read_pointer <= 0;
105             read_pointer_next <= 0;
106             axis_tvalid <= 1'b0;
107             axis_tdata <= 16'b0;
108             fifo_empty_flag <= 1'b1;
109         end else if(transfer_en && (num_words_in_FIFO_next_reg == 17'd1)) begin     // transfer occured on this clock
edge
110             // Transfer occurred on this clock edge - FIFO now empty
111             read_pointer <= read_pointer + 1;       // increment read pointer
112             axis_tvalid <= 1'b0;
113             axis_tdata <= axis_tdata;
114             read_pointer_next <= read_pointer_next;
115             fifo_empty_flag <= 1'b1;
116
117         end else if(transfer_en) begin     // transfer occured on this clock edge
118             // Transfer occured on this clock edge and data is in FIFO - prepare next TDATA
119             read_pointer <= read_pointer + 1;       // increment read pointer
120             axis_tvalid <= 1'b1;
121             axis_tdata <= fifo[read_pointer_next];
122             read_pointer_next <= read_pointer_next + 1; // should increment read_pointer_next
123             fifo_empty_flag <= 1'b0;
124
125         end else if((num_words_in_FIFO_next_reg == 17'd0) && (fifo_empty_flag == 1'b1)) begin
126             // FIFO is empty (not full), no transfer occurred on this clock edge
127             read_pointer <= read_pointer;
128             axis_tvalid <= 1'b0;
```

```
129             axis_tdata <= axis_tdata;
130             read_pointer_next <= read_pointer;
131             fifo_empty_flag <= 1'b1;
132 
133         end else if(read_pointer_next == read_pointer) begin
134             // No transfer on this clock edge, retrieve data from FIFO for next transfer (I think this is for the very
first when the FIFO starts empty)
135             read_pointer <= read_pointer;
136             axis_tvalid <= 1'b1;
137             axis_tdata <= fifo[read_pointer_next];
138             read_pointer_next <= read_pointer + 1;
139             fifo_empty_flag <= 1'b0;
140 
141         end else begin
142             // No transfer on this clock edge, TDATA is ready
143             read_pointer <= read_pointer;
144             axis_tvalid <= 1'b1;
145             axis_tdata <= axis_tdata;
146             read_pointer_next <= read_pointer_next;
147 
148         end
149 
150     end
151 
152     always @(posedge M_AXIS_ACLK) begin
153         // if transfer has actually occured, increment packet_counter
154         if(!M_AXIS_ARESETN) begin
155             packet_counter <= 0;
156             axis_tlast <= 1'b0;
157 
158         // current with nested ifs //
159         end else if(transfer_en) begin
160             if(packet_counter == (packet_size-1)) begin
161                 packet_counter <= 0;
162                 axis_tlast <= 1'b0;
163             end else if(packet_counter == (packet_size-2)) begin
164                 packet_counter <= packet_counter + 1;
165                 axis_tlast <= 1'b1;
166             end else begin
167                 packet_counter <= packet_counter + 1;
168                 axis_tlast <= 1'b0;
169             end
170         end
171     end
172 
173     // Write to FIFO
174     always @(posedge dataclk) begin
175         if(!M_AXIS_ARESETN) begin
176             write_pointer <= 0;
177         end else if(write_en) begin
178             fifo[write_pointer] <= FIFO_data_in;
179             write_pointer <= write_pointer + 1;
180         end else begin
181             write_pointer <= write_pointer;
182         end
183     end
184 
185      endmodule
186 
```

C.2 Vivado IP Integrator block diagram - Rhythm AXI integrated with PYNQ base overlay

Note: Block diagram provided as a vector. For full resolution, please view on a PC. Highlighted in light green is the RHD2000 subsystem which was added to the base overlay (remainder of the block diagram).



C.3 µLED array driver code listings

driver.c

```
// PYNQ Microblaze driver for HearLight LED control system
// Ryan Greer, University of Strathclyde
// 9th December 2022
// Updated for new control system v2: June 2025

#include <gpio.h>
#include <spi.h>
#include <timer.h>
#include <pyprintf.h>
#include <stdint.h>

#define LTC2662_CMD_POWER_DOWN_N 0x40
#define LTC2662_CMD_SPAN 0x60
#define LTC2662_CMD_WRITE_N_UPDATE_N 0x30
#define LTC2662_CMD_WRITE_CODE_N 0x00
#define LTC2662_CMD_UPDATE_ALL 0x90
#define LTC2662_CMD_UPDATE_N 0x10
#define LTC2662_CMD_WRITE_ALL_UPDATE_ALL 0xA0


#include <xparameters.h>
#include <xsysmon.h>
#define SYSMON_DEVICE_ID XPAR_SYSMON_0_DEVICE_ID

static XSysMon SysMonInst;
XSysMon_Config *SysMonConfigPtr;
XSysMon *SysMonInstPtr = &SysMonInst;

spi dac;
gpio cs_a;
gpio cs_b;
gpio cs_c;
gpio cs_d;
gpio sr_data;
gpio sr_clk;
gpio sr_latch;
gpio sr_clr_n;

gpio trig_in_1;
gpio trig_in_2;

int16_t fault_code = 0;
uint32_t max_switch_counts = 0;
uint32_t max_device_counts = 0;
float dac_refs[8] = {3.125, 6.25, 12.5, 25, 50, 100, 200, 300};

/*
Function to transfer block through SPI.
*/
uint8_t dac_write(gpio *cs_handle, uint8_t dac_command, uint8_t selected_dac, uint16_t dac_code){
    // length 3 which corresponds to shorter address of 24 bits
    char tx_array[3];
    char rx_array[3];

    tx_array[0] = dac_command + selected_dac;
    tx_array[1] = (dac_code >> 8) & 0xFF;
    tx_array[2] = dac_code & 0xFF;

    gpio_write(*cs_handle, 0);
    spi_transfer(dac, tx_array, rx_array, 3);
    gpio_write(*cs_handle, 1);

```

```
63      return rx_array[0]; // fault code
64  }
65
66  /*
67  sets softspan range of all DACs based on 'current_ref'
68              'current_ref' sets softspan range (maximum current)
69                  0 : 3.125 mA
70                  1 : 6.25 mA
71                  2 : 12.5 mA
72                  3 : 25 mA
73                  4 : 50 mA
74                  5 : 100 mA
75                  6 : 200 mA
76                  7 : 300 mA
77  */
78  uint8_t dac_config(uint8_t channel, uint16_t current_ref){
79      uint8_t fault_reg = 0;
80
81      uint8_t selected_dac = channel % 5;
82
83      uint8_t user_command = 0;
84
85      if(current_ref == 0){
86          user_command = 1;
87      }
88      else if(current_ref == 1){
89          user_command = 2;
90      }
91      else if(current_ref == 2){
92          user_command = 3;
93      }
94      else if(current_ref == 3){
95          user_command = 4;
96      }
97      else if(current_ref == 4){
98          user_command = 5;
99      }
100     else if(current_ref == 5){
101         user_command = 6;
102     }
103     else if(current_ref == 6){
104         user_command = 7;
105     }
106     else if(current_ref == 7){
107         user_command = 15;
108     }
109     else if(current_ref == 8){
110         user_command = 8;
111     }
112
113     if((channel >= 0) && (channel < 5)){
114         fault_reg |= dac_write(&cs_a, LTC2662_CMD_SPAN, selected_dac, user_command);
115     }
116     else if((channel >= 5) && (channel < 10)){
117         fault_reg |= dac_write(&cs_b, LTC2662_CMD_SPAN, selected_dac, user_command);
118     }
119     else if((channel >= 10) && (channel < 15)){
120         fault_reg |= dac_write(&cs_c, LTC2662_CMD_SPAN, selected_dac, user_command);
121     }
122     else{
123         fault_reg |= dac_write(&cs_d, LTC2662_CMD_SPAN, selected_dac, user_command);
124     }
125
126     return fault_reg;
127 }
128
129 /*
130 Switches on/off DAC channel indicated by 'channel' argument and considers both DACs
131             'channel' => 0->19
132             'on_off' => '1' is on and '0' is off
133             'current_code' => current value in DAC counts
134 */
135 uint8_t dac_channel_control(uint8_t channel, uint8_t on_off, uint16_t current_code){
136     uint8_t fault_reg = 0;
137
```

