Millikelvin Confocal Microscopy of Semiconductor Membranes and Filter Functions for Unital Quantum Operations

Der Fakultät für Mathematik, Informatik und Naturwissenschaften der RWTH Aachen University vorgelegte Dissertation zur Erlangung des akademischen Grades eines Doktors der Naturwissenschaften

von

Tobias Hangleiter, M.Sc.

aus

Filderstadt

JARA-Institute for Quantum Information

an initiative of

RWTH Aachen University
Quantum Technology Group
Department of Physics
Otto-Blumenthal-Straße
52070 Aachen

Forschungszentrum Jülich GmbH
JARA-Institute for Quantum Information
Peter-Grünberg-Institute (PGI-11)
Wilhelm-Johnen-Straße
52428 Jülich

Tobias Hangleiter
Millikelvin Confocal Microscopy of Semiconductor Membranes and Filter Functions for Unital Quantum Operations
September 10, 2025

Summary

Zusammenfassung

Acknowledgements

Tobias Hangleiter
Aachen, September 2025

Preface

It's been a journey.

Tobias Hangleiter
Aachen, September 2025

How to read this thesis

Contents

Publications

[1] Yaiza Aragonés-Soria, René Otten, Tobias Hangleiter, Pascal Cerfontaine, and David Gross. "Minimising Statistical Errors in Calibration of Quantum-Gate Sets." June 7, 2022. DOI: 10.48550/arXiv.2206.03417. arXiv: 2206.03417 [quant-ph]. Prepublished.

[2] Pascal Cerfontaine, Tobias Hangleiter, and Hendrik Bluhm. "Filter Functions for Quantum Processes under Correlated Noise." In: *Physical Review Letters* 127.17 (Oct. 18, 2021), p. 170403. DOI: 10.1103/PhysRevLett.127.170403.

[3] Thomas Descamps, Feng Liu, Sebastian Kindel, René Otten, Tobias Hangleiter, Chao Zhao, Mihail Ion Lepsa, Julian Ritzmann, Arne Ludwig, Andreas D. Wieck, Beata E. Kardynał, and Hendrik Bluhm. "Semiconductor Membranes for Electrostatic Exciton Trapping in Optically Addressable Quantum Transport Devices." In: *Physical Review Applied* 19.4 (Apr. 28, 2023), p. 044095. DOI: 10.1103/PhysRevApplied.19.044095.

[4] Thomas Descamps, Feng Liu, Tobias Hangleiter, Sebastian Kindel, Beata E. Kardynał, and Hendrik Bluhm. "Millikelvin Confocal Microscope with Free-Space Access and High-Frequency Electrical Control." In: *Review of Scientific Instruments* 95.8 (Aug. 9, 2024), p. 083706. DOI: 10.1063/5.0200889.

[5] Denny Dütz, Sebastian Kock, Tobias Hangleiter, and Hendrik Bluhm. "Distributed Bragg Reflectors for Thermal Isolation of Semiconductor Spin Qubits." In preparation.

[6] Sarah Fleitmann, Fabian Hader, Jan Vogelbruch, Simon Humpohl, Tobias Hangleiter, Stefanie Meyer, and Stefan van Waasen. "Noise Reduction Methods for Charge Stability Diagrams of Double Quantum Dots." In: *IEEE Transactions on Quantum Engineering* 3 (2022), pp. 1–19. DOI: 10.1109/TQE.2022.3165968.

[7] Fabian Hader, Jan Vogelbruch, Simon Humpohl, Tobias Hangleiter, Chimezie Eguzo, Stefan Heinen, Stefanie Meyer, and Stefan van Waasen. "On Noise-Sensitive Automatic Tuning of Gate-Defined Sensor Dots." In: *IEEE Transactions on Quantum Engineering* 4 (2023), pp. 1–18. DOI: 10.1109/TQE.2023.3255743.

[8] Tobias Hangleiter, Pascal Cerfontaine, and Hendrik Bluhm. "Filter-Function Formalism and Software Package to Compute Quantum Processes of Gate Sequences for Classical Non-Markovian Noise." In: *Physical Review Research* 3.4 (Oct. 18, 2021), p. 043047. DOI: 10.1103/PhysRevResearch.3.043047.

[9] Tobias Hangleiter, Pascal Cerfontaine, and Hendrik Bluhm. "Erratum: Filter-function Formalism and Software Package to Compute Quantum Processes of Gate Sequences for Classical Non-Markovian Noise [Phys. Rev. Research 3, 043047 (2021)]." In: *Physical Review Research* 6.4 (Oct. 16, 2024), p. 049001. DOI: 10.1103/PhysRevResearch.6.049001.

[10] Isabel Nha Minh Le, Julian D. Teske, Tobias Hangleiter, Pascal Cerfontaine, and Hendrik Bluhm. "Analytic Filter-Function Derivatives for Quantum Optimal Control." In: *Physical Review Applied* 17.2 (Feb. 2, 2022), p. 024006. DOI: 10.1103/PhysRevApplied.17.024006.

[11] Paul Surrey, Julian D. Teske, Tobias Hangleiter, Pascal Cerfontaine, and Hendrik Bluhm. "Data-Driven Qubit Characterization and Optimal Control Using Deep Learning." In preparation.

[12] Kui Wu, Sebastian Kindel, Thomas Descamps, Tobias Hangleiter, Jan Christoph Müller, Rebecca Rodrigo, Florian Merget, Beata E. Kardynal, Hendrik Bluhm, and Jeremy Witzens. "Modeling an Efficient Singlet-Triplet-Spin-Qubit-to-Photon Interface Assisted by a Photonic Crystal Cavity." In: *Physical Review Applied* 21.5 (May 24, 2024), p. 054052. DOI: 10.1103/PhysRevApplied.21.054052.

