Printing electrically tunable SQUIDs

THESIS

submitted in partial fulfillment of the
requirements for the degree of

BACHELOR OF SCIENCE
in
PHYSICS

Author :	T.L. de Vries
Student ID :	s2311976
Supervisor :	Dr. K. Lahabi - Prof. dr. J. Aarts
2^{nd} corrector :	Prof. dr. ir. T.H. Oosterkamp

Leiden, The Netherlands, July 9, 2021

Printing electrically tunable SQUIDs

T.L. de Vries

Huygens-Kamerlingh Onnes Laboratory, Leiden University
P.O. Box 9500, 2300 RA Leiden, The Netherlands

July 9, 2021

Abstract

SQUIDs, or Superconducting QUantum Interference Devices, are some of the most sensitive types of magnetometers known to date. The production of SQUIDs, which is typically a multi-step lithographic process, has limited the application of SQUIDs in non-planar and fragile systems. Recently, it was demonstrated that SQUIDs can be printed using a direct-write technique known as Electron Beam Induced Deposition (EBID), as was shown by van den Berg. The SQUIDs produced by van den Berg are a proof of principle, however, such SQUID devices are not yet suitable for real magnetometry measurements. This thesis aims to develop miniaturised EBID SQUIDs, where the flux can be electrically tuned. Hence, the SQUIDs can perform magnetometry measurements in real applications.

Contents

Version of July 9, 2021– Created July 9, 2021 - 16:58

Chapter 1

Introduction

1.1 Introduction

SQUIDs, or Superconducting QUantum Interference Devices have gained an undeniably important role in the field of magnetometry. These devices, namely, are known to produce extremely accurate measurements of flux, with sensitivity of up to 5×10^{-18} T [2], and noise as low as $3 \times 10^{-15}\,\mathrm{T}/\sqrt{\mathrm{Hz}}$ [3]. Since SQUIDs cannot only measure flux, but any quantity that can be converted to flux, they are extremely versatile, and thus used in many fields, such as the medical field, or in geophysics. The application of SQUIDs, however, has been limited by its fabrication process. This, namely, is a multi-step process, including lithographic metal deposition and lift-off. These processes require the sample to have a planar surface. Moreover, lithography requires a lot of specialised equipment, which is not always available. In addition, the lithographic processes are potentially detrimental to many systems, even further reducing the application base of SQUID measurements.
One specific example that goes to show these factors is the measurement of time-reversal symmetry breaking. Time-reversal symmetry breaking appears in some specific superconductors, such as Sr_2RuO_4 (see [4]), of which a micro-structured crystal can be seen in fig. 1.1(a) The special attribute of these types of materials, is that even in the ground state, their wave function has an intrinsic orbital momentum, meaning that their Cooper pairs spontaneously rotate around each other. These orbiting charged particles then create a local magnetic field. It can be theoretically simulated that the magnetic field of a mesoscopic ring of Sr_2RuO_4 should produce

a field like that in fig. 1.1(b). If one were measure this field[1], a magnetometer would be required that is both very sensitive, as well as able to measure adequately locally. A carefully positioned nanoSQUID can realise both these things, however, because the crystal surface is non-planar, lithography cannot be conducted, and thus the field has never been imaged.
Recently, Blom et al. and van den Berg have provided an alternative, direct-write fabrication method, that requires only a Scanning Electron Microscope (SEM) equipped with a gas injection system. In this process, Electron Beam Induced Deposition (EBID) is used to deposit both superconducting and non-superconducting material in the vacuum chamber of the SEM, making SQUID fabrication possible. Not only does this eliminate the need for lithography and vastly reduce the amount of equipment needed, but the process is also suitable for non-planer surfaces, such as the Sr_2RuO_4 crystal. Moreover, the process allows for careful placement of the device onto the sample.
van den Berg has shown the production of multiple functioning SQUIDs with this method, however, in addition to being relatively large, these are not suitable for real magnetometry measurements. Specifically, the SQUID needs to be electrically tunable to be used effectively (see sec. 1.2.1). This brings us to the aim of this thesis: Building upon the methods of van den Berg and Blom et al. to develop an EBID SQUID that is ready for practical applications, by making them flux tunable firstly. Secondly; able to perform more local measurements, by reducing the dimensions. And thirdly; more robust and reproducible.

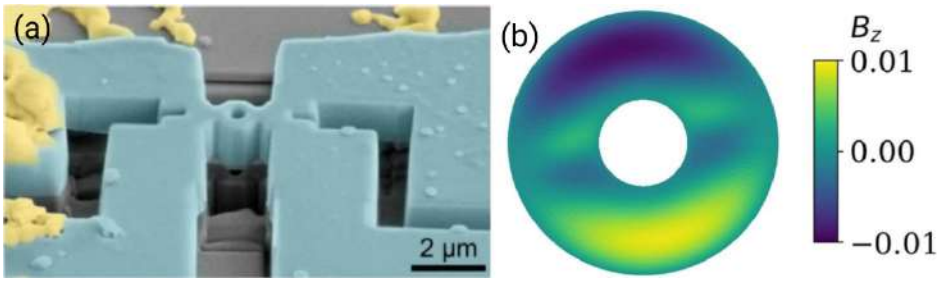

Figure 1.1: *(a) Image of circular, mesoscopic, Sr_2RuO_4 crystal. (b) Theoretical simulation of the magnetic field produced in the crystal in (a) due to time-reversal symmetry breaking. magnetic field in units of critical field of Sr_2RuO_4 ($\sim$ 72 mT [4]). From [6].*

[1] Sr_2RuO_4 remains, to this day, a mysterious material, because there are large discrepancies between theory and empirical evidence, making it an attractive material to study. For more information, see [4].

1.2 Basic concepts

In this section, the reader is provided with some of the basic theoretical concepts involved in this document.

1.2.1 DC Superconducting QUantum Interference Devices (SQUIDs)

The SQUID was built upon the findings of Brian David Josephson in his 1962 paper "Possible new effects in superconductive tunnelling" [7]. In this, Josephson predicted that the configuration shown in fig. 1.2, would allow for current flow through the barrier, driven by a phase difference between the two superconductors (as opposed to a voltage) . The behaviour of these junctions was put to good use in the development of the SQUID. In this work, we will focus specifically on the configuration of a DC SQUID[2], which is visible in fig. 1.3

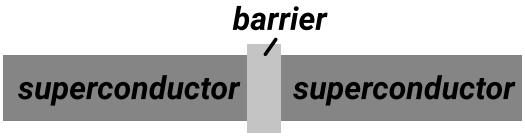

Figure 1.2: *schematic representation of a Josephson junction.*

The DC SQUID simply consists of two, parallel, Josephson junctions. Because the junctions couple the superconductors on each side, we effectively have a superconducting loop. This loop has a macroscopic phase, which must wind by multiples of 2π. As a result, the magnetic flux through the loop is quantised. That is, flux must always thread the loop in integer multiples of the flux quantum $\phi_0 = \frac{h}{2e} = 2.067 \times 10^{-15}$ Wb (For more details see [8]). Whenever an external field is applied, which does not account for a full quantum, a screening current I_s will appear in the loop (see fig. 1.3 for the current flow). This current produces extra, or opposing, flux, in order to bring the total flux up, or down to a full quantum.

To measure this screening current we can add a bias current I_b to the loop (see fig. 1.3). By sweeping this bias current we are able to find the critical current of our SQUID[3]. From fig. 1.3 it is visible that in one of

[2]The other commonly used configuration is the RF SQUID, for more information I refer to [8].

[3]In a perfectly symmetrical SQUID, and in the absence of any screening current, this critical current should be exactly twice the junction critical current

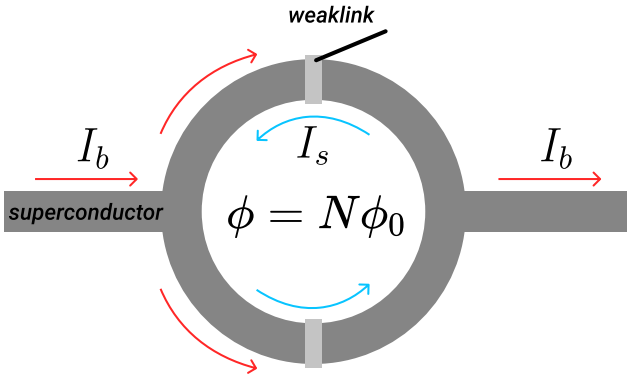

Figure 1.3: *Schematic representation of a dc squid. dark grey areas indicate superconducting material, and light grey areas the weak link material. arrows indicate current flow.*

the SQUID arms, the bias current will add to the screening current. This effectively lowers the bias current needed to reach the critical current. Essentially, the critical current gets suppressed. This is visualised on the left of fig. 1.4. The non-linearity of this IV curve is ultimately what allows for sensitive measuring. If we now set the bias current, like is shown in this graph, and sweep the flux, a voltage modulation will appear. This is visible on the right of fig. 1.4. Seen as the screening current will be periodic in the flux quantum, so will the suppression of critical current be. Because of the Josephson junctions, part of the current in the loop will be accounted for by the phase difference. This allows us to measure the critical current without losing the quantum behaviour. The flux quantum, being made up of physical constants only, makes the periodicity of SQUID patterns universally predictable. This allows SQUIDs to be made with extreme sensitivity.

If we are trying to measure some small field (thus around zero flux), we will have a very low gradient in our voltage modulation (see right graph in fig. 1.4), thus not allowing for sensitive measurement. Ideally, we would want to offset our modulation pattern by a quarter of a period, so that small fields are always measured at the maximum gradient (the slope of the sinusoidal curve). To achieve this, one could apply an additional external magnetic field to the sample, supplying roughly $\frac{1}{4}\phi_0$ of flux to our SQUID loop. Though this would work in some cases, in applications such as the Sr_2RuO_4 crystal shown in sec. 1.1, this external field would affect the system, and therefore cannot be used. To offset our pattern otherwise, requires intrinsic electric tunability.