```
138     uint8_t selected_dac = channel % 5;
139 
140     if(on_off){
141         if((channel >= 0) && (channel < 5)){
142             fault_reg |= dac_write(&cs_a, LTC2662_CMD_WRITE_N_UPDATE_N, selected_dac, current_code);
143         }
144         else if((channel >= 5) && (channel < 10)){
145             fault_reg |= dac_write(&cs_b, LTC2662_CMD_WRITE_N_UPDATE_N, selected_dac, current_code);
146         }
147         else if((channel >= 10) && (channel < 15)){
148             fault_reg |= dac_write(&cs_c, LTC2662_CMD_WRITE_N_UPDATE_N, selected_dac, current_code);
149         }
150         else{
151             fault_reg |= dac_write(&cs_d, LTC2662_CMD_WRITE_N_UPDATE_N, selected_dac, current_code);
152         }
153     }
154     else{
155         if((channel >= 0) && (channel < 5)){
156             fault_reg |= dac_write(&cs_a, LTC2662_CMD_POWER_DOWN_N, selected_dac, current_code);
157         }
158         else if((channel >= 5) && (channel < 10)){
159             fault_reg |= dac_write(&cs_b, LTC2662_CMD_POWER_DOWN_N, selected_dac, current_code);
160         }
161         else if((channel >= 10) && (channel < 15)){
162             fault_reg |= dac_write(&cs_c, LTC2662_CMD_POWER_DOWN_N, selected_dac, current_code);
163         }
164         else{
165             fault_reg |= dac_write(&cs_d, LTC2662_CMD_POWER_DOWN_N, selected_dac, current_code);
166         }
167     }
168 
169     return fault_reg;
170 }
171 
172 /*
173 Opens/closes switches in the switch arrays based on 'sw' argument
174             'open_close' = '1' is switch closed and '0' is switch open
175 */
176 void switch_control(gpio *sw, uint8_t open_close){
177     if(open_close){
178         gpio_write(*sw, 0);
179     }
180     else{
181         gpio_write(*sw, 1);
182     }
183 }
184 
185 /*
186 Opens/closes switches in the switch arrays based on 'sw' argument using shift register
187             'open_close' = '1' is switch closed and '0' is switch open
188 */
189 void switch_control_sr(uint8_t sw, uint8_t open_close){
190 
191 }
192 
193 /*
194 Returns 16-bit fault code which indicates which channels have had current corrected due to exceeding limit on system.
195 10 least significant bits with '1' indicating fault, '0' with no fault, LSB is channel 1 (P1).
196 */
197 int16_t leds_read_fault(){
198     return fault_code;
199 }
200 
201 /*
202 Configures DACs and GPIOs for switches, must be called before 'leds_start'
203 */
204 void leds_configure(uint8_t dac_ref, int max_switch_current, int max_device_current){
205     // RPC handle to control DACs through SPI
206     dac = spi_open(13, 12, 11, 6);
207     delay_ms(100); // small delay required or first SPI transfers not recognised
208 
209     // chip select for DACs A to D B - initialise to 'high' (slave disabled)
210     cs_a = gpio_open(9);
211     gpio_set_direction(cs_a, GPIO_OUT);
212     gpio_write(cs_a, 1);
```

```
213
214     cs_b = gpio_open(7);
215     gpio_set_direction(cs_b, GPIO_OUT);
216     gpio_write(cs_b, 1);
217
218     cs_c = gpio_open(10);
219     gpio_set_direction(cs_c, GPIO_OUT);
220     gpio_write(cs_c, 1);
221
222     cs_d = gpio_open(8);
223     gpio_set_direction(cs_d, GPIO_OUT);
224     gpio_write(cs_d, 1);
225
226     // switch array control through shift register
227     // UPDATE THIS TO USE FUNCTIONS...
228     sr_data = gpio_open(2);
229     gpio_set_direction(sr_data, GPIO_OUT);
230     gpio_write(sr_data, 0);
231
232     sr_clk = gpio_open(4);
233     gpio_set_direction(sr_clk, GPIO_OUT);
234     gpio_write(sr_clk, 0);
235
236     sr_latch = gpio_open(3);
237     gpio_set_direction(sr_latch, GPIO_OUT);
238     gpio_write(sr_latch, 0);
239
240     sr_clr_n = gpio_open(5);
241     gpio_set_direction(sr_clr_n, GPIO_OUT);
242     gpio_write(sr_clr_n, 1);
243
244     // analog inputs for measuring voltage from multiplexers
245     // HOW DOES THIS WORK??
246     // need to investigate
https://github.com/Xilinx/PYNQ_Peripherals/blob/main/pynq_peripherals/modules/analog/src/analog.c
247     // 'device' is the pin
248     // and https://github.com/Xilinx/embeddedsw/blob/master/XilinxProcessorIPLib/drivers/sysmon/src/xsysmon.h
249
250
251     // trigger inputs - pins A4 (17) and A5 (18)
252     trig_in_1 = gpio_open(0);
253     gpio_set_direction(trig_in_1, GPIO_IN);
254
255     trig_in_2 = gpio_open(18);
256     gpio_set_direction(trig_in_2, GPIO_IN);
257
258     // set softspan range for all DACs
259     for(uint8_t i = 0; i < 20; i++){
260         dac_config(i, dac_ref);
261     }
262
263 }
264 /*
265 Start LED patterns generating based on alloacted array in DDR memory.
266 */
267 void leds_start(void* led_counts_buffer, void* trig, void* stop){
268
269     // NEED TO HAVE 2 INPUTS - ONE FOR EACH CONNECTED DEVICE
270     uint16_t (*led_counts)[20] = (uint16_t (*)[20])led_counts_buffer;
271
272     uint16_t *trig_ptr = (uint16_t *) trig;
273     *trig_ptr = 0;
274
275     uint16_t *stop_ptr = (uint16_t *) stop;
276
277     gpio_write(sr_clr_n, 0);
278     gpio_write(sr_clr_n, 1);
279
280     // set softspan for N track chips to switch to high Z
281     dac_write(&cs_c, LTC2662_CMD_SPAN, 0, 0);
282     dac_write(&cs_c, LTC2662_CMD_SPAN, 1, 0);
283     dac_write(&cs_c, LTC2662_CMD_SPAN, 2, 0);
284     dac_write(&cs_c, LTC2662_CMD_SPAN, 3, 0);
285     dac_write(&cs_c, LTC2662_CMD_SPAN, 4, 0);
286     dac_write(&cs_d, LTC2662_CMD_SPAN, 0, 0);
```