[13] Kui Wu, Sebastian Kindel, Thomas Descamps, Tobias Hangleiter, Jan Christoph Müller, Rebecca Rodrigo, Florian Merget, Hendrik Bluhm, and Jeremy Witzens. "An Efficient Singlet-Triplet Spin Qubit to Fiber Interface Assisted by a Photonic Crystal Cavity." In: *The 25th European Conference on Integrated Optics*. Ed. by Jeremy Witzens, Joyce Poon, Lars Zimmermann, and Wolfgang Freude. Cham: Springer Nature Switzerland, 2024, pp. 365–372. DOI: 10.1007/978-3-031-63378-2_60.

Software

The following open-source software packages were developed (at least partially) during the work on this thesis.

[1] Tobias Hangleiter, Isabel Nha Minh Le, and Julian D. Teske, *Filter_functions* version v1.1.3, May 14, 2024. Zenodo. DOI: 10.5281/ZENODO.4575000,

[2] Tobias Hangleiter, *Lindblad_mc_tools* Aug. 8, 2025. URL: https://git.rwth-aachen.de/tobias.hangleiter/lindblad_mc_tools.

[3] Tobias Hangleiter, *Mjolnir* Aug. 8, 2025. URL: https://git-ce.rwth-aachen.de/qutech/python-mjolnir.

[4] Tobias Hangleiter, Simon Humpohl, Max Beer, Paul Surrey, and René Otten, *Python-Spectrometer* version 2025.9.1, Sept. 8, 2025. Zenodo. DOI: 10.5281/zenodo.17079009,

[5] Tobias Hangleiter, Simon Humpohl, Paul Surrey, and Han Na We, *Qutil* version 2025.8.1, Aug. 12, 2025. Zenodo. DOI: 10.5281/zenodo.16816394,

Part I

A Flexible Python Tool For Fourier-Transform Noise Spectroscopy

Part II

Characterization and Improvements of a Millikelvin Confocal Microscope

Part III

Optical Measurements of Electrostatic Exciton Traps in Semiconductor Membranes

The `mjolnir` measurement framework

6

To facilitate the optical experiments in this part of the present thesis, I wrote a software package that controls the measurement workflow and all other interactions with the setup described in Part II, `mjolnir`[1] [1]. The package is written in Python, is open source, and has a documentation as well as some rudimentary tests.[2] In this chapter, I lay out the goals and non-goals behind its development and briefly describe the design, features, as well as the typical workflow.

1: https://git.rwth-aachen.de/qutech/mjolnir

2: Testing code interacting with hardware is notoriously difficult to achieve in an automated manner.

6.1 Rationale

There exist various software solutions for measurement control, ranging from commercial (*e.g.*, Keysight Labber[3]) over open source (`quantify`,[4] `labcore`[5]) to home-built (`elicit`,[6] `QuMADA`[7]). While most of these frameworks attempt to achieve some degree of universality and not be tailored towards a specific type of experiment or setup, some bias always exists and most are geared towards fully electrical experiments. On the other hand, software for optical experiments naturally exists as well, but it often suffers the same shortcomings coming from a different direction, and there is fairly little on offer for experiments at the interface of quantum optics and transport. An added complication is the availability of drivers. Striving for full automation of the setup, a number of different drivers are required given the complexity of the setup and the mix of electrical, optical, and mechanical instruments (Part II). Thus, `QCoDeS` forms a good basis for our specific needs due to its large driver coverage. However, the wide array of physical instruments demands some level of abstraction to promote reproducibility and ease-of-use. Furthermore, while `QCoDeS` provides measurement functionality, their definition involves a large amount of repetitive setup code that needs to be duplicated for each measurement, invites errors, and degrades legibility. The `doNd`[8] functionality abstracts some of this away for multi-dimensional loops but is not flexible enough for our purposes.

3: https://www.keysight.com/de/de/products/software/application-sw/labber-software.html

4: https://quantify-os.org/

5: https://github.com/toolsforexperiments/labcore

6: https://git-ce.rwth-aachen.de/qutech/frameworks/qool-tool

7: https://github.com/qutech/qumada

8: https://microsoft.github.io/Qcodes/api/dataset/index.html#qcodes.dataset.dond

By contrast, the experiments conducted in the present thesis do not place high demands on timing accuracy and measurement speed as typical photoluminescence (PL) integration times are relatively long. The software thus does not need to prioritize sophisticated triggering and pulsing logic required by qubit experiments. Because the requirements are rather specific, `mjolnir` is therefore single-minded. It does not aspire to be suitable for applications other than the ones it is designed for. The implementation is extremely biased towards the particular set of instruments in use in the lab and does not attempt to generalize to allow for different instruments to be used. At the same time, it is designed to be modular, and different functionalities should be able to easily be replaced by other solutions.

6.2 Instrument abstraction

Central to the `mjolnir` package is the abstraction of physical instruments into logical ones, thereby grouping logical functionality provided by different physical devices. Take for instance the tunable continuous-wave