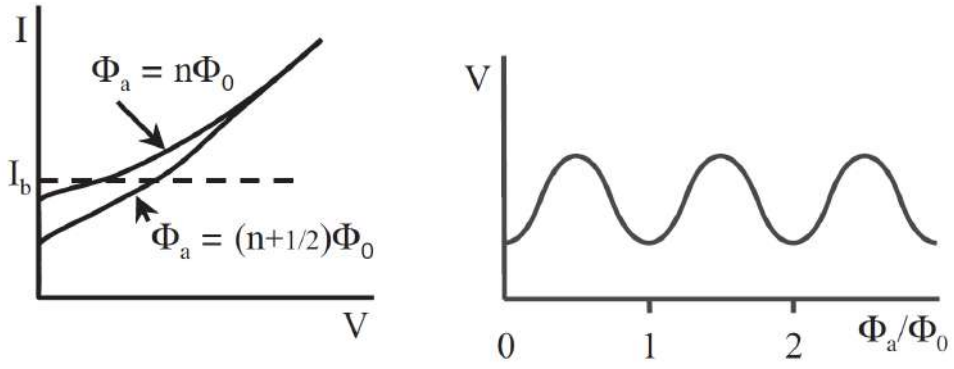

Figure 1.4: left*: Non-linear voltage current relation for the dc squid for integer multiples of ϕ_0 and half-integer multiples of ϕ_0, visible is suppression of the critical current.* right*: Flux-voltage relation for the dc squid if biased with current as is shown in the left image. We see voltage oscillations periodic in ϕ_0. From [8].*

1.2.2 Electron Beam Induced Deposition (EBID)

The technique used to fabricate SQUIDs in this thesis is known under the name Electron Beam Induced Deposition or EBID. The process is schematically shown in fig. 1.5. The whole operation is conducted in a high vacuum inside of an SEM.
Firstly, a gas needle (coming from a gas injection system integrated with the SEM) dispenses gas molecules (also called precursor molecules) above a substrate. Both the precursor and substrate can be made of many different materials but in our case, we exclusively use $W(CO)_6$ as a precursor and Si with a layer of SiO_2 as the substrate. Then, an electron beam is shone onto the substrate and precursor molecules. Some electrons will collide with the precursor molecules, and disassociate a carbon monoxide group from the tungsten. The tungsten can now bond to the substrate. By this process, a deposit is formed under the electron beam.

The properties of the material that is deposited during EBID are highly affected by the parameters of the electron beam and the SEM environment (for a detailed overview of EBID parameters and their effects see [10]). The main causes for this are differences in the amount, and type of electrons involved. To elaborate, the electrons produced directly by the beam are called Primary Electrons (PE's), these have relatively high energy. Once PE's reach the substrate (or deposit) the electrons are scattered into lower

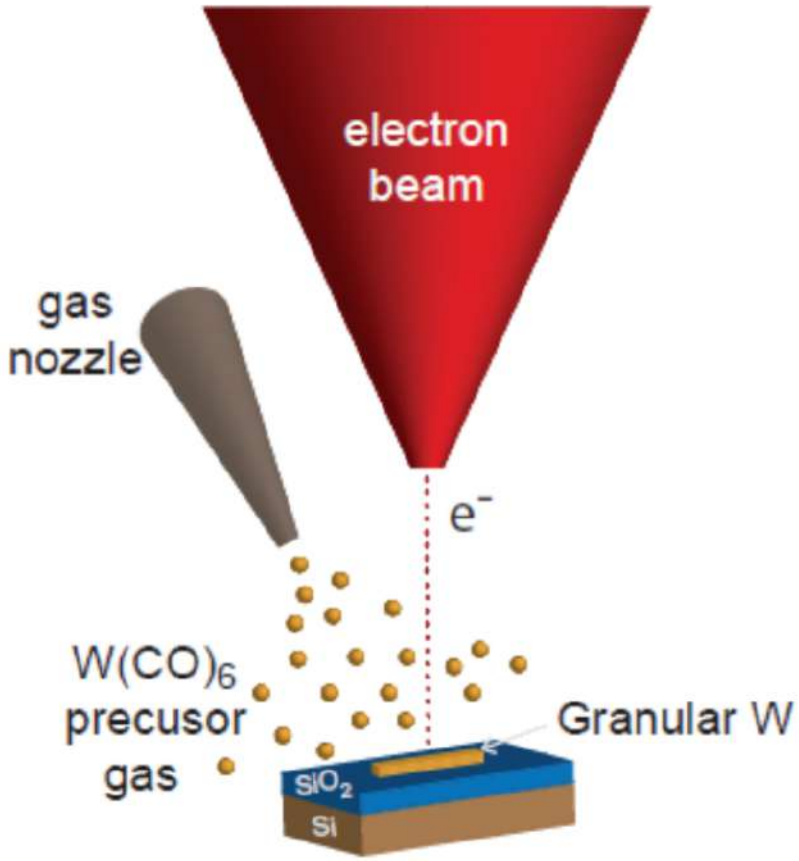

Figure 1.5: *Schematic representation of Electron Beam Induced Deposition. Precursor molecules are dispensed by a gas nozzle, and disassociated by an electron beam. The non-volatile species form a deposit on the substrate. From [9].*

energy Back Scattered Electrons(BSE's), and Secondary Electrons (SE's). Ultimately, this latter group is mostly responsible for the disassociation of the molecules, but their presence is affected, among others, by the already existing deposit. Aside from the beam energy and beam current, the most relevant parameters for our purpose are the pitch and the dwell time. The pitch determines the step size of the electron beam, a smaller pitch will give more overlap between consecutive beam positions, and thus an ultimately higher dose per irradiated area. The dwell time has a similar effect, as it states the duration for the beam to stay in each spot.

1.3 Previous works in Leiden

Blom et al. showed the fabrication of relatively high T_c ($\approx$ 4.7K) material using EBID, which is what allowed the fabrication of superconducting devices. It was found that by using the right parameters, both superconducting, as well as metallic material could be deposited in the same SEM environment. A resistance-temperature measurement of both these materials, as well as an image of the superconducting wire can be seen in fig. 1.6. Specifically, a beam energy of 10 keV, and a beam current of 20 nA were used. To create the superconducting material, a dwell time of around 25 ms was used, and a pitch of approximately 1 nm. For the metallic material much shorter dwell time (4 ms), and a larger pitch (14 nm) was used. These materials were used to create Josephson Junctions. See [5].
Using the methods of Blom et al., van den Berg was then able to produce SQUIDs. One of these SQUIDs, along with a field measurement can be seen in fig. 1.7. From the field measurement, it can be seen that the critical current of the device, is oscillating with respect to the applied magnetic field. Although these SQUIDs function, they are limited in use for a couple of reasons. Firstly, the area of the SQUIDs is relatively large ($\sim$ 1.4 µm^2), making it unable to measure fields very locally (for comparison, the outer radius of the Sr_2RuO_4 crystal from sec. 1.1 (fig. 1.1) is $\sim$ 0.7 µm). Secondly, the design includes quite some curves, and an overall complicated pattern to print. This allows for unwanted inconsistencies in the wires and overall irreproducible results. Most importantly, these SQUIDs cannot be electrically tuned. It should also be noted that, as discussed by van den Berg, all devices show some unexplained residual resistance ($\sim$ 0.5 Ω), even at the base temperature of the measuring devices ($\sim$ 1.5 K).

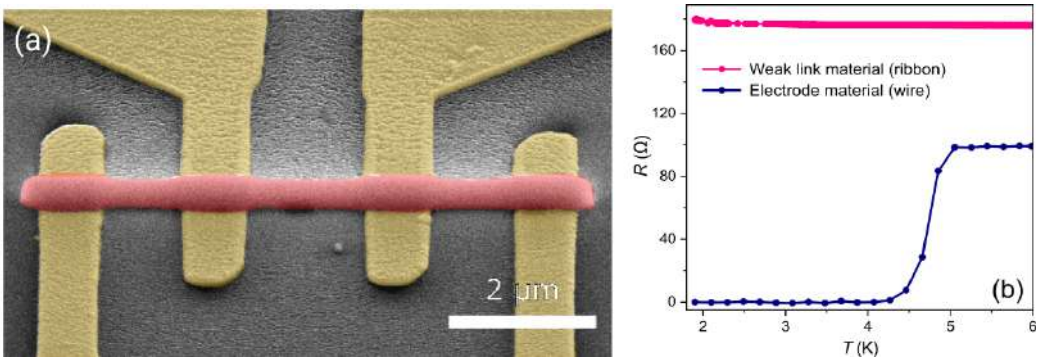

Figure 1.6: *(a) SEM image of wire printed with EBID. (b) Resistance-temperature measurement of wire in (a) in blue, and of weak link material in pink. From [5].*

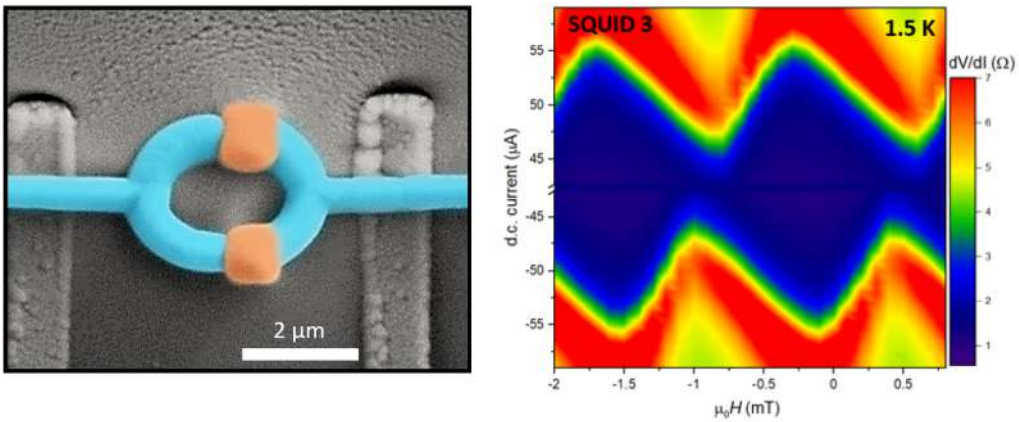

Figure 1.7: left*: False colour image of an EBID SQUID.* right*: field measurement of SQUID in the left image. From [1].*

Chapter 2

New SQUID: Design and Development

As explained in the previous chapter, this research aims to produce an EBID SQUID, suitable to perform accurate measurements on non-planar surfaces. This chapter will provide the results that were obtained during the research, as well as outline some of the thought processes that have gone into each consecutive decision that was made.

2.1 Original concept and design

To make a more applicable SQUID, we set the following three design principles: Firstly, we aim to minimise the area of our SQUID. This is to ultimately produce more local measurements. Secondly, as is explained in sec. 1.2.1, it is important to allow for electric tunability in the design. Lastly, in effort to make a more reproducible SQUID, we would like to maximise the simplicity of our design.
Looking at the SQUID in fig. 1.7 We can see that the wires have a certain thickness ($\sim$ 300 nm), which cannot be reduced. Because of this, the wires cannot be bent further than they are currently. Thus, to miniaturise the SQUID we are required to rethink the complete design. These considerations led to the design displayed in fig. 2.1. Here, dark grey material, indicating superconductor, and light grey material, indicating a weak link, can be seen. The contacts by which the device will be measured are drawn in gold. These are not part of the SQUID, but rather of the substrate that they are printed on.