```
287     dac_write(&cs_d, LTC2662_CMD_SPAN, 1, 0);
288     dac_write(&cs_d, LTC2662_CMD_SPAN, 2, 0);
289     dac_write(&cs_d, LTC2662_CMD_SPAN, 3, 0);
290     dac_write(&cs_d, LTC2662_CMD_SPAN, 4, 0);
291
292     dac_write(&cs_c, LTC2662_CMD_UPDATE_ALL, 0, 0);
293     dac_write(&cs_d, LTC2662_CMD_UPDATE_ALL, 0, 0);
294
295     for(uint8_t i = 0; i < 20; i++){
296         gpio_write(sr_data, 1); // open switch
297         gpio_write(sr_clk, 0);
298         gpio_write(sr_latch, 0);
299         gpio_write(sr_clk, 1);
300         gpio_write(sr_latch, 1);
301         gpio_write(sr_clk, 0);
302         gpio_write(sr_latch, 0);
303     }
304
305     // For UOA
306     // loop forever
307     while(1){
308
309         if(*stop_ptr == 1){
310             // set channel currents to zero
311             for(uint8_t i = 0; i < 20; i++){
312                 dac_channel_control(i, 0, 0);
313             }
314             break;
315         }
316
317         *trig_ptr = gpio_read(trig_in_1);
318
319
320         // switch on single row for each matrix
321         // for some reason I don't understand one switch must always be closed otherwise they all close (need to
investigate with scope) - doesn't affect this implementation
322         for(uint8_t i = 0; i < 10; i++){
323             gpio_write(sr_data, 1); // open switch
324             gpio_write(sr_clk, 0);
325             gpio_write(sr_latch, 0);
326             gpio_write(sr_clk, 1);
327             gpio_write(sr_latch, 1);
328             gpio_write(sr_clk, 0);
329             gpio_write(sr_latch, 0);
330         }
331
332         gpio_write(sr_data, 0); // close switch
333
334         for(uint8_t sw = 0; sw < 10; sw++){
335
336             // set channel currents to zero
337             dac_write(&cs_a, LTC2662_CMD_WRITE_ALL_UPDATE_ALL, 0, 0);
338             dac_write(&cs_b, LTC2662_CMD_WRITE_ALL_UPDATE_ALL, 0, 0);
339             // use sr to switch to GND
340             // switch on single row for each matrix
341             // for some reason I don't understand one switch must always be closed otherwise they all close (need to
investigate with scope) - doesn't affect this implementation
342
343             gpio_write(sr_clk, 0);
344             gpio_write(sr_latch, 0);
345             gpio_write(sr_clk, 1);
346             gpio_write(sr_latch, 1);
347             gpio_write(sr_data, 1); // open switch
348
349             // set current DAC values
350             dac_write(&cs_a, LTC2662_CMD_WRITE_CODE_N, 0, led_counts[sw][0]);
351             dac_write(&cs_a, LTC2662_CMD_WRITE_CODE_N, 1, led_counts[sw][1]);
352             dac_write(&cs_a, LTC2662_CMD_WRITE_CODE_N, 2, led_counts[sw][2]);
353             dac_write(&cs_a, LTC2662_CMD_WRITE_CODE_N, 3, led_counts[sw][3]);
354             dac_write(&cs_a, LTC2662_CMD_WRITE_CODE_N, 4, led_counts[sw][4]);
355             dac_write(&cs_b, LTC2662_CMD_WRITE_CODE_N, 0, led_counts[sw][5]);
356             dac_write(&cs_b, LTC2662_CMD_WRITE_CODE_N, 1, led_counts[sw][6]);
357             dac_write(&cs_b, LTC2662_CMD_WRITE_CODE_N, 2, led_counts[sw][7]);
358             dac_write(&cs_b, LTC2662_CMD_WRITE_CODE_N, 3, led_counts[sw][8]);
359             dac_write(&cs_b, LTC2662_CMD_WRITE_CODE_N, 4, led_counts[sw][9]);
```

```
360             dac_write(&cs_a, LTC2662_CMD_UPDATE_ALL, 0, 0);
361             dac_write(&cs_b, LTC2662_CMD_UPDATE_ALL, 0, 0);
362 
363             delay_us(780);
364         }
365 
366         // set channel currents to zero
367         dac_write(&cs_a, LTC2662_CMD_WRITE_ALL_UPDATE_ALL, 0, 0);
368         dac_write(&cs_b, LTC2662_CMD_WRITE_ALL_UPDATE_ALL, 0, 0);
369 
370     }
371 
372     return;
373 }
374 
375 void leds_off(){
376     // set channel currents to zero
377     for(uint8_t i = 0; i < 20; i++){
378         dac_channel_control(i, 0, 0);
379     }
380 }
381 
382 /*
383 Continuous pattern driver.
384     'current_dac_counts' contains 16-bit values for each current channel (0-19)
385     'switches' least-signifcant 20 bits indicates if switch is closed or open
386 */
387 void leds_start_cont(unsigned short *current_dac_counts, unsigned int switches){
388 
389     // set channel currents to zero
390     for(uint8_t i = 0; i < 20; i++){
391         dac_channel_control(i, 0, 0);
392     }
393 
394     // this loop opens all the switches except the ones that should be closed
395     for(uint8_t sw_loop = 0; sw_loop < 20; sw_loop++){
396         // if((sw_loop % 20) == sw){
397         if((switches & (0x0001<<sw_loop))){
398             gpio_write(sr_data, 0); // close switch
399         }
400         else{
401             gpio_write(sr_data, 1); // open switch
402         }
403         gpio_write(sr_clk, 0);
404         gpio_write(sr_latch, 0);
405         gpio_write(sr_clk, 1);
406         gpio_write(sr_latch, 1);
407         gpio_write(sr_clk, 0);
408         gpio_write(sr_latch, 0);
409     }
410 
411     // switch current channels to relevant values
412     for(uint8_t i = 0; i < 20; i++){
413         dac_channel_control(i, 1, current_dac_counts[i]);
414     }
415 
416     // should return fault code for all 4 chips eventually
417 }
418 
419 /*
420 Connects output corresponding to 'channel' via mux to ADC
421 */
422 #define LTC2662_CMD_MUX 0xb0
423 void monitor_mux(uint8_t channel){
424     int8_t fault_reg = 0;
425     uint8_t selected_dac = channel % 5;
426     uint16_t mux_code = selected_dac + 24;
427 
428     fault_reg |= dac_write(&cs_a, LTC2662_CMD_MUX, 0, mux_code);
429     fault_reg |= dac_write(&cs_b, LTC2662_CMD_MUX, 0, mux_code);
430     fault_reg |= dac_write(&cs_c, LTC2662_CMD_MUX, 0, mux_code);
431     fault_reg |= dac_write(&cs_d, LTC2662_CMD_MUX, 0, mux_code);
432 }
433 
434 void read_adc(unsigned short *adc_data){
```