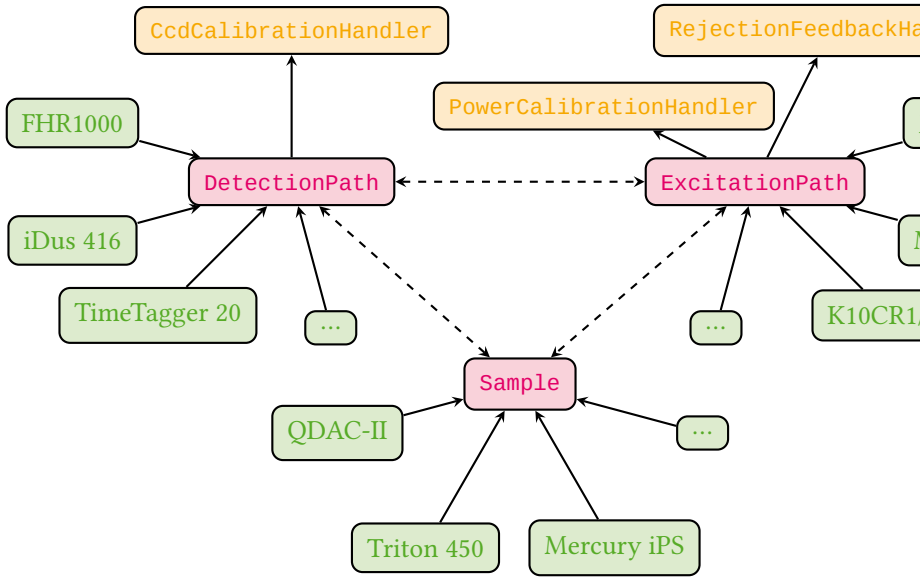

Figure 6.1: Abstraction of physical instruments. `mjolnir` defines three logical instrument classes (magenta) aspects of the experiment and group various physical instruments (green). Logical instruments expose various include communication with multiple physical instruments. `Sample` can be subclassed to suit the particular type gation. Calibration handlers (orange) are controlled by the logical instruments governing the optical path. Log of a `QCoDeS Station` and can interact with each other as indicated by the dashed arrows.

(cw) M Squared Solstis laser. It is cooled by a Thermotek T225p chiller and pumped by a Lighthouse Photonics Sprout-G diode-pumped solid state (DPSS) laser. Behind its exit aperture, a Thorlabs MFF101/M acts as a shutter, a neutral-density (ND) filter mounted on a Thorlabs K10CR1/M controls the output power, while a Thorlabs PM100D monitors the power at the optical head and an Attocube AMC100 controls the polarization state of the optical head. Thus, seven different instruments from five different manufacturers are required to control the illumination state of the sample. As a user, however, one would simply like to en- or disable the laser and set wavelength, power, and polarization configuration of the optical head.

To simplify and abstract away the particularities of the physical instruments providing control over parts of a user-facing parameter such as the excitation wavelength,[9] three logical instruments implemented as subclasses of the `QCoDeS Instrument` class govern the experimental apparatus:

9: For example, the power meter needs to be informed when changing the wavelength to keep the internal calibration up to date.

1. `ExcitationPath`. Controls all physical instruments related to illumination of the sample, including the white light source.
2. `DetectionPath`. Controls instruments related to the detection of radiation emitted by the sample, *i. e.*, the spectrometer, charge-coupled device (CCD), and photon counting card.
3. `Sample`. Controls the QDevil QDAC-II voltage source, cryostat, and magnet, and implements a software representation of the device under test (DUT).

Figure 6.1 shows a graph outlining the relationship between physical and logical instruments. In the following, I give a brief overview of the functionalities comprised by these logical instruments.

6.2.1 Excitation path

As already outlined above, the `ExcitationPath` object controls everything related to the illumination of the sample. The `active_light_source` parameter switches between laser, white light, and no illumination. Turning on the laser comprises enabling the chiller, switching on the pump laser, waiting for it to ramp up, and asserting the wavelength lock is acquired. For safety reasons, confirmation by the user that they are physically in the lab is required. Furthermore, the `ExcitationPath` instrument implements setting the excitation power by means of a calibration of the ND filter, see Section 6.3. Since the output power of the laser varies depending on wavelength, the object furthermore provides a `wavelength_constant_power` parameter that automatically recalibrates the power once a new wavelength is set. Finally, it manages the state of the automatic excitation rejection control, see Section 6.3.

6.2.2 Detection path

The `DetectionPath` object is responsible for the optical analysis. Chiefly, it manages the Horiba FHR1000 spectrometer and provides convenient shorthands for selecting grating (`active_grating`) and exit port (`active_detection_path`, either `"ccd"` or `"apd"`). It handles initialization of the spectrometer and the CCD including the CCD cooler and oversees calibration of the CCD pixel-to-wavelength relation, see Section 6.3. From the CCD calibration and its pixel size, the dispersion at the imaging plane of the currently selected grating can be obtained, allowing the user to set the monochromator bandwidth of the spectrometer in terms of a wavelength window rather than the more abstract exit slit width.

6.2.3 Sample

The `Sample` class initializes the QDevil QDAC-II digital-to-analog converter (DAC) and magnet power supply. Mainly, though, it serves as an abstraction of the actual DUT and its "control knobs" such as gates. Users implement subclasses for different sample designs. As the samples we are concerned with in the present thesis, I discuss the main implementation, `TrapSample`, representing samples with exciton traps. On a single sample, there are any number of traps consisting of one or both sets of top and bottom "central" and "guard" gates. Each trap is implemented as a `QCoDeS InstrumentModule`, `Trap`, that is part of a `ChannelList`. The currently active trap (*i. e.*, the one currently in focus by the microscope) is selected by the `active_trap` parameter and accessible through the `trap` property of the `TrapSample` object. A `Trap` exposes the virtual gate parameters[10] `difference_mode` and `common_mode` (see **??**) as the difference and sum of top and bottom gate voltages, respectively, as well as the unmodified top and bottom gate voltage and corresponding leakage current parameters. DAC channels are mapped to their corresponding `Trap` parameters using `yaml` configuration files specific to each sample and hosted in a separate repository.

10: Using the QDevil QDAC-II's built-in `Arrangement_Context` virtual gate functionality.

6.3 Calibrations

`mjolnir` implements three automatic calibration procedures in the classes `CcdCalibrationHandler`, `PowerCalibrationHandler`, and `RejectionMeasurements`

`FeedbackHandler` (see Figure 6.1). As their names suggest, these handle calibration of the CCD pixel-to-wavelength conversion, the power transmission through the rotatable ND filter, and the waveplate and polarizer angles for optimal laser rejection, respectively.