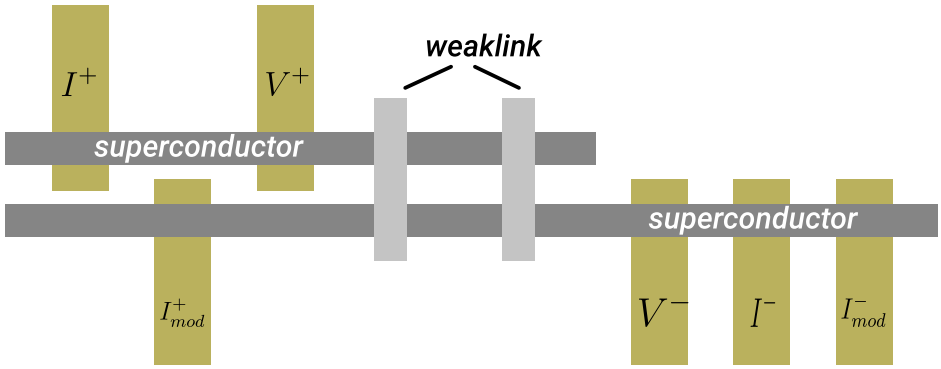

Figure 2.1: *Design of new SQUID. dark grey areas indicate superconducting material, and light grey areas the weak-link material. Gold squares indicate the contacts with their intended purpose*

The design full fills all criteria: Firstly, the vertical dimension is determined by the preferred weak link length for the junctions (which is about 150 nm in our case, see [5]), which puts this vertical dimension directly at the low limit. The horizontal dimension of the SQUID is determined solely by the distance between the weak links, which can be freely set to be a small as feasible. Combined, the minimalisation of these dimensions allow for an ultimately small area for the SQUID. Secondly, as is apparent from fig. 2.1, a so-called modulation line is included in the design. By passing a current through the contacts labelled I^+_{mod} and I^-_{mod}, the superconducting wire will gain a magnetic field around it. Part of this field will pierce our SQUID loop. This allows us to electrically tune the SQUID, as is explained in sec. 1.2.1. Finally, the design includes only four straight lines, making for a much simpler printing process as compared to the design used by van den Berg. The hope is that this will create a more user-friendly, and reproducible deposition process.

2.2 Methods

In this section, the exact fabrication process is outlined, as well as the measurement process.

2.2.1 Creating a StreamFile

The SEM we are using is equipped with software, to make patterns to be used in EBID. This software, however, is relatively limited in that it only allows for straight lines and rectangles, among a few other simple shapes.

Even more importantly, the software does not allow for control over the printing direction (a line from point A to point B will be scanned from A to B, and then from B back to A. In some cases, we would only want to go from A to B, but not back). To make more intricate designs, and have more control over this scanning direction, a so-called StreamFile (SF) is required. This is a file that contains a list of coordinates for the electron beam, along with a dwell time for each position, in a format readable to the SEM. To create these files, a python code was used (see B for the full code). It is noteworthy that we only use these SFs for the pattern of the superconducting wires, whereas the weak links will be printed using a pattern from the SEM software (as was also done by van den Berg).

2.2.2 Device fabrication

Once the patterns are created the process of fabrication starts. Firstly, a substrate is prepared. the substrate is made of silicon (Si), with a layer of Silcondioxide (SiO_2) and gold (Au) contacts, created by lithographic deposition and lift-off. The placement of these contacts is specifically designed to fit our design but does change per substrate[1] as the needs for our design develop. The substrate is loaded into an Apreo SEM, which is equipped with a gas injection system (when a substrate is not in use it is stored in a vacuum). Once loaded, the parameters of the SEM are set (beam current, beam energy, etc.), and it is prepared for deposition (setting the sample height relative to the detector). We can now rotate the sample to our preferred orientation and insert the gas nozzle (the position of the nozzle is set absolutely with respect to the SEM, which is why it is important to set the sample height correctly).
At this point we want to load our patterns. The SEM is set to 20000x magnification (all SFs are made specifically for this magnification), and the desired SF is uploaded via the patterning dialogue of the SEM. The pattern is positioned on the contacts as was designed, and parameters are set (for SF this only includes the number of passes and the precursor gas used). Now, again via the patterning dialogue, we insert a rectangular, SEM generated pattern. We fit the rectangle to one of the weak links as is dictated by the design and set the printing parameters following the findings of Blom et al. and van den Berg. These include the dwell time, pitch, precursor gas, and the number of passes. For the second weak link, the rectangle pattern is simply copied. It is also important that the patterning order is set to parallel (rather than serial so that both weak links can be printed

[1] Each substrate will typically contain 9 to 16 identical sets of contacts

simultaneously).
Now we carefully focus the beam and get ready for deposition. The patterns for both weak links are disabled so that only the SF is enabled. The number of passes is set, and the patterning can now be started. Once the pattern completes, and the result is as expected, the SF is disabled and the weak link patterns are enabled. The number of passes is reset[2], and the patterning is started again.
It is unlikely for the pattern to come out as expected in one try. Therefore it is common practice to test the patterns beforehand on a blank part of the substrate. Once the design is as desired the beam is moved to the contacts and the final device is printed. If the end result looks good, the patterns can be exported for later use (this way the exact placements of weak links relative to wires can be kept constant in between sessions).

2.2.3 Transport measurements

To test our devices we generally have taken two types of measurements. Both of these take place inside of a cryostat. To start, we measure the resistance of our device against temperature (RT). Here, we send two current values through the contacts labelled I^+ and I^- in fig. 2.1, and then measure the voltage over the V^+ and V^- contacts. From this, we can calculate a resistance. This we repeat while the temperature of the device is constantly being lowered down to around 1.5 K. The second type of measurement, which we will call a field sweep, is used to find field dependence. With the temperature at a constant value (below the superconducting transition), we take a current-voltage (IV) measurement of the device. This is essentially done in the same way that we measured the resistance, however, now we take a range of values for current rather than just two. This should show the critical current of the device. We now repeat this measurement over a range of values for an externally applied, out of plane, magnetic field[3]. If we see that the critical current oscillates with respect to the field, we have found SQUID behaviour.

[2]When the SEM is set to parallel deposition it requires all patterns to have an equal number of passes, therefore it needs to be reset in between.

[3]An estimate can be made on the area of the SQUID from the SEM images. This translates to an expected periodicity in the magnetic field. For us to conclude on SQUID characteristics, at least a few periods should be measured.

2.3 Development and intermediate results

In the following section, we will go over the findings that were made during the development of the new SQUID. These findings will be reported in chronological order, as each iteration of our design was made to solve a problem of the previous.

2.3.1 Initial attempt

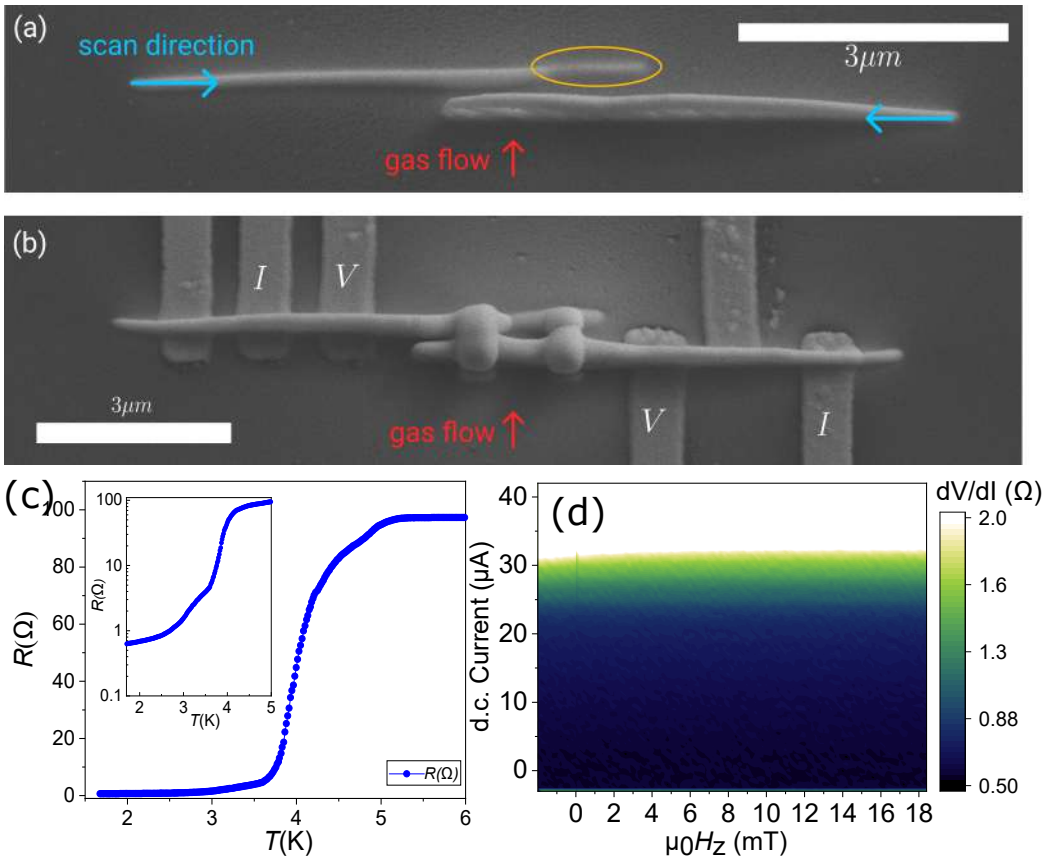

Figure 2.2: *(a) SEM image of parallel EBID wires. Gas flux direction is indicated with a red arrow, the scan direction in blue. The scale bar is* 3 µm*. The Back wire is shadowed by the front wire and the deposit is much thinner (highlighted). Taken during testing phase. (b) SEM image of early SQUID device. Current and voltage contacts are indicated with I and V respectively. (c) Resistance temperature measurement of the device shown in (b). A transition is visible. (d) Field sweep of the sample shown in (b) at* 1.5 K*. The critical current (around* 42 µA*) does not appear to be dependent on the magnetic field.*