```
435
436     u32 xStatus;
437
438     // SysMon Initialize
439     SysMonConfigPtr = XSysMon_LookupConfig(SYSMON_DEVICE_ID);
440     if(SysMonConfigPtr == NULL)
441         xil_printf("SysMon LookupConfig failed.\n\r");
442     xStatus = XSysMon_CfgInitialize(SysMonInstPtr, SysMonConfigPtr,
443                                     SysMonConfigPtr->BaseAddress);
444     if(XST_SUCCESS != xStatus)
445         xil_printf("SysMon CfgInitialize failed\r\n");
446     // Clear the old status
447     XSysMon_GetStatus(SysMonInstPtr);
448
449     // Wait for the conversion complete
450     while ((XSysMon_GetStatus(SysMonInstPtr) &
451             XSM_SR_EOS_MASK) != XSM_SR_EOS_MASK);
452
453
454     // uint16_t adc_data;
455     adc_data[0] = XSysMon_GetAdcData(SysMonInstPtr, XSM_CH_AUX_MIN+1);  // A0
456     adc_data[1] = XSysMon_GetAdcData(SysMonInstPtr, XSM_CH_AUX_MIN+9);  // A1
457     adc_data[2] = XSysMon_GetAdcData(SysMonInstPtr, XSM_CH_AUX_MIN+6);  // A2
458     adc_data[3] = XSysMon_GetAdcData(SysMonInstPtr, XSM_CH_AUX_MIN+15); // A3
459
460     // return adc_data;
461 }
462
463
```

driver.py

```
1   import numpy as np
2   import os
3   import re
4   import pandas as pd
5   from matplotlib import pyplot as plt
6   import csv
7
8   import led_driver
9
10  from pynq.lib import MicroblazeRPC
11  from pynq import allocate
12
13  DAC_REF = 25
14
15  def led_driver_setup(ol):
16      """Programmes the microblaze with the LED driver capable of any pattern/LED brightness.
17          'ol' is base overlay instance
18      """
19      lib_path = os.path.dirname(led_driver.__file__)
20      c_src_path = os.path.join(lib_path, 'driver/driver.c')
21
22      with open(c_src_path, 'r') as f:
23          led_driver_source = f.read()
24
25      return MicroblazeRPC(ol.iop_arduino, led_driver_source)
26
27  def experiment_setup(ol):
28      dac_refs = {0 : 3.125,
29      1 : 6.25,
30      2 : 12.5,
31      3 : 25,
32      4 : 50,
33      5 : 100,
34      6 : 200,
35      7 : 300}
36      switch_max_current = 130
37      device_max_current = 2500
38
39      # led_driver_programme = led_driver_setup(ol)
40
41      lib_path = os.path.dirname(os.path.realpath(__file__))
42      driver_src_path = os.path.join(lib_path, 'driver/driver.c')
```

```
43
44      with open(driver_src_path, 'r') as f:
45          led_driver_source = f.read()
46
47      led_driver_programme = MicroblazeRPC(ol.iop_arduino, led_driver_source)
48
49      # HARD-CODED 25 MA LIMIT
50      led_driver_programme.leds_configure(3, switch_max_current, device_max_current)
51
52      return led_driver_programme
53
54  def quad_regression(led_irradiance, a, b, c):
55      return a*led_irradiance**2 + b*led_irradiance + c
56
57  def get_frames_counts(frames, device_id, ext_cables_used=True):
58      """Returns 3D array of size number of frames x 10 x 10 based on input data.
59          Output is calibrated based on irradiance to currents regression file and provides a 'counts' value for led
driver programme
60          Output is also re-orientated to correctly match indexing of LEDs
61      """
62      calibration_file = pd.read_csv(f'./Device calibration files/irr_to_current-{device_id}.dat', header=None)
63
64      # These files contain coefficients in order: P10N10, P9N10, ... P10N9, P9N9, ...
65      # Reverse so order is P1N1, P2N1, ... P1N2, P2N2, ...
66      calibration_data = np.array(calibration_file[::-1])
67
68      frames = np.swapaxes(np.swapaxes(frames,0,2),1,2)
69
70      # calibrate frames so correct current applied for each LED at each timestep - irr. already multiplied by 10 - AND
CAP LED_COUNTS AT 20mA
71      frames_currents = np.array([np.array([quad_regression(led_irradiance, *params) for led_irradiance, params in
zip(led_irradiances.reshape(-1), calibration_data)]).reshape(10,10) for led_irradiances in frames])
72      frames_currents[frames_currents > 20] = 20
73      frames_counts = (frames_currents / DAC_REF * 65535).astype(np.uint16)
74
75      # alternate rows and columns if extension cables are used
76      if ext_cables_used:
77          frames_counts_temp = np.copy(frames_counts[:,::2,:])
78          frames_counts[:,::2,:] = frames_counts[:,1::2,:]
79          frames_counts[:,1::2,:] = frames_counts_temp
80          frames_counts_temp = np.copy(frames_counts[:,:,::2])
81          frames_counts[:,:,::2] = frames_counts[:,:,1::2]
82          frames_counts[:,:,1::2] = frames_counts_temp
83
84      return frames_counts
85
86
87  # Device health scan
88  def get_currents_switches_cont_pattern(led_counts_cont_in):
89      """Returns array of 20 current count values and switch positions based on 'led_counts'.
90         'led_counts' should be pre-determined from LED currents.
91      """
92      led_counts_cont = np.zeros((20,20), dtype=np.uint16)
93      led_counts_cont[19:9:-1,:10] = led_counts_cont_in
94      switches = [int(any(row)) for row in led_counts_cont]
95      switches_value = np.sum([val*2**bit for bit,val in enumerate(switches)]).astype(np.uint32)
96
97      current_dac_counts = np.sum(led_counts_cont, axis=0)
98
99      return current_dac_counts, switches_value
100
101 LED_LABELS = [f'P{p}N{n}' for p, n in zip(np.tile(np.linspace(1,10,10).astype(int),10),
np.repeat(np.linspace(1,10,10).astype(int),10))]
102 # VOLTAGE_DIV = 0.1
103 VOLTAGE_DIV = 0.026 # Due to hardware issue with electronics at ADC input
104 CURRENTS_MA = np.linspace(0.01,10,50)
105 VOLTAGE_MIN_5MA = 2.5
106 VOLTAGE_MAX_5MA = 6.5
107
108 def measure_voltage(led_driver_programme, channel):
109     led_driver_programme.monitor_mux(channel)
110     voltages_measurements = allocate(shape=(4,), dtype=np.uint16)
111     led_driver_programme.read_adc(voltages_measurements)
112
113     return (voltages_measurements[np.r_[2,0,3,1]] / 65535 * 3.33 / VOLTAGE_DIV)
```