6.3.1 CCD calibration

Because the dispersion of the spectrometer gratings depends to a small degree on wavelength, the calibration function converting horizontal CCD screen pixels into wavelengths needs to be updated for every central wavelength selected using the grating. In principle, this requires measuring the position of several reference wavelengths on the CCD screen and fitting a polynomial of second or third degree. As the tunable cw laser is coupled to a wavelength meter with built-in calibration lamp (HighFinesse WS6), it lends itself naturally as a source of the reference wavelength. In practice, the deviation from an established calibration function is usually small enough that simply shifting the center pixel (that is, the zeroth-order polynomial term) suffices for reasonably small wavelength ranges of tens of nanometers. The `CcdCalibrationHandler` implements three different calibration schemes specified as the `ccd_calibration_update_mode` parameter of the `DetectionPath` instrument:

1. `"full"`. The pixel position of the maximum intensity of nine equidistant wavelengths centered around the grating central wavelength and spread out over the entire range of the CCD screen ($\sim 45\,\mathrm{nm}$ for the $600\,\mathrm{gr/mm}$ grating and $\sim 9\,\mathrm{nm}$ for the $1800\,\mathrm{gr/mm}$ grating) is measured sequentially and fitted to a third-degree polynomial.
2. `"fast"`. The laser is set to the grating central wavelength and the zeroth-order coefficient of the previous calibration polynomial is shifted by the pixel distance between the peak and the horizontal center of the CCD screen.
3. `"dirty"`. The zeroth-order coefficient of the calibration polynomial is shifted by the difference in nominal grating and laser wavelengths.

By default, the `"dirty"` mode is enabled as it does not require any grating moves or wavelength changes and is therefore the fastest. Rather than recalibrating, old calibration data can also be loaded from disk if it is not older than a user-specified date. Finally, note that the spectrometer grating motor also requires occasional recalibration. Both the coefficient of linearity (converting motor steps to selected central wavelength) and the offset in motor steps can change over time. Currently, this calibration is not automated but is instead compensated for by shifting the central wavelength on the CCD screen. In principle, the driver for the Horiba FHR1000 spectrometer implements all necessary functionality for automating the calibration procedure already.

6.3.2 Power calibration

11: Thorlabs NDC-25C-4 [2].

The ND filter[11] mounted on the Thorlabs K10CR1/M rotation stage has a continuously variable optical density (OD) (Equation 3.26) of 0.04 to 4.0, which depends quite sensitively on the wavelength. In order to allow reliably setting a specified excitation power at any excitation wavelength, the `mjolnir` package automates the calibration of the filter angle against transmitted power. Two modes are implemented, `"full"` and `"fast"`. In the former mode, the transmitted power is sampled at a given angular

interval and the resulting data fitted using a quadratic smoothing spline. Figure 6.3 shows a typical measurement and spline fit at 795 nm. The specified power is then set by inverting the spline and evaluating at the given power.

As performing the full calibration is quite time-consuming and furthermore the inversion not fully reliable, the `"fast"` calibration update mode sets the rotator angle by performing a noisy optimization. Here, first the last valid full calibration is loaded and its spline scaled to the current power level. Then, the residual sum of squares between the target power and the power meter reading is minimized using the `noisyopt` package [3] with the starting value given by the angle predicted by the rescaled calibration spline.

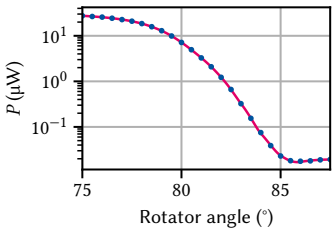

Figure 6.3: Power calibration using the continuously variable ND filter. Error bars from statistics are too small to see. Magenta line is a quadratic smoothing spline.

6.3.3 Rejection feedback

The excitation rejection by means of cross-polarization extinction has already been discussed in Subsection 3.1.4. As shown there, the OD achieved by cross-polarizing the excitation beam with the analyzer mounted in the detection path reaches a maximum close to 6 and is extremely sensitive to small variations in the angles of the $\lambda/4$- and $\lambda/2$-plates controlling polarization state. Close to the optimum, the OD is well-described by a parabola as function of those angles, with a second-order polynomial coefficient on the order of $-0.15/\mathrm{mdeg}^2$. Since the optics including the waveplates display chromatic dispersion, the excitation rejection needs to be re-optimized whenever the laser wavelength is changed, and the exponential angular dependence of the transmitted intensity places high demands on the dynamic range of the photo-detector used for measurement. Therefore, the reflected laser intensity is measured with the avalanche photodiode (APD) signal digitized by the Swabian Instruments Time Tagger 20 on the side exit port of the spectrometer.[12] The Time Tagger can reliably cover a count rate range from 10^2 cps to 10^6 cps, compared to the Andor iDus 416 CCD covering only 10^3 cps to 10^5 cps.

12: The signals from both APDs are combined to achieve the best signal-to-noise ratio (SNR), see Part II.

In the `mjolnir` package, the calibration is then carried out as a bivariate noisy minimization using `noisyopt` [3]. Upon initialization the power is set to a small value to avoid overexposing the detectors. Then, the spectrometer is set to select the laser wavelength, the side exit port is selected, and the optimization over the $\lambda/4$- and $\lambda/2$-angles is run within conservatively set bounds of $[-2.5°, 2.5°]$, which has been found to be a good compromise of robustness and convergence speed. A caching mechanism stores the optimal angles together with the current wavelength, which serves as a starting guess upon subsequent calibration runs. The calibration is quite tedious and time-consuming for several reasons. Chiefly, moving the waveplate angles is slow simply because it is a mechanical movement, and optimizers typically do not take into account the distance between subsequent sample points, potentially resulting in unnecessarily inefficient exploration of the parameter space. Incorporating a penalty on the sample distance into the optimization algorithm would hence improve the convergence speed. Additionally, due to vibrations in the cryostat (Chapter 4), the root mean square (RMS) intensity of the back-scattered laser is large – and the larger the closer the excitation rejection is to its optimal value[13] – and thus requires long averaging times for a robust measurement. Finally, the algorithm outlined above fails when the starting guess is too far away from the optimum, such as when the new wavelength is on the order of tens of nm away

13: The count rate can vary by up to a factor of two with the frequency of the pulse tube refrigerator (PTR) pulses.

from the previously optimized one. A good – albeit time-consuming – strategy is then to incrementally update the wavelength.