Version of July 9, 2021– Created July 9, 2021 - 16:58

In the first attempts, the device shown in fig. 2.2(b) was fabricated and measured[4] (fig. 2.2(a) shows the SQUID wires, before weak link placement). In fig. 2.2(c) we see the resistance temperature measurement of the device. In this RT curve, we can distinguish multiple transitions, of which one, around 4 K, is rather large. This large transition most likely corresponds to the wires going superconducting, as they will make up most of the resistance. The inset shows the RT plot on a logarithmic scale. Here we can distinguish another, smaller, transition at a lower temperature, as well as a long tail (meaning the resistance is still decreasing at base temperature and does not reach 0 Ω[5]). This small transition could suggest the onset of the proximity effect in the weak links (as was the case in SQUID devices from van den Berg, and the junctions of Blom et al.). The smaller transitions above 4 K are not explained as such, and indicate, that sections of the device might be transitioning separately. In succession to the RT measurement, a field sweep was performed at 1.5 K which is shown in fig. 2.2(d). Here we can see that the critical current is not affected by the application of a magnetic field. The device does not show SQUID behaviour.
To explain the failed characteristic, we consider the apparent inhomogeneity of our system. Fig. 2.2(a) and (b) reveal these inconsistencies. Specifically, in fig. 2.2(a), we can distinguish a particularly thin section of deposit, on the right side of the back wire (highlighted). This is caused by what we will call shadowing. Shadowing is reported by van Dorp and Hagen and is determined by the gas flow direction, relative to the scan direction, which is shown in fig. 2.2(a). If we consider the scan direction of the wires, we can conclude that the highlighted area is being deposited while there already is an existing wire section in front of it. This section acts as a dam, simply blocking precursor molecules from reaching the area behind, causing a thin deposit.
To circumvent this effect, it was decided to change the direction of gas flow. This led to the device in fig. 2.3. This device will be the subject of the next subsection.

2.3.2 Pillaring and detector mode

After changing the gas direction, the design of the new SQUID was slightly altered. In order to produce an effective SQUID it is important that the sec-

[4]The modulation line in this device was left out due to a small error in the contact placement. At this point in the research, this line is not essential.

[5]In the reports of Blom et al., all produced SQUIDs show some finite resistance, even at base temperature. Even though this remains unexplained it has not been detrimental for the functioning of the SQUIDs

Version of July 9, 2021– Created July 9, 2021 - 16:58

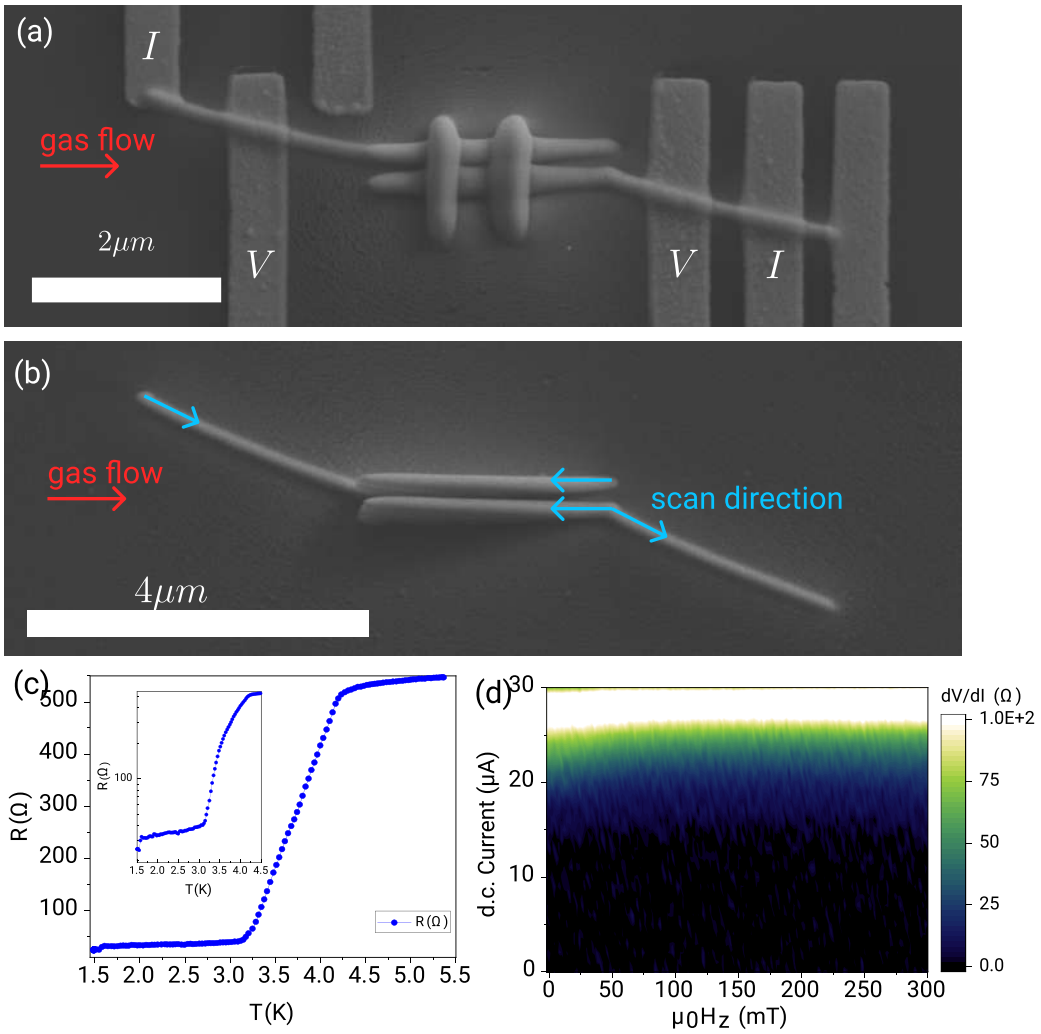

Figure 2.3: *(a) SEM image of the deposited device. red arrow indicates gas flux. (b) SEM image of deposited wires before weak links (taken during testing fase). We see that the parallel wires are much more symmetric as compared to fig. 2.2. Blue arrows indicate directions of printing. (c) Resistance temperature measurement of the device in (a). A transition appears around* 4 K. *(d) Field sweep of the device in (a). The critical current appears to not be affected by the magnetic field.*

tions of wire, which are going to receive the weak links, are symmetrical (see [8]). To this end, we required these sections to be printed simultaneously, and in the same direction. These considerations resulted in the scanning strategy apparent from fig. 2.3(b), and ultimately the device shown in fig. 2.3(a).
We can see from the wires in fig. 2.3(a) that indeed they are much more

symmetrical, in comparison to the wires in fig. 2.2(a). But how does this device perform? The field sweep in fig. 2.3(d) unfortunately shows no field dependence of the critical current. The device is not a SQUID.

For an explanation we again analyse the RT of the device, which is shown in fig. 2.3(c). Here, rather than seeing multiple, separate transitions, we see one, relatively broad transition. This suggests an inhomogeneous system. fig. 2.3(b) appears to reveal some of these inconsistencies. We can see that the parallel sections of wire are growing in height as they are being scanned. Looking closely at the wires we can distinguish a build-up of inconsistent material, below the surface of the wire.
The cause was found in a previously reported effect (see [10]), which we will call pillaring. Pillaring occurs when each section of the substrate is irradiated by too many electrons. This will allow the deposit to grow slightly upwards, creating a small pillar, and causing inconsistent material. Pillaring can be avoided, rather straightforwardly, by lowering the dwell time of the beam or increasing the step size. It was found, however, that the most consistent solution to this issue was changing the detector mode.
In all initial attempts the detector was set to Optiplan mode, in which the beam is further focused, and thus reduced in diameter, by magnets ($\varnothing \approx 35\,\mathrm{nm}$). This results in a higher spatial resolution for printing, but also in much more sensitivity to focus. It was found that differences in focus, however slight, were majorly affecting the existence of pillars in the structures. For this reason, the detector was switched to standard mode for the remainder of deposits ($\varnothing \approx 70\,\mathrm{nm}$).

2.3.3 The issue with EBID wires

The design was further changed after the results of the previous section. This included reducing the size of the weak links, as they are unnecessarily large. In addition, the top and bottom wires are made symmetric by adding two more wire sections, which also reinstalls the option for a modulation line. Lastly, the corners in the design are rounded in hopes of avoiding a contaminated interface at the angle. These changes resulted in the device that can be seen in fig. 2.4(a). In contrast to the previous attempts, we see that our wires are now much more consistent, and do not show any obvious deformities.
Measuring this device has led to the RT shown in fig. 2.4 (b). It appears indeed that we are now getting a single, sharp transition around 4 K, rather than multiple, or broad transitions. Then, at lower temperatures, we seem