```
114
115 def device_health_scan(led_driver_programme, folder_name):
116     """Top level function to run voltage measurements under a sweep of currents across whole device.
117     """
118
119     voltage_measurements_all = []
120
121     channel = 0
122
123     for led_idx, led_label in enumerate(LED_LABELS):
124         voltage_measurements = []
125
126         for current in CURRENTS_MA:
127             led_counts_cont = ((np.array(LED_LABELS).reshape(10,-1) == led_label) *
np.uint16(current/25*65535)).astype(np.uint16)
128             led_driver_programme.leds_start_cont(*get_currents_switches_cont_pattern(led_counts_cont))
129             # sleep(0.001)
130             voltage_measurements.append(measure_voltage(led_driver_programme,channel)[int((led_idx%10)/5)])
131         channel = (channel+1)%5
132
133         voltage_measurements_all.append(voltage_measurements)
134
135         print(f'LED {led_label} ({led_idx+1}/100)...', end='\r')
136
137     led_driver_programme.leds_off()
138
139     # plot map of voltage where current == 5 mA
140     voltage_measurements_all_array = np.array(voltage_measurements_all).reshape(100,-1)
141
142     voltage_at_approx_5mA = np.zeros((100),)
143     for led_idx, led_label in enumerate(LED_LABELS):
144         voltage_at_approx_5mA[led_idx] = voltage_measurements_all_array[led_idx][np.argmin(np.abs(CURRENTS_MA-5))]
145
146     # characterise performance
147     led_performance_map = []
148     for led_idx, led_label in enumerate(LED_LABELS):
149         if voltage_measurements_all_array[led_idx][np.argmin(np.abs(CURRENTS_MA-5))] < VOLTAGE_MIN_5MA:
150             led_performance_map.append([0,0,255]) # short
151         elif voltage_measurements_all_array[led_idx][np.argmin(np.abs(CURRENTS_MA-5))] > 8:
152             led_performance_map.append([255,0,0]) # open
153         elif (voltage_measurements_all_array[led_idx][np.argmin(np.abs(CURRENTS_MA-5))] > VOLTAGE_MIN_5MA) &
(voltage_measurements_all_array[led_idx][np.argmin(np.abs(CURRENTS_MA-5))] < VOLTAGE_MAX_5MA):
154             led_performance_map.append([0,255,0]) # good
155         else:
156             led_performance_map.append([255,128,128]) #bad
157
158     newpath = f'../health_scans/{folder_name}'
159     if not os.path.exists(newpath):
160         os.makedirs(newpath)
161
162     # save both maps
163     print('LED performance')
164     plt.imshow(np.array(led_performance_map).astype(np.uint8).reshape(10,10,3))
165     plt.savefig(f'../health_scans/{folder_name}/led_performance_map.png')
166
167     print('LED voltage at 5 mA')
168     plt.imshow((np.array(voltage_at_approx_5mA)).astype(np.uint8).reshape(10,10), vmin=0, vmax=12)
169     plt.colorbar()
170     plt.savefig(f'../health_scans/{folder_name}/voltage_at_5mA.png')
171
172     # save voltage measurements as CSV
173     with open(f'../health_scans/{folder_name}/voltage_measurements.csv', 'w', newline='\n') as f:
174         csv_writer = csv.writer(f, delimiter=',')
175         csv_writer.writerow(['Current (mA)'])
176         csv_writer.writerow(CURRENTS_MA)
177         csv_writer.writerow(['Voltage (V)'])
178         for led_idx, (led_label, voltage_measurements) in enumerate(zip(LED_LABELS, voltage_measurements_all)):
179             csv_writer.writerow([led_label])
180             csv_writer.writerow(voltage_measurements_all[led_idx])
181
182
183
```

C.4 Control system Jupyter Notebook demos

LED driver example code

Ryan Greer, 26th July 2025

- Uses newer simplified API

Run on startup

In [3]:
```
import numpy as np
import plotly.express as px
from time import sleep
from pynq import Overlay
from pynq import allocate
from led_driver.driver import experiment_setup
from led_driver.driver import get_frames_counts
from led_driver.driver import device_health_scan

base = Overlay("base.bit")
```

In [4]:
```
%%capture

led_driver_programme = experiment_setup(base)
led_counts = allocate(shape=(20,20), dtype=np.uint16)
trig = allocate(shape=(1,), dtype=np.uint16)
stop = allocate(shape=(1,), dtype=np.uint16)
stop[0] = 0
led_counts.fill(0)
led_driver_programme.leds_start(led_counts, trig, stop)
```

Test LEDs

In [ ]:
```
led_counts[:10,:10][4,5] = 10000 # switch on LED at P5N6
```

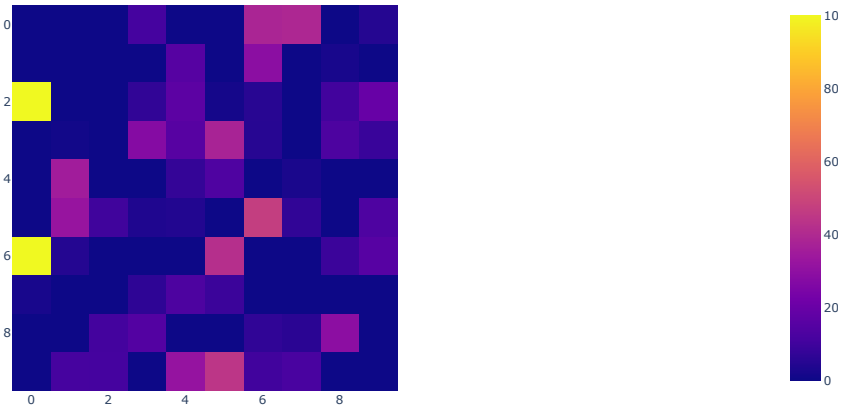
In [ ]:
```
led_counts.fill(0)
```

Load experimental protocol

In [ ]:
```
# multiply cortical irradiance by 10 to account for driver duty cycle
GO_PATTERN = np.load('./timearray_12kHz.npy')*10
NOGO_PATTERN = np.load('./timearray_4kHz.npy')*10
```

In [3]:
```
import plotly.express as px
import numpy as np
px.imshow(np.random.randn(10,10))
```

In [6]:
```
px.imshow(GO_PATTERN[:,:,11])
```



Run experiment

In [7]:
```
TRIALS = ['go','go','nogo','go','nogo']
```

In [11]:
```
for trial_idx, TRIAL in enumerate(TRIALS):
    print(f'TRIAL: {str(trial_idx+1).zfill(3)}' , end='\r')

    # Set up the LED values and wait for trigger
    if TRIAL == 'go':
        frames_counts = get_frames_counts(GO_PATTERN, 'H4-059', ext_cables_used=False)
    else:
        frames_counts = get_frames_counts(NOGO_PATTERN, 'H4-059', ext_cables_used=False)

    # wait for trigger then start pattern sequence
    while(trig[0] == 0):
        pass

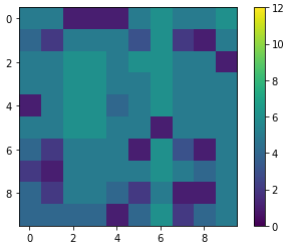
    # Run through frame sequence
    for frame_counts in frames_counts:
        led_counts[:10,:10] = frame_counts
        sleep(0.025) # wait for 25 ms
    led_counts[:10,:10] = 0
```