6.4 Measurement routines

Measurements in `mjolnir` use the `QCoDeS` infrastructure for data acquisition and storage. To abstract away code for repetitive setup and teardown tasks, the user interacts with a `MeasurementHandler` object through the `measure()` method. The class can be subclassed to customize the aforementioned tasks for different types of experiments (such as those that involve illumination with the laser and detection with the CCD, `LaserCcdMeasurementHandler`), or add default parameters to every measurement (such as leakage currents or the laser power). A measurement's independent parameters are passed to `measure()` as instances of a `Sweep` class which completely determines the measurement structure. `Sweep` objects support syntactic sugar for concatenation (`@`), nesting (`|`), and parallelization (`&`). Dependent parameters (those that should be measured) can be simple parameters or instances of a `Measure` class.[14] The state of external parameters, that is, parameters that are not varied or measured during a measurement, can be defined for the duration of the measurement using the `parameter_contexts` argument, a mapping from `QCoDeS` parameters to valid values thereof. This uses the `Parameter.set_to()` context manager to set the parameters to the given values before the measurement starts and restore their original values upon exit in a controlled manner, allowing users to ensure their device is in a well-defined state. Custom setup and teardown tasks can furthermore be specified through the `add_before_run` and `add_after_run` arguments, respectively.

14: Besides some optimization for measurements of parameters delegating from the same physical parameter, the latter exists mostly for (primitive) live plotting.

A typical workflow for a PL measurement using laser and CCD is sketched in Listing 6.1. The `MeasurementHandler` object automatically takes care of, among other things, arming the CCD, acquiring a background image in the dark, and opening the laser shutter for the duration of the measurement. Acquired data is saved to the `QCoDeS` database, but a helper function exporting to the `xarray Dataset` format is available. Saved together with the data are a snapshot of the `QCoDeS Station` as well as arbitrary custom metadata. More details and examples on the measurement functionality can be found in the documentation.[15] Lastly, I note that the measurement functionality in `mjolnir` is independent of the instrument abstraction and calibration logic and as such could be replaced or augmented with minimal effort by that of other `QCoDeS`-based frameworks such as `quantify` or `labcore` to make use of their infrastructure. I leave this option for future work.

15: https://qutech.pages.rwth-aachen.de/mjolnir/

d measurement
olnir package.
tion object man-
The sweeps ob-
loop on whose
rence mode pa-
central gate is
and on whose
ser power, ad-
plitter ratio, is
ally spaced grid.
rs (Measure ob-
licitly specified
rementHandler
ectrum as well
akage currents
y default. The
gument is used
er wavelength
common mode
trap to −1 V.
ument is passed
alize method,
up the CCD for

```python
from mjolnir.measurement.handler import LaserCcdMeasurementHandler
from mjolnir.measurement.sweeps import GridSweep
from mjolnir.measurement.measures import Measure

handler = LaserCcdMeasurementHandler(station)

# excitation_path, sample are instances of the ExcitationPath and
# TrapSample classes, respectively
power_sweep = GridSweep(excitation_path.power_at_sample,
                        rng=(5e-9, 5e-7), num_points=11,
                        spacing='geom')
gate_sweep = GridSweep(sample.trap.central_difference_mode,
                       rng=(-2, -1), num_points=51))

# two-dimensional sweep over power and gate voltage
sweeps = power_sweep | gate_sweep
# keep track of the MXC temperature
measures = sample.fridge.T8

dataset = handler.measure(
    sweeps,
    measures,
    parameter_contexts={
        sample.trap.central_common_mode: -1,
        detection_path.central_wavelength: 825,
    },
    exposure_time=2
)
```

6.5 Plotting

Optical measurements using `mjolnir` often produce multi-dimensional datasets by virtue of the default parameter measured, the CCD spectrum, being vector-valued or *batched*. To visualize such datasets with more than two dimensions, `mjolnir` provides the `plot_nd()` function that plots 2D slices through the data as a false-color image and generates interactive slider widgets that allow users to modify the slice coordinates. This facilitates interactively exploring large datasets and recognizing trends present in the data beyond two dimensions. Figure 6.4 shows an exemplary plot window for a four-dimensional data set obtained from a CCD measurement sweeping laser wavelength, power, and a gate voltage, with Listing 6.2 showing the code necessary to produce the plot.

The plotting functionality is implemented using `matplotlib`. While not the most performant library, its flexibility in decorating and customizing plots as well as scaling data, such as the asinh-scale normalization used in Figure 6.4, made it the preferred choice. To nonetheless improve the responsivity of the interactive elements, the plotted artists are *blitted*[16] using the `qutil.plotting.BlitManager` class from the `qutil` library [4]. The data variable to be shown in the main panel can be selected with the `array_target` argument, which accepts a string and chooses the closest match in the dataset since the full parameter names are often a bit unwieldy. Similarly, the coordinates to be plotted on the x- and y-axes are selected with the `horizontal_target` and `vertical_target` arguments, respectively.

In addition to plotting measurement data, `mjolnir` implements live plot-

16: Blitting refers to storing constant backgrounds of raster graphics and only redrawing changing elements. See the `matplotlib` documentation.