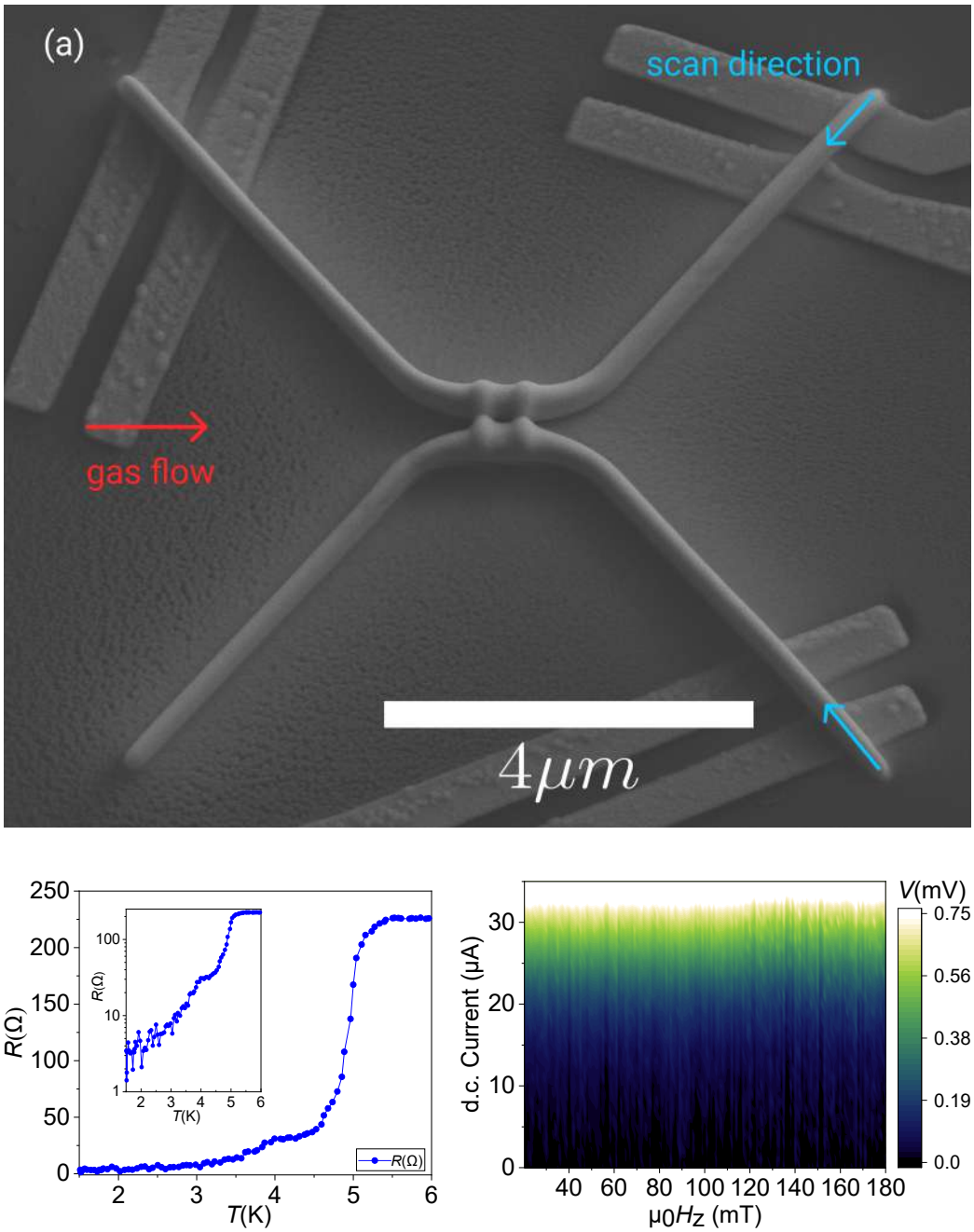

Figure 2.4: *(a) SEM image of device. Red arrow shows gas flow direction, blue shows scan direction. (b) Resistance temperature measurement of device in (a).(c) Field measurement of device in (a).*

to get another small transition, as well as a long tail. This was assumed to signify the leftover resistance from the weak links after the wires have

gone fully superconducting. With high hope, this device was now measured for field dependence. The results can be seen in fig. 2.4 (c). Unfortunately, it appears that this device, too, shows no sign of SQUID behaviour. Because the device cannot be faulted on appearance, we try to find an explanation for its failure in the RT of the device (see fig. 2.4 (b)), and specifically this long tail. As explained we assume that this tail is the result of the weak links somehow, but this had at this point not been confirmed. To test this suspicion, an experiment was set up, by which an RT of the device in fig. 2.4 was taken along two current paths. These are shown in fig. 2.5 (a) As we can see, one of the paths runs through the entire SQUID, much like we have done before, whereas the other path runs only through the wire. An RT was taken along both paths, the results of which can be seen in fig. 2.5(b). These results are rather interesting. Contrary to our assumptions, the wire measured separately is wholly reproducing the second transition and large tail seen in the RT of the SQUID. This was assumed to originate in the weak links. More importantly, this result is inconsistent with the RT of superconducting wires produced by Blom et al., as shown in fig. 1.6(b) Namely, we are not reproducing the single sharp transition around 4.7 K, nor are we reaching $0\,\Omega$ at any measured temperature. This behaviour should be resolved before any further attempts at producing a SQUID are made.

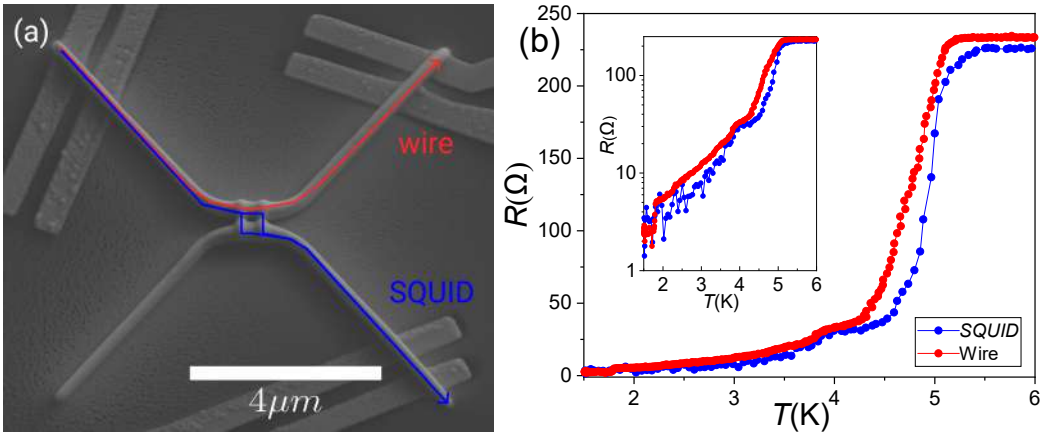

Figure 2.5: *(a) Arrows show two current paths through the device from fig. 2.4. One goes through the SQUID, the other only though the wire. (b) Resistance temperature measurements taken along the current paths shown in (a). Both current paths have similar characteristics.*

2.3.4 Identifying the issue with the EBID wires

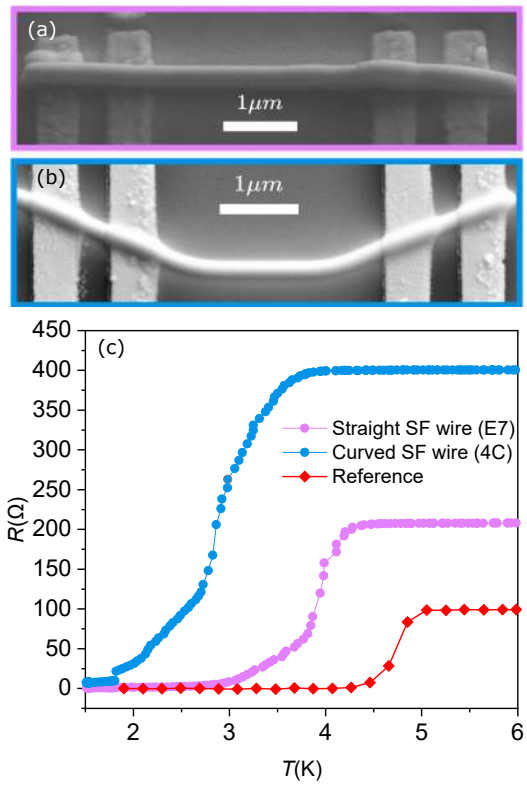

Figure 2.6: *(a) Curved EBID wire. (b) Straight EBID wire. (c) Resistance temperature measurements of wires shown in (a) and (b). Reference is the wire in fig. 1.6.*

After the findings of the previous paragraph, it was concluded that further experimentation should focus on dissolving the incongruity between our results and those of Blom et al..
Initially, it makes sense to look for flaws in the pattern, as the python code used to make them was handwritten, and not nearly as rigorously tested as the other parts of the process. Firstly it was tested whether the geometry of the wires was affecting our RTs. To this end, a straight wire was made with the python code. The two wires, along with their respective RT measurements are shown in fig. 2.6. The figure also shows the RT of the wire material created by Blom et al. (see fig. 1.6) as a reference. As we can see, even though straightening the wire has made an improvement, neither RT compares to the reference. Specifically, neither wire shows zero resistance at base temperature. The issue remains.

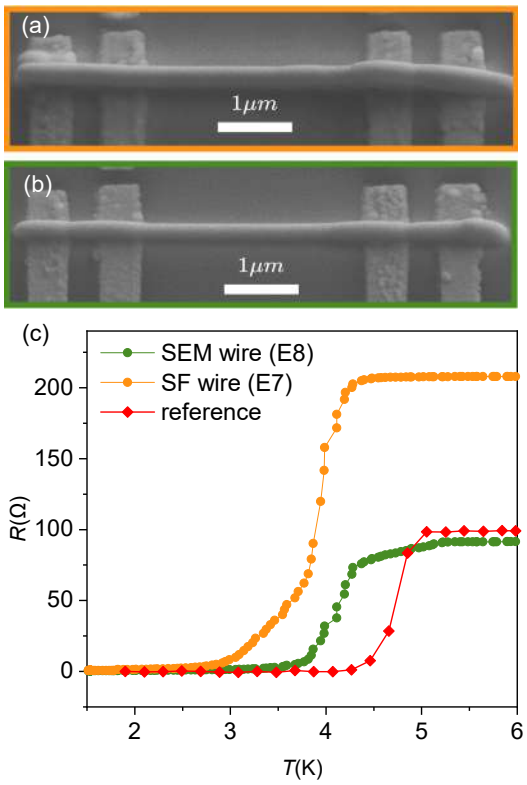

Figure 2.7: *(a) SF patterned EBID wire. (b) SEM patterned EBID wire. (c) Resistance temperature measurements of wires shown in (a) and (b). Reference is the wire in fig. 1.6.*

Subsequent to this conclusion, it was questioned whether our python code itself was flawed, by printing a wire using the software in the SEM. This wire, with RT measurement, can be seen in fig. 2.7 in comparison to an SF patterned wire, as well as the original reference from Blom et al.. Similarly to the previous test, although there seems to be an improvement, this new wire also does not reach zero resistance. We must conclude that the pattern is not causing the problem.

The next test regarded the printing direction of the wire, relative to the gas flow. In fig. 2.8(a) and (b), a closeup of two wires can be seen. In one, the scanning direction is perpendicular to the direction of gas flow. In the other this is parallel. We notice from the closeup that in the perpendicular case, the structure of the wire is smooth and consistent. For the other wire, however, this is not so. We can distinguish alternating light and dark areas, indicating more, and less metallic regions respectively. All the devices and wires in succession to that of fig. 2.2 have been printed in such a parallel manner. Could this turn out to be affecting our transport behaviours?