```
TRIAL: 005
```

Device health scan

In [10]:
```
# DRIVER MUST BE STOPPED FOR THIS
stop[0] = 1
device_health_scan(led_driver_programme, '10-12-2025 DEVICE 4')
```

```
LED performance/100)...
LED voltage at 5 mA
```



Stop experiment

In [18]:
```
# RUN THIS TO STOP THE DRIVER
stop[0] = 1
```

In [ ]:
```
# RUN THIS TO RESTART THE DRIVER
stop[0] = 0
led_driver_programme.leds_start(led_counts, trig, stop)
```

In [ ]:
```
```

Electrophysiology recording

Import libraries

In [1]:
```
from pynq import Overlay, allocate
import numpy as np
from pynq import MMIO
from time import sleep
import ipywidgets as widgets
import plotly.express as px
import plotly.graph_objects as go
from plotly.subplots import make_subplots
import asyncio
import time
```

Programme FPGA, configure FPGA and send initialisation to RHD

In [2]:
```
# Write registers to configure RHD2000
ol = Overlay("/home/xilinx/jupyter_notebooks/HearLight/electrophysiology/rhd_data_streaming.bit")

IP_BASE_ADDRESS = 0x40000000
ADDRESS_RANGE = 0x0FFF

# 1 kHz - 10 ms per packet (2 datastreams)
PACKET_SIZE = 3520

mmio = MMIO(IP_BASE_ADDRESS, ADDRESS_RANGE)

addresses_to_write = [0b000001100, # ep03wirein - set sample rate
                      0b100000000, # ep40trigin - trigger MMCM reprogramming
                      0b100000000, # ep40trigin - trigger MMCM reprogramming
                      0b001010000, # ep14wirein - data streams enable
                      0b100000100, # ep41trigin - bit 0 (SPI start)
                      0b000000000, # ep00wirein - bit 0 (reset), bit 1 (run continuous), bit 2 (dsp settle)
                      0b000000000, # ep00wirein - bit 0 (reset), bit 1 (run continuous), bit 2 (dsp settle)
                      0b001010100, # ep15wirein - mAXIS FIFO packet size
                      0b100000100] # ep41trigin - bit 0 (SPI start)

data_to_write = [0x0000077D, # 1 kHz
                 0x00000001, # enable MMCM reprogramming
                 0x00000000, # disable MMCM reprogramming
                 0x00000003, # enable first two data streams
                 0x00000000, # SPI stop
                 0x00000003, # reset and SPI run continuous
                 0x00000002, # disable reset, keep SPI run continuous
                 PACKET_SIZE,
                 0x00000001] # SPI start

for address, data in zip(addresses_to_write, data_to_write):
    mmio.write(address, data)
    sleep(0.1)

COMMANDS = {'READ(R)' : '11RRRRRR00000000',
            'WRITE(R,D)' : '10RRRRRRDDDDDDDD',
            'CALIBRATE' : '0101010100000000'}

COMMAND_SEQUENCE = [COMMANDS['READ(R)'].replace('RRRRRR', format(63,'06b')),
                    COMMANDS['READ(R)'].replace('RRRRRR', format(63,'06b')),
                    COMMANDS['WRITE(R,D)'].replace('RRRRRR', format(0,'06b')).replace('DDDDDDDD', format(int('0xDE',16),'08b')),
                    COMMANDS['WRITE(R,D)'].replace('RRRRRR', format(1,'06b')).replace('DDDDDDDD', format(int('0x42',16),'08b')),
                    COMMANDS['WRITE(R,D)'].replace('RRRRRR', format(2,'06b')).replace('DDDDDDDD', format(int('0x04',16),'08b')),
                    COMMANDS['WRITE(R,D)'].replace('RRRRRR', format(3,'06b')).replace('DDDDDDDD', format(int('0x00',16),'08b')),
                    COMMANDS['WRITE(R,D)'].replace('RRRRRR', format(4,'06b')).replace('DDDDDDDD', format(int('0x80',16),'08b')),
                    COMMANDS['WRITE(R,D)'].replace('RRRRRR', format(5,'06b')).replace('DDDDDDDD', format(int('0x00',16),'08b')),
                    COMMANDS['WRITE(R,D)'].replace('RRRRRR', format(6,'06b')).replace('DDDDDDDD', format(int('0x80',16),'08b')),
                    COMMANDS['WRITE(R,D)'].replace('RRRRRR', format(7,'06b')).replace('DDDDDDDD', format(int('0x00',16),'08b')),
                    COMMANDS['WRITE(R,D)'].replace('RRRRRR', format(8,'06b')).replace('DDDDDDDD', format(int('0x16',16),'08b')),
                    COMMANDS['WRITE(R,D)'].replace('RRRRRR', format(9,'06b')).replace('DDDDDDDD', format(int('0x80',16),'08b')),
                    COMMANDS['WRITE(R,D)'].replace('RRRRRR', format(10,'06b')).replace('DDDDDDDD', format(int('0x17',16),'08b')),
                    COMMANDS['WRITE(R,D)'].replace('RRRRRR', format(11,'06b')).replace('DDDDDDDD', format(int('0x80',16),'08b')),
                    COMMANDS['WRITE(R,D)'].replace('RRRRRR', format(12,'06b')).replace('DDDDDDDD', format(int('0x2C',16),'08b')),
                    COMMANDS['WRITE(R,D)'].replace('RRRRRR', format(13,'06b')).replace('DDDDDDDD', format(int('0x86',16),'08b')),
                    COMMANDS['WRITE(R,D)'].replace('RRRRRR', format(14,'06b')).replace('DDDDDDDD', format(int('0xFF',16),'08b')),
                    COMMANDS['WRITE(R,D)'].replace('RRRRRR', format(15,'06b')).replace('DDDDDDDD', format(int('0xFF',16),'08b')),
                    COMMANDS['WRITE(R,D)'].replace('RRRRRR', format(16,'06b')).replace('DDDDDDDD', format(int('0xFF',16),'08b')),
                    COMMANDS['WRITE(R,D)'].replace('RRRRRR', format(17,'06b')).replace('DDDDDDDD', format(int('0xFF',16),'08b')),
                    COMMANDS['WRITE(R,D)'].replace('RRRRRR', format(18,'06b')).replace('DDDDDDDD', format(int('0xFF',16),'08b')),
                    COMMANDS['WRITE(R,D)'].replace('RRRRRR', format(19,'06b')).replace('DDDDDDDD', format(int('0xFF',16),'08b')),
                    COMMANDS['WRITE(R,D)'].replace('RRRRRR', format(20,'06b')).replace('DDDDDDDD', format(int('0xFF',16),'08b')),
                    COMMANDS['WRITE(R,D)'].replace('RRRRRR', format(21,'06b')).replace('DDDDDDDD', format(int('0xFF',16),'08b')),
                    COMMANDS['CALIBRATE'],
                    COMMANDS['READ(R)'].replace('RRRRRR', format(63,'06b')),
                    COMMANDS['READ(R)'].replace('RRRRRR', format(63,'06b')),
                    COMMANDS['READ(R)'].replace('RRRRRR', format(63,'06b')),
                    COMMANDS['READ(R)'].replace('RRRRRR', format(63,'06b')),
                    COMMANDS['READ(R)'].replace('RRRRRR', format(63,'06b')),
                    COMMANDS['READ(R)'].replace('RRRRRR', format(63,'06b')),
                    COMMANDS['READ(R)'].replace('RRRRRR', format(63,'06b')),
                    COMMANDS['READ(R)'].replace('RRRRRR', format(63,'06b')),
                    COMMANDS['READ(R)'].replace('RRRRRR', format(63,'06b'))]