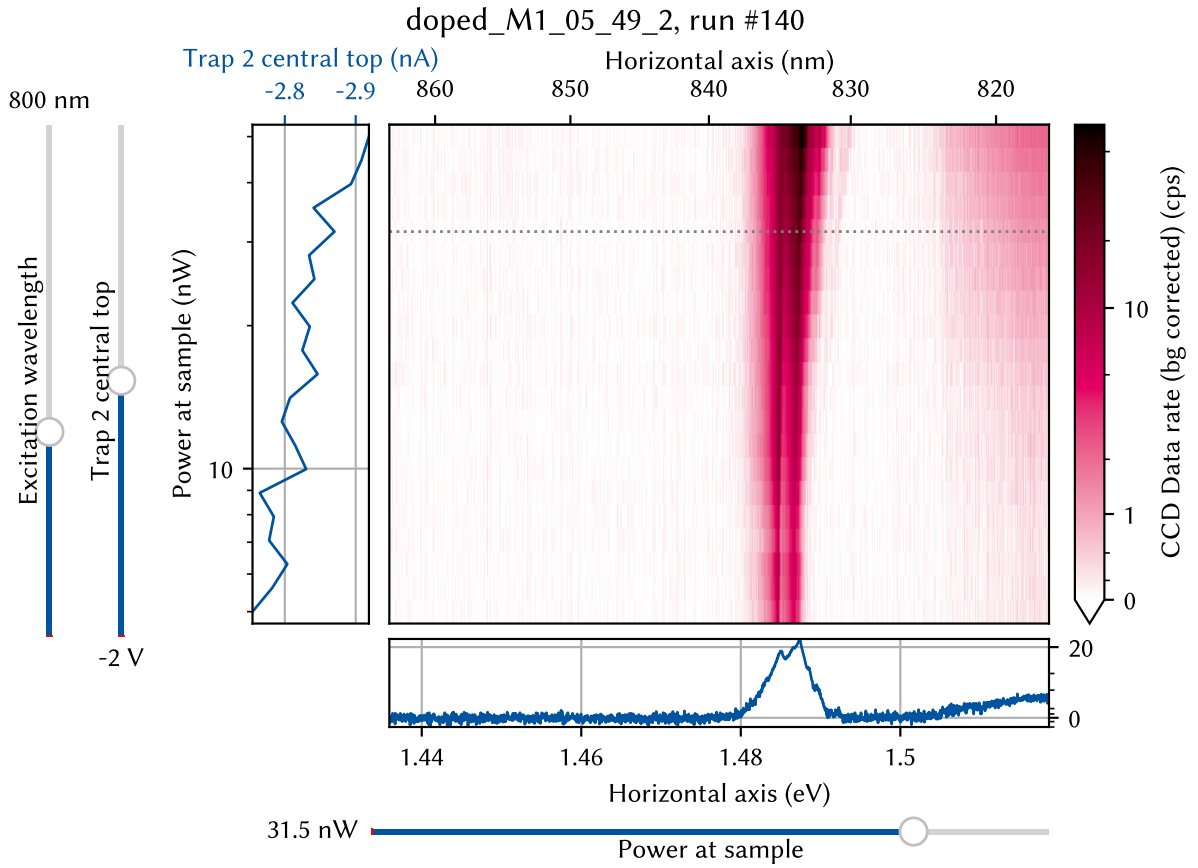

Figure 6.4: The `plot_nd()` plot window. The bottom panel, enabled by default, shows a horizontal line cut through the 2D data of the main panel whose y-value can be set with the adjacent slider widget. The left panel shows the leakage currents of gates that participate in the sweep and, if measured, the laser power. A vertical slider widget is added for each data dimension not displayed on the x or y axis of the main plot. The figure title shows the sample as well as the measurement identifiers assigned by `QCoDeS`.

ting of instrument data leveraging the `qutil.plotting.live_view` module. Since most instruments do not support multiple concurrent connections, switching control between the Python console and the graphical user interface (GUI) program provided by the manufacturer can often be tedious.[17] Thus, making use of their continuous data acquisition functionality is prohibitively time-consuming, but at the same time optical experiments frequently call for live observation of instrument data, for example when aligning the optics. Live plotting is currently implemented for three different instruments and corresponding parameters; the Thorlabs PM100D power meter (power reading), the Andor iDus 416 CCD (spectrum or image), and the Swabian Instruments Time Tagger 20 counting card (count rate(s)). Data acquisition is executed in a background thread while the plot, implemented in `matplotlib`, currently runs in the main thread, and hence does not update while the interpreter is blocked. In principle, though, the `live_view` module also supports plotting in a separate process, making a multiprocessing implementation straightforward. Interactive GUI elements in the `matplotlib` figure allow for pausing and automatically rescaling the data.[18] Since for the CCD data acquisition runs using the *run till abort* mode of the device and as such continuously, the instrument is blocked for the duration of the live plot and it therefore needs to be closed before starting a measurement. By contrast, the power meter and Time Tagger parameters are simply queried regularly in sequential mode and can hence in principle remain open and

17: For instance, the CCD GUI, Andor Solis, by default disables the thermoelectric cooler when exiting the program.

18: These two features can also be activated with the SPACE and R keys.

```python
import matplotlib as mpl
from mjolnir.plotting import plot_nd

fig, axes, sliders = plot_nd(
    dataset_or_run_id=140, vertical_target='power_at_sample',
    yscale='log^\prime, norm=mpl.colors.AsinhNorm(vmin=0),
    fig_kw=dict(figsize=(6.33585, 4.6))
)
```

Listing 6.2: Code to produce the plot shown in Figure 6.4. If `dataset_or_run_id` is an integer, the currently connected `QCoDeS` database is queried for this run identifier. Otherwise, it should be an `xarray` `Dataset`.

running while performing measurements.

Part IV

A Filter-Function Formalism For Unital Quantum Operations

Appendix

Supplementary to Part III: Additional measurements and TMM simulations

B

In this appendix, I give additional information on the bandstructure simulations and show further measurements supporting the interpretation of data shown in **??**. Moreover, I investiate the influence of the epoxy thickness on the simulations performed in **??**.