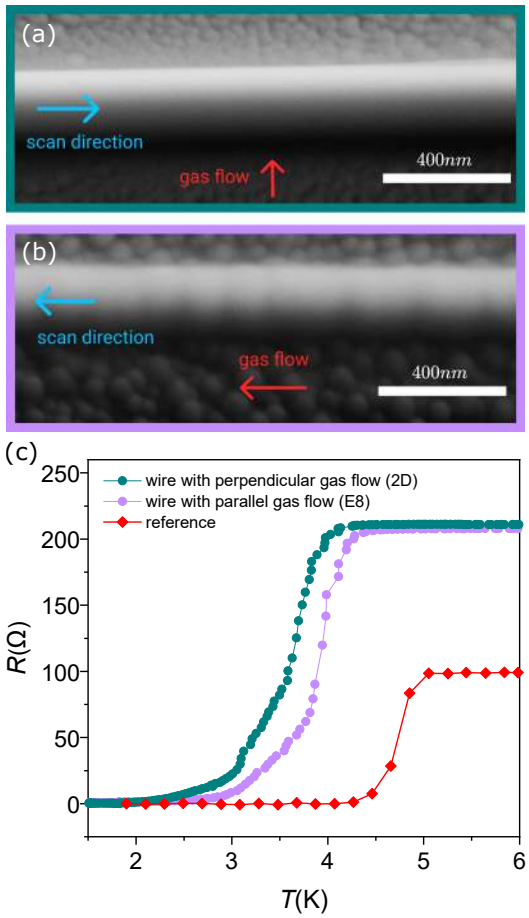

Figure 2.8: *(a) EBID wire scanned perpendicular to gasflow. (b) EBID wire scanned parallel to gas flow. (c) Resistance temperature measurements of wires shown in (a) and (b). Reference is the wire in fig. 1.6.*

Two wires were printed and their RTs compared. The result can be viewed in fig. 2.8(c). Interestingly enough we see no significant improvement for either wire, once again neither is reaching zero resistance.

A full overview of all tests can be seen in fig. 2.9. From the overview, it becomes apparent that our results have been extremely unpredictable, and no trend can be discerned from any of the tests that were performed. None of the wires shows zero resistance at any point. The fact that not a single trend was found in any of the previous tests, shows that there likely is an outside factor, that was never considered, causing this highly irregular behaviour. But what could this be? The question was asked to someone with more technical experience with the EBID process. By them, we were advised to look at the nozzle height. The nozzle height indicates the vertical

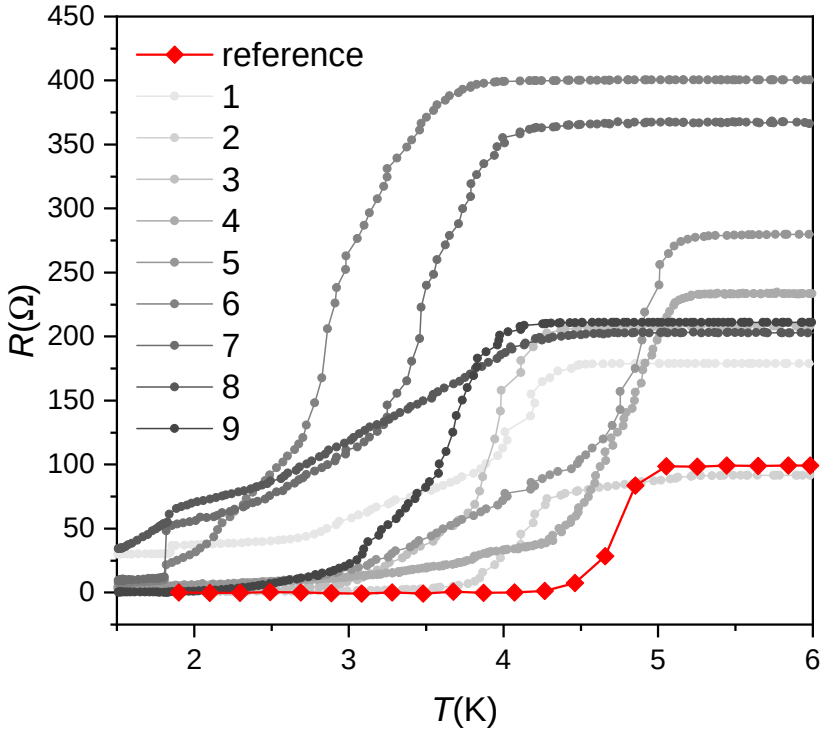

Figure 2.9: *Graph of all RTs that were taking while trying to find the problem of the wires. It is clear that the results show no trend. More importantly, none of the wires presented in this graph (except the reference), reach zero resistance. An overview of the wires is given in tab. 2.1*

	Pattern	Single/Double	Curved/Straight	Print direction	Res (1.5 K)
1	SF	Single	Curved	Parallel	29.9 Ω
2	SEM	Single	Straight	Parallel/Antiparallel	0.5 Ω
3	SF	Single	Sraight	Antiparallel	0.7 Ω
4	SF	Double	Curved	Antiparallel	2.5 Ω
5	SF	Single	Curved	Parallel	4.2 Ω
6	SF	Single	Curved	Parallel	7.2 Ω
7	SF	Double	Curved	Parallel	7.3 Ω
8	SF	Single	Straight	Perpendicular	34.2 Ω
9	SEM	Single	Straight	Perpendicular	0.3 Ω
ref.	SEM	Single	Straight	Perpendicular	0.3 Ω

Table 2.1: *Table of printing strategies for the wires printed, to solve the problem described in sec. 2.3.4. RTs of each wire is shown in fig. 2.9. Data includes origin of the pattern (SF/SEM), whether a double (as in a SQUID), or a single wire was printed (Single/Double), whether the design contains curves (Curved/Straight), the printing direction (parallel/antiparallel/perpendicular), and the resistance at* 1.5 K.

distance between the substrate and the underside of the nozzle. This is an important parameter, as increasing the nozzle height will leave more room for precursor molecules to spread, and thus create a lower local pressure at the substrate. This increases the mean free path length for our particles, affecting the deposition process [11]. For our set-up this nozzle height should be set to $\sim$ 100 µm [12]. This height is set by careful calibration after each time the gas injection system is removed (this would be done, for instance, to refill the precursor gas). To check the distance, the stage was set to height, like was done for the entirety of this research (see 2.2.2). Then, the stage was slowly raised in 5 µm increments, until the substrate touched the gas nozzle[6]. The nozzle height was found to be $\sim$ 500 µm, about five times more than it should be!

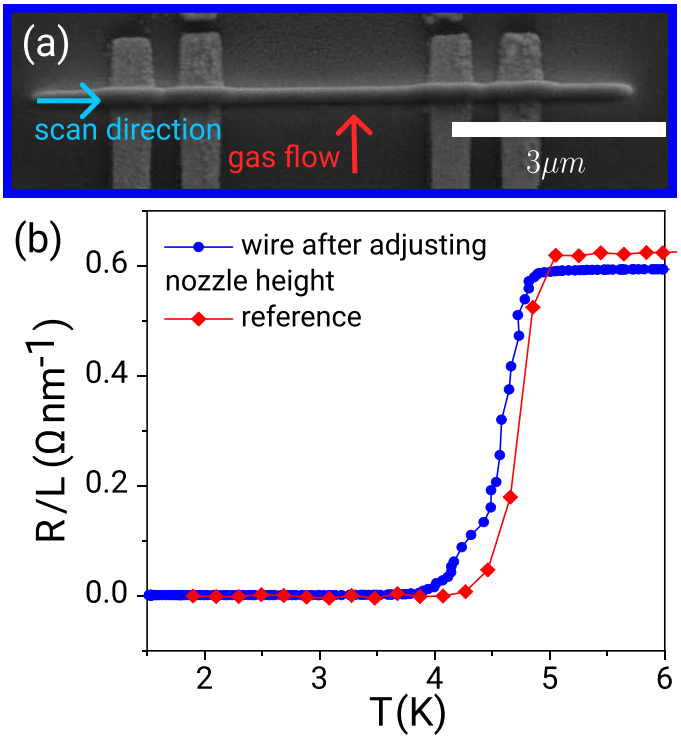

Figure 2.10: *(a) EBID wire after adjusting the nozzle height. (b) Resistance temperature measurements of wire shown in (a). Reference is the wire in fig. 1.6. Both curves are normalised to the length of the wire.*

in succession to this finding, a new wire was printed with an adjusted nozzle height. This wire, along with its transport measurement, can be

[6]When substrate and nozzle touch, a new ground path is created, through which the image is visibly distorted

Version of July 9, 2021– Created July 9, 2021 - 16:58

seen in fig. 2.10, in comparison to the reference. Both curves are normalised to the lengths of the wires. We see that the new wire shows a sharp transition around a similar critical temperature as that of reference. We also identify an extremely similar resistivity. At last, we appear to have found our culprit!
It is to be noted that this wire does show $0.3\,\Omega$ resistance at 1.5 K. This resistance cannot currently be explained, but due to the overall striking similarity between the two RTs in 2.10, as well as the time scope of this project, it was decided to move forward nevertheless.

2.4 Final results

The discoveries made in sec. 2.3.4 left just barely enough time in this project to have one final attempt at producing a functioning SQUID. The results of this attempt can be seen in fig. 2.11 (a). In fig. 2.11(b) then, we see its field measurement. By the oscillations of the critical current with respect to the magnetic field, we can conclude we have, albeit at the very last moment, created a functional SQUID!

In the following section we will analyse these results and put them in comparison to the SQUIDs produced by van den Berg. It is to be noted that only a few characteristics have quantitatively been calculated, due to the time constraints of this project.
The first thing to mention is the period of oscillation, which is about 10 mT. Because the period is always a flux quantum, we know the effective area of our SQUID to be $0.27\,\mu m^2$. An indication of such an area is highlighted in fig. 2.11(c). A second thing apparent from our SQUID pattern, is a slight asymmetry between the oscillations for positive and negative currents. This indicates an asymmetry in inductance between the wires, as is explained in [8]. In this specific case, such an asymmetry is not surprising. As we can see in fig. 2.11(c), the top wire is slightly thicker than the bottom one. This would account for a difference in inductance.
The next characteristic that we will look at is the critical current. From our oscillation pattern, we can discern a maximum critical current of $\sim 5\,\mu A$. This is significantly lower than the critical current of any previous SQUID (which would typically be at least $10\,\mu A$). As we can see from our pattern, in some areas the critical current completely vanishes, which is not ideal. Fortunately, Blom et al. have shown the ability to increase this critical current, by making the weak links more metallic (this can be done by increasing the dwell time). Thus, in consecutive iterations, we expect to increase our critical current with this relatively easy method. This also