# add surplus read commands (each command sequence should have length 60 - for some reason it needs to be 61)
for i in range(61-34):
    COMMAND_SEQUENCE.append(COMMANDS['READ(R)'].replace('RRRRRR', format(63,'06b')))

aux_command_sequence = np.array([int(command,2) for command in COMMAND_SEQUENCE]).astype(np.uint16)
aux_command_sequence_buffer = allocate(shape=(61,), dtype=np.uint16)
aux_command_sequence_buffer[:] = aux_command_sequence[:]
dma_aux_cmd = ol.axi_dma_aux
dma_aux_cmd.sendchannel.transfer(aux_command_sequence_buffer)
dma_aux_cmd.sendchannel.wait()
```

Recording and closed-loop demo

In [19]:
```
dma = ol.axi_dma_stream
```

In [20]:
```
SUBSAMPLING = 2
N_PACKETS_IN_BUFFER = 1000
```

```
DRAM_BUFFER_SIZE = PACKET_SIZE*N_PACKETS_IN_BUFFER # 16-bit values - must be integer multiple of PACKET_SIZE

TIME_TO_SHOW = 2 # seconds
TARGET_PLOT_SAMPLES = FS*TIME_TO_SHOW # not subsampled
N_DATA_STREAMS_ENABLED = 2
LENGTH_BLOCK = 35*N_DATA_STREAMS_ENABLED + 16 + N_DATA_STREAMS_ENABLED % 4

N_SAMPLES_PACKET = int(PACKET_SIZE / LENGTH_BLOCK)
N_PACKETS_PLOT = int(np.ceil(TARGET_PLOT_SAMPLES / N_SAMPLES_PACKET))
PLOT_SAMPLES = N_PACKETS_PLOT * N_SAMPLES_PACKET

MAGIC_NUMBER = np.uint64(0xD7A22AAA38132A53)
```

In [22]:
```
UPDATE_PLOT = 20 # every packet reads
```

In [90]:
```
TIME_TO_RUN = 300 # seconds
PACKET_INDEX_SHOW = 28

class rhd_demo():
    def __init__(self):
        self._stop = False
        # self.queue = asyncio.Queue()

        self.num_words_in_fifo_meter = widgets.IntProgress(min=0, max=131071)

        plot_x = np.linspace(0,PLOT_SAMPLES-1,PLOT_SAMPLES) # x in 'samples' currently
        plot_y = np.zeros(plot_x.shape)
        self.fig = go.FigureWidget(go.Scattergl(x=plot_x[::SUBSAMPLING],
                                    y=plot_y[::SUBSAMPLING],
                                    mode='lines'))
        self.fig.update_layout(xaxis_title='Sample index')
        self.fig.update_layout(yaxis_title='ADC counts')
        self.figs = []
        for i in range(4):
            self.figs.append(go.FigureWidget(go.Scatter(x=plot_x[::SUBSAMPLING],
                                y=plot_y[::SUBSAMPLING],
                                mode='lines')))#,
                                #hoverinfo="skip")))
            self.figs[i].update_layout(xaxis_title='Sample index')
            self.figs[i].update_layout(yaxis_title='ADC counts')

        self.all_plots = go.FigureWidget(make_subplots(rows=2, cols=2))
        rows = [1,1,2,2]
        cols = [1,2,1,2]

        # Get the Express fig broken down as traces and add the traces to the proper plot within in the subplot
        for traces, row, col in zip(self.figs, rows, cols):
            self.all_plots.append_trace(traces['data'][0], row=row, col=col)

        self.rhd_data_buffer = allocate((N_PACKETS_IN_BUFFER, PACKET_SIZE), dtype=np.int16) # buffer for multiple data packets
        self.rhd_data = allocate((PACKET_SIZE,), dtype=np.int16) # allocated memory for single data packet
        self.num_words_in_fifo_all = []

        self.packet_index_show = PACKET_INDEX_SHOW

    def get_data_packet(self):
        dma.recvchannel.wait()
        dma.recvchannel.transfer(self.rhd_data)

    # Sliding window buffer update
    def update_buffer(self, index):
        self.rhd_data_buffer[index,:] = np.array(self.rhd_data)
        index = (index + 1) % N_PACKETS_IN_BUFFER  # Circular index
        return index

    def get_data_packet_update_buffer(self, index):
        dma.recvchannel.wait()
        dma.recvchannel.transfer(self.rhd_data_buffer[index,:])
        index = (index + 1) % N_PACKETS_IN_BUFFER  # Circular index
        return index

    def get_datastream_index_first(self, rhd_data):
        if np.argwhere(rhd_data[:LENGTH_BLOCK].view(np.uint64) == MAGIC_NUMBER).shape[0]:
            datastream_index_first = np.argwhere(rhd_data[:LENGTH_BLOCK].view(np.uint64) == MAGIC_NUMBER)[0][0]*4
        elif np.argwhere(rhd_data[1:LENGTH_BLOCK+1].view(np.uint64) == MAGIC_NUMBER).shape[0]:
            datastream_index_first = np.argwhere(rhd_data[1:LENGTH_BLOCK+1].view(np.uint64) == MAGIC_NUMBER)[0][0]*4+1
        elif np.argwhere(rhd_data[2:LENGTH_BLOCK+2].view(np.uint64) == MAGIC_NUMBER).shape[0]:
            datastream_index_first = np.argwhere(rhd_data[2:LENGTH_BLOCK+2].view(np.uint64) == MAGIC_NUMBER)[0][0]*4+2
        elif np.argwhere(rhd_data[3:LENGTH_BLOCK+3].view(np.uint64) == MAGIC_NUMBER).shape[0]:
            datastream_index_first = np.argwhere(rhd_data[3:LENGTH_BLOCK+3].view(np.uint64) == MAGIC_NUMBER)[0][0]*4+3
        else:
            datastream_index_first = -1 # it will fix on the next cycle
            # time.sleep(0.01)
            print('ERROR: could not find magic number in packet')

        return datastream_index_first

    async def save_data(self, data, buffer_save_idx):
        np.save(f'./EXP1.3_2/rhd_data_{buffer_save_idx}.npy', data)

    def stop(self):
        self._stop = True

    def read_rhd_data(self):
        write_index = 0

        datastream_indices_plot = np.zeros((N_PACKETS_PLOT,N_SAMPLES_PACKET), dtype=int)
        channel_data_indices_plot = np.zeros((N_PACKETS_PLOT,N_SAMPLES_PACKET), dtype=int)