B.1 Self-consistent Poisson-Schrödinger simulation of the membrane band structure

The samples studied in **??** were all grown using molecular beam epitaxy (MBE) and had similar nominal designs. To compare the design charge carrier density with the measurements obtained from photoluminescence (PL) of the Fermi edge, I simulated the band structure using a self-consistent Poisson-Schrödinger solver [5]. Table B.1 shows the nominal doping concentration N_d as well as the charge carrier density in the QW, n, obtained from a simulation using the parameters given in Table B.2.

Table B.1: Heterostructure parameters used in the present thesis. The doping density N_d is nominal, whereas the charge carrier density in the quantum well (QW), n, is computed using the nominal doping values with parameters given in Table B.2.

Wafer	N_d (cm^{-3})	n (cm^{-2})
M1_05_49	6.5×10^{17}	1.95×10^{11}
15460 (Honey)	1.8×10^{18}	4.26×10^{11}
15271 (Fig)	8.0×10^{17}	3.91×10^{11}

Table B.2: Simulation parameters used to compute the charge carrier density n in Table B.1.

Parameter	Value	Unit
E_DX [a]	−71.5	meV
ΔE_c [b]	240	meV
m_c	0.067	m_e
Δz	0.5	nm
T	10	mK
V_FP [c]	0.76	V

[a] Energy of the DX-center below the Fermi level.
[b] Conduction band offset at the GaAs/AlGaAs interface.
[c] Fermi level pinning voltage.

B.2 Additional data

B.2.1 Combined plot of PL and PLE data

In **??**, I showed PLE measurements of a large exciton trap. When integrating the PL spectrum over the detection energy, the data showed several distinct features including an onset of the absorption that depends on the electric field applied by the difference-mode voltage V_DM. Figure B.1 shows the same data as panels (a) and (b) of **??**, but drawn in the same plot. It is clear that the onset of absorption occurs just as the PL emission drops off, indicating a small Stokes shift.

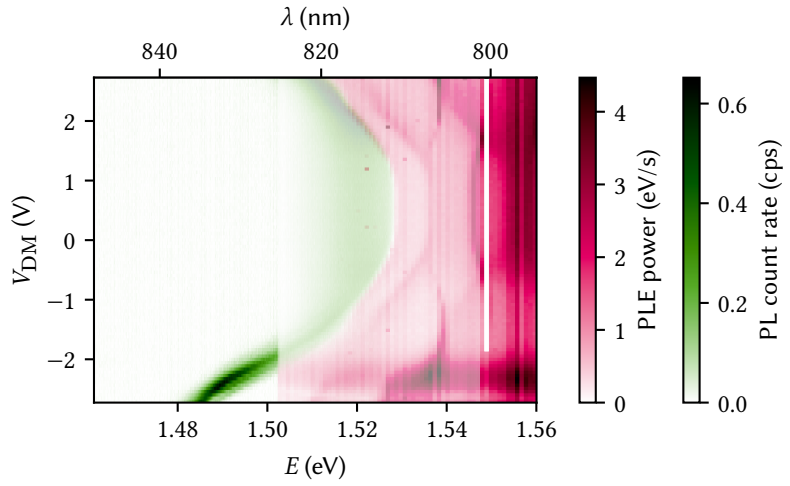

Figure B.1: Combined plot of the PL with excitation at 795 nm (green) and photoluminescence excitation (PLE) (magenta). The data overlap in the range of 801 nm to 825 nm, where they are plotted with 50 % transparency. The onset of the absorption edge in PLE coincides with the high-energy shoulder of the PL emission.

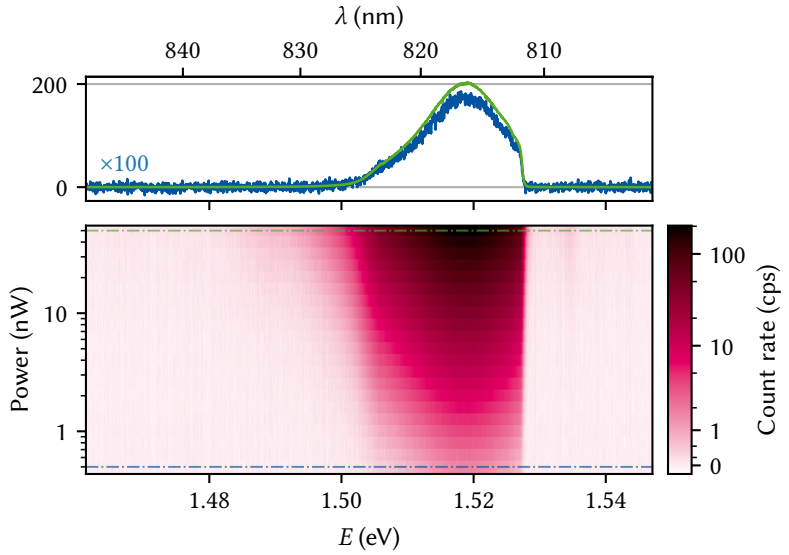

Figure B.2: Two-dimensional electron gas (2DEG) PL of an unbiased QW as function of excitation power. The Fermi edge at 1.5275 eV does not broaden over two orders of magnitude as the line cuts in the upper panels also show.

B.2.2 2DEG PL as function of power

In **??**, it was determined that the electrons recombining from the Fermi edge of the 2DEG were much hotter at ~ 1 K than the cryostat temperature at ~ 10 mK. The fit to the trion-exciton lineshape in **??** yielded similar results. Figure B.2 demonstrates that this is not a local heating of the lattice due to excess laser power. Shown is the 2DEG PL as function of excitation power from 5 nW to 500 nW, with the upper panel showing line cuts at highest and lowest power. There is no discernible broadening in the Fermi edge on the high-energy side of the spectrum.