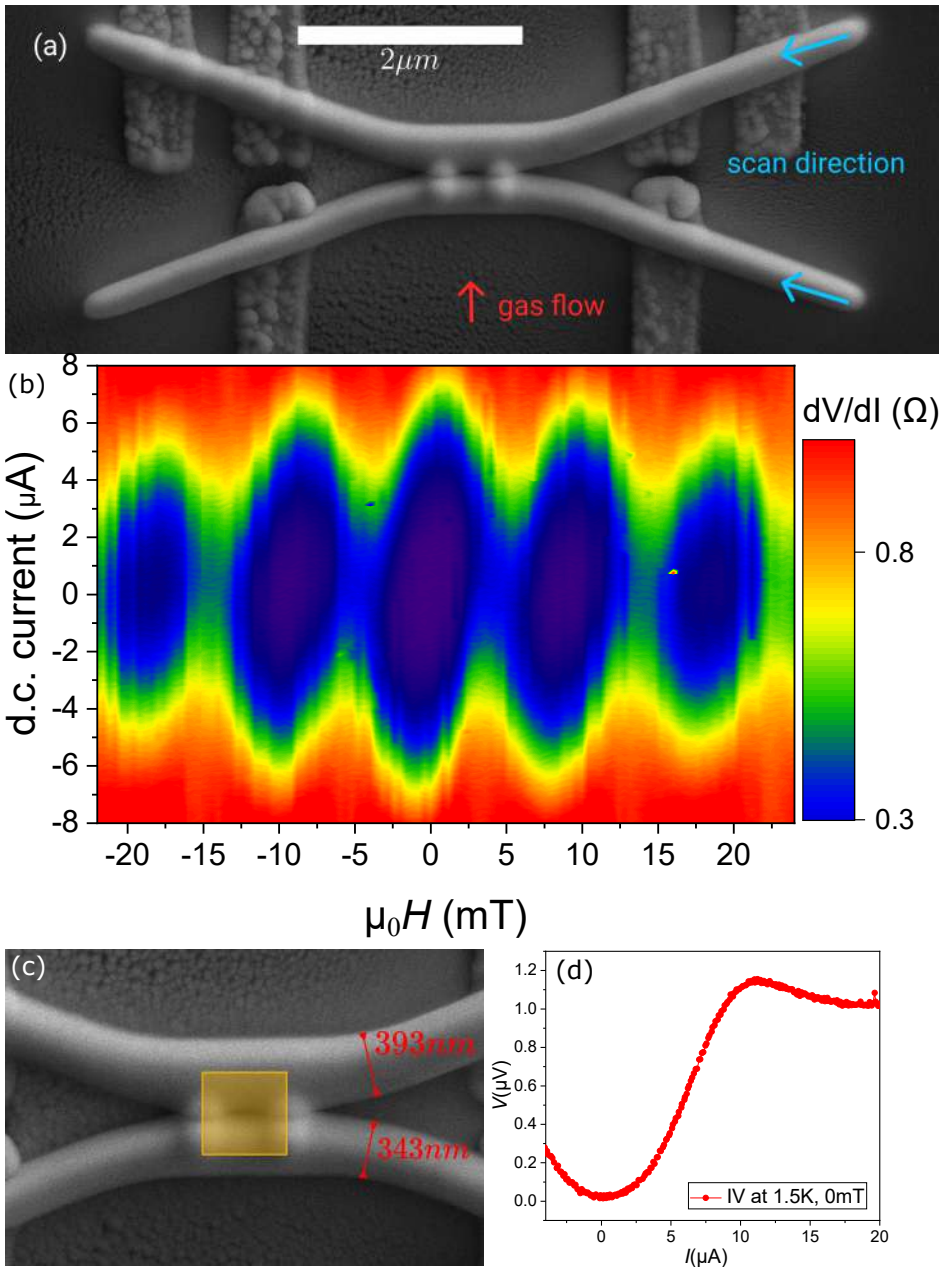

Figure 2.11: *(a) SEM image of SQUID device. Red arrow shows gas flow direction, and blue the scanning direction. (b) Field measurement of SQUID in (a). Critical current is oscillating due to change in magnetic field, with a period of* $\sim$ 10 mT.*(c) Closeup of SQUID in (a). Estimated effective area is indicated, as well as wire thicknesses. (d) Voltage-current characteristic of device shown in (a) at* 1.5 K *and* 0 mT*. IV shows zero resistance.*

links to the following aspect of our pattern: modulation depth. The modulation depth measures how the minimum critical current compares to the maximum critical current, and is expressed in the $\beta_L = \frac{2I_cL}{\phi_0}$ parameter [8], which links to critical current and inductance. Optimally, we would have $\beta_L = 1$ [8], corresponding to a minimum critical current, half the value of the maximum. In our case we see a modulation depth exceeding the entire critical current, which indicates $\beta_L < 1$. Luckily, as can be seen from its formula, β_L is linked to the critical current, thus, by increasing the critical current, we will be able to tune β_L upwards to the optimum of 1.
Another interesting observation to make is that the SQUID reaches zero resistance. This can be seen from the IV in fig. 2.11(d)[7]. The resistance that was observed in the wire shown in 2.10 seems to have vanished for this SQUID. This zero resistance stands in contrast to the residual resistances measured by van den Berg for SQUIDS. It is speculated that this has to do with the way the junctions are printed; in the design of van den Berg, superconducting wires were printed end-to-end, whereas in our research they are printed side-to-side. In addition to these observations it has been calculated that $\sim 50\,\mu A$ of current would be needed to offset the oscillation pattern by $\frac{1}{4}\phi_0$. To calculate this value, the following expression for kinetic inductance was used: $L_K(T) = \mu_0\lambda(T)\frac{l}{A}$, where $\lambda(T)$ is the magnetic penetration depth (which was taken to be 830 nm [5]), l the length of wire that would contribute to flux in the SQUID loop, and A, the cross-section of the wire. A full comparison between the characteristics of our SQUID and those of van den Berg can be seen in tab. 2.2.

	New SQUID	Old SQUID
Periodicity	10 mT	
Area	0.27 μm^2	1.4 μm^2
Critical current	$\sim 5\,\mu A$	10 µA - 65 µA
β_L	< 1	2-5
Residual resistance	$0\,\Omega$	0.2 Ω - 0.7 Ω

Table 2.2: *Various characteristics of the SQUID shown in 2.11, compared to characteristics of previous SQUIDs (data taken from van den Berg)*

[7] The actual value is about 15 mΩ, however this is due to thermal rounding of the IV curve

Chapter 3

Conclusions and outlook

3.1 Conclusions

In this thesis, a design was made for an electrically tunable EBID SQUID. The design was based upon three aspects: miniaturisation, simplicity and electric tunability. The research was unfortunately impeded by a, for a long time unexplained, problem, which rendered us unable to reproduce results from previous research. The culprit was found to be the distance between the substrate and gas nozzle, which was about 5 times too great. Eventually, a SQUID was produced (see 2.4), that is ought to satisfy all three conditions set at the start. The SQUID has an area of $0.27\,\mu m^2$, incorporates a modulation line for electric tunability, and is made to be relatively simple to print. This is discussed in more depth in the next section.

3.2 Discussion and outlook

When starting the development process for our EBID SQUIDs, three design principles were made. This paragraph will reflect on those principles, and see how the final result compares.
The first design principle was simplicity, in order to obtain reproducibility and robustness. Unfortunately, due to time constraints, only one SQUID was made, and thus reproducibility could not be tested. That being said, it is obvious that our design, using only two lines of superconductor, is much simpler than the previous, circular design. Moreover, the fact that, once the nozzle height was adjusted, the first SQUID produced, was working, is very promising with regards to the reproducibility of these results.
The second principle was the minimisation of the SQUID area. As dis-

cussed in sec. 2.4, we managed to produce an effective area of $0.27\,\mu m^2$, which is about $\frac{1}{5}$ of the area of the previous design. In this aspect, a major improvement has been made.
The last design principle was the inclusion of a modulation line in order to generate electric tunability. Tough a modulation line was successfully built into the design, this too has unfortunately not been tested due to time constraints.
This brings us to an outlook on the development of these SQUIDs. Namely, the next thing to test for is the electric tunability of the SQUID. To do this we simply send a current through our modulation line, and then take a field measurement similar to that in fig. 2.11(b). The oscillation pattern should now be offset. This experiment is being developed at the time of writing. If electrical tunability can be shown it is of the order to provide a proof of concept experiment, where one of these SQUIDs can be printed atop a magnetic crystal. If the SQUID performs, the design can be tested for reproducibility, sensitivity and noise. Overall, this research has shown a significant improvement in the fabrication of EBID SQUIDs. It showed the production of an EBID SQUID able to do local measurements. Moreover, the design of the SQUID was specifically developed to be both reproducible, and electrically tunable. To measure magnetic fields, such as that of $Sr2RuO4$, only a few steps now remain.

Bibliography

[1] Timothy van den Berg. Direct-write printed superconducting quantum interference devices using focused electron beam induced deposition. Master's thesis, Leiden university, 2021.

[2] Shannon K'doah. *The gravity probe B science instrument*, page 26.

[3] D. Drung, C. Abmann, J. Beyer, A. Kirste, M. Peters, F. Ruede, and T. Schurig. Highly sensitive and easy-to-use squid sensors. *IEEE Transactions on Applied Superconductivity*, 17(2):699–704, 2007. doi: 10.1109/TASC.2007.897403.

[4] Andrew P Mackenzie, Thomas Scaffidi, Clifford W Hicks, and Yoshiteru Maeno. Even odder after twenty-three years: the superconducting order parameter puzzle of sr2ruo4, 2017.

[5] Tycho Blom, Thomas Mechielsen, Remko Fermin, Marcel Hesselberth, J. Aarts, and Kaveh Lahabi. Direct-write printing of josephson junctions in a scanning electron microscope. *ACS Nano*, 15:322–329, 11 2020. doi: 10.1021/acsnano.0c03656.

[6] Yuuki Yasui, Kaveh Lahabi, Victor FernÃ¡ndez Becerra, Remko Fermin, Muhammad Shahbaz Anwar, Shingo Yonezawa, Takahito Terashima, Milorad V. MiloÅ¡eviÄ, Jan Aarts, and Yoshiteru Maeno. Spontaneous emergence of josephson junctions in homogeneous rings of single-crystal sr2ruo4. *npj Quantum Materials*, 5(1), Apr 2020. ISSN 2397-4648. doi: 10.1038/s41535-020-0223-7. URL `http://dx.doi.org/10.1038/s41535-020-0223-7`.

[7] B.D. Josephson. Possible new effects in superconductive tunnelling. *Physics Letters*, 1(7):251–253, 1962. ISSN 0031-9163. doi:

https://doi.org/10.1016/0031-9163(62)91369-0. URL https://www.sciencedirect.com/science/article/pii/0031916362913690.