        buffer_save_idx = 0
        seconds_per_packet = PACKET_SIZE / LENGTH_BLOCK * 1/FS
        packets_to_save = int(0.1 / seconds_per_packet) # number of packets to save (each save is 1 second)

        update_plot_idx = 0
        multi_plot_idx = 0

        time_counter = 0

        rolling_mean = np.zeros((10,))

        dma.recvchannel.transfer(self.rhd_data)
        datastream_index_first = self.get_datastream_index_first(self.rhd_data)
        datastream_indices = np.r_[datastream_index_first:PACKET_SIZE:LENGTH_BLOCK]
        if (datastream_indices[-1]+self.packet_index_show) > (PACKET_SIZE-1):
            # if the last channel index is off the end of the packet
            channel_data_indices = np.r_[datastream_index_first+self.packet_index_show-LENGTH_BLOCK:PACKET_SIZE:LENGTH_BLOCK]
        else:
            channel_data_indices = datastream_indices+self.packet_index_show

        task = None
```

```
        self.channel_data = []
        self.timestamp_low_data = []
        self.timestamp_high_data = []
        n_packets_mean = 100
        self.rolling_mean = np.zeros((n_packets_mean,))

        self.led_pulse_packet_indices = []

        packet_counter = -1
        start_time = time.monotonic()

        # wait until FIFO empties so all tests start similar
        while(mmio.read(0b010000000) > 100):
            pass

        while True:
            if self._stop:
                self._stop = False
                break

            # update FIFOmeter
            while(mmio.read(0b010000000) < 100):
                pass
            self.num_words_in_fifo_all.append(mmio.read(0b010000000))

            write_index = self.get_data_packet_update_buffer(write_index)

            # continue
            if write_index%100 == 0: # every 100 ms
                datastream_index_first = self.get_datastream_index_first(self.rhd_data_buffer[(write_index-1)%N_PACKETS_IN_BUFFER,:])
                datastream_indices = np.r_[datastream_index_first:PACKET_SIZE:LENGTH_BLOCK]

                # save channel data and timestamps into array for plotting later
                if (datastream_indices[-1]+self.packet_index_show) > (PACKET_SIZE-1):
                    # if the last channel index is off the end of the packet
                    channel_data_indices = np.r_[datastream_index_first+self.packet_index_show-LENGTH_BLOCK:PACKET_SIZE:LENGTH_BLOCK]
                else:
                    channel_data_indices = datastream_indices+self.packet_index_show

                if (datastream_indices[-1]+5) > (PACKET_SIZE-1):
                    timestamp_low_indices =  np.r_[datastream_index_first+5-LENGTH_BLOCK:PACKET_SIZE:LENGTH_BLOCK]
                else:
                    timestamp_low_indices = datastream_indices+5
                if (datastream_indices[-1]+4) > (PACKET_SIZE-1):
                    timestamp_high_indices =  np.r_[datastream_index_first+4-LENGTH_BLOCK:PACKET_SIZE:LENGTH_BLOCK]
                else:
                    timestamp_high_indices = datastream_indices+4

                # copy n number of most recent packets
                end_index = N_PACKETS_IN_BUFFER if (write_index == 0) else write_index
                self.channel_data.append(np.array(self.rhd_data_buffer[end_index-100:end_index,channel_data_indices]).copy())
                self.timestamp_low_data.append(np.array(self.rhd_data_buffer[end_index-100:end_index,timestamp_low_indices]).copy())
                self.timestamp_high_data.append(np.array(self.rhd_data_buffer[end_index-100:end_index,timestamp_high_indices]).copy())

            # check if any sample in current packet above threshold - actviate LEDs
            if (time.monotonic()-start_time) > 5:
                  if (self.rhd_data_buffer[(write_index-1)%N_PACKETS_IN_BUFFER,:][channel_data_indices][0] > 19000):
                    self.led_pulse_packet_indices.append(packet_counter)
                    led_counts[:10,10:] = 4096
                    start_time = time.monotonic()  # Record when we start saving
            elif (time.monotonic()-start_time) > 0.5:
                led_counts.fill(0)

            packet_counter += 1

            PLOT = True
            if PLOT:
                channel_data_indices_plot[:-1,:] = channel_data_indices_plot[1:,:]
                channel_data_indices_plot[-1,:] = channel_data_indices[:N_SAMPLES_PACKET]

                if update_plot_idx%UPDATE_PLOT == 0:
                    channel_data_indices_plot_flattened = np.array([channel_data_indices_plot_packet+PACKET_SIZE*packet_index for packet_index, channel_data_indices_plot_packet i

                    # update single plot
                    self.fig.data[0].update({'y' : np.roll(self.rhd_data_buffer, -write_index, axis=0)[-N_PACKETS_PLOT:,:].reshape(-1)[channel_data_indices_plot_flattened[:]][::S

                    self.num_words_in_fifo_meter.value = mmio.read(0b010000000)

                update_plot_idx += 1

            # save rhd_data_buffer into numpy file every 'packets_to_save' packets
            SAVE = False
            if SAVE:
                if (write_index % packets_to_save) == 0:
                    end_index = N_PACKETS_IN_BUFFER if (write_index == 0) else write_index
                    try:
                        pass
                    except:
                        pass
                    task = asyncio.ensure_future(self.save_data(self.rhd_data_buffer[end_index-packets_to_save:end_index,:], buffer_save_idx))
                    buffer_save_idx += 1
```

In [92]:
```
# del rhd_demo_inst.rhd_data_buffer
```

In [93]:
```
rhd_demo_inst = rhd_demo()
```

In [94]:
```
rhd_demo_inst.num_words_in_fifo_meter
```

```
IntProgress(value=0, max=131071)
```

In [95]:
```
rhd_demo_inst.fig.update_layout(yaxis_range=(-10000,30000))
rhd_demo_inst.fig
```

```
FigureWidget({
    'data': [{'mode': 'lines',
              'type': 'scattergl',
              'uid': 'e0c0959…
```

In [96]:
```
rhd_demo_inst.packet_index_show = 28
rhd_demo_inst.read_rhd_data()
```

```
ERROR: could not find magic number in packet
KeyboardInterrupt
```

In [ ]:
```

```

Statement on previously published material

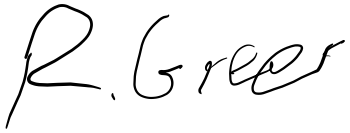
This thesis incorporates material previously published in:

Greer, R., Verdier, A., Butt, E. *et al*. Chronically implantable μLED arrays for optogenetic cortical surface stimulation in mice. *Nat Commun* **17**, 878 (2026). https://doi.org/10.1038/s41467-025-68191-5

I was the principal contributor to this publication, which represents a significant proportion of the work undertaken as part of my PhD research. I led the writing of the manuscript and carried out the majority of the analyses. Specifically, I prepared five of the six main figures, and fourteen of the seventeen supplementary figures. A few figures are presented in the thesis that I did not prepare, and these are highlighted in the Acknowledgements, and relevant sections.

The publication was a collaborative effort and all co-authors contributed to the work as detailed in the Author Contributions statement in the manuscript. The material incorporated into this thesis is derived from my contribution and has been integrated into the broader context of the thesis.

Signed: Ryan Greer



10/06/2026