B.3 Dependence of TMM simulations on epoxy thickness

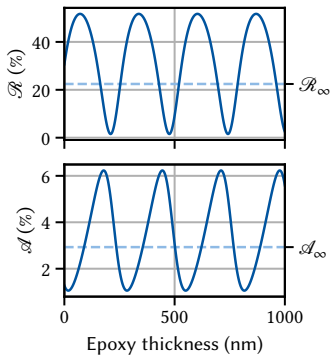

Figure B.3: Reflectance (top) and QW absorptance (bottom) as function of epoxy thickness assuming coherent back-scattering. The period corresponds to the wavelength in epoxy, $\lambda_{epo} = \lambda_0/n_{epo}$.

In **??**, I carried out transfer-matrix method (TMM) simulations of the membrane structure to elucidate the quenching of PL intensity when focusing the laser on gates on the top or bottom side of the membrane. There, I assumed incoherent scattering of photons at the epoxy/Si interface below the membrane. This assumption is based on the fact that Descamps [6] found the epoxy thickness, on the order of a few µm, to vary significantly across the chip. If the variation is fast enough, we can expect the phase of back-scattered photons to average out and hence destroy coherence. Nonetheless, we should estimate the influence of coherent back-scattering. Figure B.3 shows the reflectance $\mathscr{R}$ and QW absorptance $\mathscr{A}$ as function of the epoxy thickness. The dashed line indicates the value resulting from the incoherent simulation. Both quantities vary significantly with the epoxy thickness, but the incoherent values $\mathscr{A}_\infty$ and $\mathscr{R}_\infty$ are close to the average. From the fact that neither the PL nor the reflected laser intensity varies spatially by such large amounts in experiments, we can conclude that either the thickness variation is small enough to not play a significant role or that it varies fast enough to validate the assumption of incoherent back-scattering.

Bibliography

[1] Tobias Hangleiter, *Mjolnir* Aug. 8, 2025. URL: https://git-ce.rwth-aachen.de/qutech/python-mjolnir (cited on page 47).

[2] Thorlabs. *NDC-25C-4 Unmounted Continuously Variable ND Filter, Ø25 mm, OD: 0.04 - 4.0*. URL: https://www.thorlabs.com/thorproduct.cfm?partnumber=NDC-25C-4 (visited on 08/11/2025) (cited on page 50).

[3] Andreas Mayer, Thierry Mora, Olivier Rivoire, and Aleksandra M. Walczak. "Diversity of Immune Strategies Explained by Adaptation to Pathogen Statistics." In: *Proceedings of the National Academy of Sciences* 113.31 (Aug. 2, 2016), pp. 8630–8635. DOI: 10.1073/pnas.1600663113 (cited on page 51).

[4] Tobias Hangleiter, Simon Humpohl, Paul Surrey, and Han Na We, *Qutil* version 2025.8.1, Aug. 12, 2025. Zenodo. DOI: 10.5281/zenodo.16816394, (cited on page 53).

[5] Hendrik Bluhm, *poisson_schroedinger_1D* May 14, 2024. URL: https://git-ce.rwth-aachen.de/qutech/simulation/poisson_schroedinger_1d, (visited on 07/29/2025) (cited on page 55).

[6] Thomas Descamps. "Electrostatic Exciton Trap in a Thin Semiconductor Membrane for Optical Coupling to a GaAs Spin Qubit." PhD thesis. Aachen: RWTH Aachen University, 2021 (cited on page 56).

Glossary

Numbers
2DEG two-dimensional electron gas. xiv, 56

A
APD avalanche photodiode. 51

B
BS beam splitter. 53

C
CCD charge-coupled device. 48–54
cw continuous-wave. 47, 50

D
DAC digital-to-analog converter. 49
DPSS diode-pumped solid state. 48
DUT device under test. 48, 49

G
GUI graphical user interface. 54

M
MBE molecular beam epitaxy. 55

N
ND neutral-density. 48–51

O
OD optical density. 50, 51

P
PL photoluminescence. xiv, 47, 52, 55, 56
PLE photoluminescence excitation. xiv, 55
PTR pulse tube refrigerator. 51

Q
QW quantum well. 55, 56

R
RMS root mean square. 51

S
SNR signal-to-noise ratio. 51

T
TMM transfer-matrix method. xiv, 55, 56

Figure source files and parameters

Declaration of Authorship

I, Tobias Hangleiter, declare that this thesis and the work presented in it are my own and has been generated by me as the result of my own original research.

I do solemnly swear that:

1. This work was done wholly or mainly while in candidature for the doctoral degree at this faculty and university;
2. Where any part of this thesis has previously been submitted for a degree or any other qualification at this university or any other institution, this has been clearly stated;
3. Where I have consulted the published work of others or myself, this is always clearly attributed;
4. Where I have quoted from the work of others or myself, the source is always given. This thesis is entirely my own work, with the exception of such quotations;
5. I have acknowledged all major sources of assistance;
6. Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself;
7. Parts of this work have been published before as:

[1] Pascal Cerfontaine, Tobias Hangleiter, and Hendrik Bluhm. "Filter Functions for Quantum Processes under Correlated Noise." In: *Physical Review Letters* 127.17 (Oct. 18, 2021), p. 170403. DOI: 10.1103/PhysRevLett.127.170403.

[2] Thomas Descamps, Feng Liu, Tobias Hangleiter, Sebastian Kindel, Beata E. Kardynał, and Hendrik Bluhm. "Millikelvin Confocal Microscope with Free-Space Access and High-Frequency Electrical Control." In: *Review of Scientific Instruments* 95.8 (Aug. 9, 2024), p. 083706. DOI: 10.1063/5.0200889.

[3] Tobias Hangleiter, Pascal Cerfontaine, and Hendrik Bluhm. "Filter-Function Formalism and Software Package to Compute Quantum Processes of Gate Sequences for Classical Non-Markovian Noise." In: *Physical Review Research* 3.4 (Oct. 18, 2021), p. 043047. DOI: 10.1103/PhysRevResearch.3.043047.

Aachen, September 10, 2025.