[8] John Clarke and Alex I. Braginski. *The SQUID Handbook Fundamentals and Technology of SQUIDs and SQUID Systems*, volume 1. Wiley-VCH, Weinheim, 2006. URL http://public.eblib.com/EBLPublic/PublicView.do?ptiID=481572.

[9] Shamashis Sengupta, Chuan Li, Cedric Baumier, Alik Kasumov, S. GuÃ©ron, H. Bouchiat, and F. Fortuna. Superconducting nanowires by electron-beam-induced deposition. *Applied Physics Letters*, 106(4):042601, 2015. doi: 10.1063/1.4906269. URL https://doi.org/10.1063/1.4906269.

[10] W. F. van Dorp and C. W. Hagen. A critical literature review of focused electron beam induced deposition. *Journal of Applied Physics*, 104(8):081301, 2008. doi: 10.1063/1.2977587. URL https://doi.org/10.1063/1.2977587.

[11] K.T. Kohlmann, M. Thiemann, and W.H. BrÃŒnger. E-beam induced x-ray mask repair with optimized gas nozzle geometry. *Microelectronic Engineering*, 13(1):279–282, 1991. ISSN 0167-9317. doi: https://doi.org/10.1016/0167-9317(91)90093-S. URL https://www.sciencedirect.com/science/article/pii/016793179190093S.

[12] A. Botman, M. Hesselberth, and J.J.L. Mulders. Improving the conductivity of platinum-containing nano-structures created by electron-beam-induced deposition. *Microelectronic Engineering*, 85(5):1139–1142, 2008. ISSN 0167-9317. doi: https://doi.org/10.1016/j.mee.2007.12.036. URL https://www.sciencedirect.com/science/article/pii/S0167931707008076. Proceedings of the Micro- and Nano-Engineering 2007 Conference.

Appendix A

Printing parameters

The following table provides an overview of the parameters that were used in printing the devices, after the nozzle height was adjusted. Device G5, and device G7 are shown in the images below. Device G8 and device G6 are those displayed in fig. 2.10(a) and 2.11(a) respectively.

Type of structure	Detector mode	substrate#	HV(kV)	Beam(nA)	dwell (ms)	passes	time(m:s)	Pitch (nm)	scan direction	Dimensions	Overlap	Pstart (mBar)	Pend (mBar)	Remarks
SEM wire	standard	2b	10	20	25	1	05:50	1nm	perpendicular	7µm	98%	2.5E-6	1.37E-5	perpendicular gasflow
SQUID wires	standard	2b	10	20	25	2	12:30	0.93nm	perpendicular	gap:1700py	98%	2.1E-6	1.2E-5	perpendicular gasflow
SQUID weaklinks	standard	2b	10	20	4	60	00:30	0.02µm	Left to right	0.15µm x 0.25µA	60%	2.1E-6	1.2E-5	
SQUID wires	standard	2b	10	20	25	2	12:30	0.93nm	perpendicular	gap:1600	98%	2.03E-6	1.18E-5	perpendicular gasflow
SQUID weaklinks	standard	2b	10	20	4	60	00:30	0.02µm	Left to right	0.15µm x 0.25µA	60%	2.03E-6	1.18E-5	
SQUID wires	standard	2b	10	20	25	2	12:30	0.93nm	parallel	gap:1500py	98%	1.99E-6	1.15E-5	parallel gasflow
SQUID weaklinks	standard	2b	10	20	4	60	00:30	0.02µm	Left to right	0.15µm x 0.25µm	60%	1.99E-6	1.15E-5	

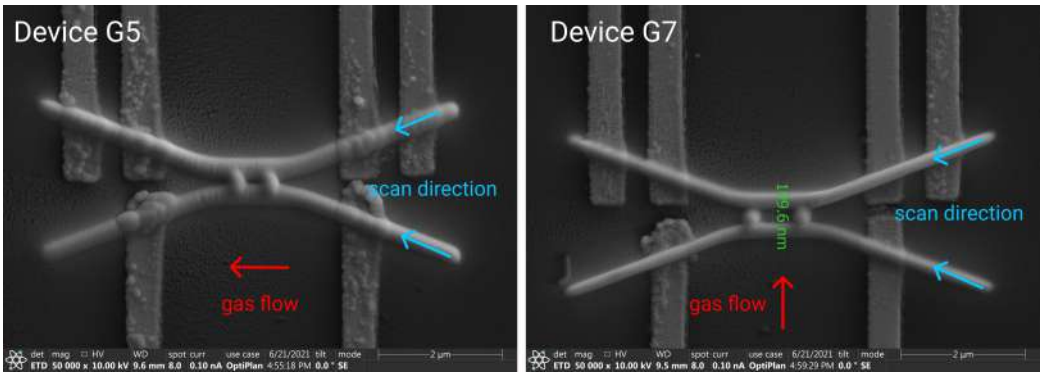

Figure A.1: *Table of printing parameters, and SEM images of devices G5 and G7. The unit "py" refers to an python code unit and relates to roughly* 0.31 nm. *Dimensions given in py units refer to the value that was used for a variable in the code, and do not necessarily relate to actual distances.*

Appendix B

Python code

```
def wire(params = None):
    '''
    ########################################
    '''
    #set parameters (default or specified)
    default = {'length':9000,'var':'n'}

    file_list = []

    dwelltime = "250008 "
    stepsize = 3

    pos_x = 0
    pos_y = 10000

    length = params['length'] if (params and params.get('length', False)) else default['length']
    var = params['var'] if (params and params.get('var', False)) else default['var']

    i = 0

    while pos_x < length:
        #normal variation
        if var == 'n':
            file_list.append(dwelltime+str(round(pos_x))+" "+str(round(pos_y))+" \n")
        #rotated by 90 dergees
        elif var == 'r90':
            file_list.append(dwelltime+str(round(pos_y))+" "+str(round(pos_x))+" \n")
        #rotated by 90 degrees and flipped
        elif var == 'r90flip':
            file_list.append(dwelltime+str(round(pos_y))+" "+str(round(-pos_x+length))+" \n")

        pos_x += stepsize

        i = i + 1

    return file_list

# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% #
def parallel_squid(params = None):
    '''
                        (top)
 ##############################################
                                            (bottom)
          ################################################

    '''
    # set parameters (default or specified)
    file_list = []
    default = {'gap':1100, 'l_wires':9000, 'offset_bottom':4000, 'var':'n'}

    gap = params['gap'] if (params and params.get('gap', False)) else default['gap']
    l_wires = params['l_wires'] if (params and params.get('l_wires', False)) else default['l_wires']
    offset_bottom = params['offset_bottom'] if (params and params.get('offset_bottom', False)) else default['offset_bottom']
    var = params['var'] if (params and params.get('var', False)) else default['var']

    pos_x_bottom = offset_bottom
```

```
    pos_y_bottom = 0

    pos_x_top = l_wires
    pos_y_top = gap

    while pos_x_top > 0 :

        # normal variation
        if var == 'n':
            file_list.append("250008 "+str(round(pos_x_top))+" "+str(round(pos_y_top))+" \n")
            file_list.append("250008 "+str(round(pos_x_bottom))+" "+str(round(pos_y_bottom))+" \n")
        # rotated by 90 degrees
        elif var == 'r90':
            file_list.append("250008 "+str(round(pos_y_top))+" "+str(round(pos_x_top))+" \n")
            file_list.append("250008 "+str(round(pos_y_bottom))+" "+str(round(pos_x_bottom))+" \n")

        pos_x_bottom += 3
        pos_x_top -= 3

    return file_list
# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% #
def parallel_squid_3(params = None):

    '''
 ####                                                         ####
     ####                                                 ####
         ####                                         ####
             ####                                 ####
                 ####         <- 1          ####
                     #########################

                                 <-2
                     #########################
                 ####                         ####
             ####                                 ####
         ####                                         ####
    ####                                                  ####
####                                                          ####
    '''
    #set given parameters (default or specified)
    file_list = []
    default = {'gap':1000, 'lp':3000, 'ld':2000, 'alpha': 30, 'curve':00, 'var':'n'}
    stepsize = 3
    dwelltime = "250008 "

    gap = params['gap'] if (params and params.get('gap', False)) else default['gap']
    lp = params['lp'] if (params and params.get('lp', False)) else default['lp']
    ld = params['ld'] if (params and params.get('ld', False)) else default['ld']
    curve = params['curve'] if (params and params.get('curve', False)) else default['curve']
    alpha = params['alpha'] if (params and params.get('alpha', False)) else default['alpha']
    var = params['var'] if (params and params.get('var',False)) else default['var']

    alpha = math.radians(alpha)

    #define start positions
    x1,y1 = lp+2*ld*math.cos(alpha), gap+2*ld*math.sin(alpha)
    x2,y2 = lp+2*ld*math.cos(alpha), 0

    shift = x1

    #first sloping part
    while x1 > ld*math.cos(alpha) + lp + curve*math.cos(alpha):
        #normal variation
        if var == 'n':
            file_list.append(dwelltime+str(round(x1))+" "+str(round(y1))+" \n")
            file_list.append(dwelltime+str(round(x2))+" "+str(round(y2))+" \n")
        #rotated by 90 degrees
        elif var == 'r90':
            file_list.append(dwelltime+str(round(y1))+" "+str(round(x1))+" \n")
            file_list.append(dwelltime+str(round(y2))+" "+str(round(x2))+" \n")
        #rotated by 90 degrees and flipped in printing direction
        elif var == 'r90flip':
            file_list.append(dwelltime+str(round(y1))+" "+str(round(-x1+shift))+" \n")
            file_list.append(dwelltime+str(round(y2))+" "+str(round(-x2+shift))+" \n")

        x1 -= stepsize*math.cos(alpha)
        y1 -= stepsize*math.sin(alpha)
        x2 -= stepsize*math.cos(alpha)
        y2 += stepsize*math.sin(alpha)

    #while x1 > ld*math.cos(alpha) + lp - curve:

    #first curve
```

```
    n_steps = int(curve*alpha/(math.tan(alpha/2)*stepsize))
    angles = np.linspace(alpha, 0, n_steps)

    for angle in angles:

        if var == 'n':
            file_list.append(dwelltime+str(round(x1))+" "+str(round(y1))+" \n")
            file_list.append(dwelltime+str(round(x2))+" "+str(round(y2))+" \n")
        elif var == 'r90':
            file_list.append(dwelltime+str(round(y1))+" "+str(round(x1))+" \n")
            file_list.append(dwelltime+str(round(y2))+" "+str(round(x2))+" \n")
        elif var == 'r90flip':
            file_list.append(dwelltime+str(round(y1))+" "+str(round(-x1+shift))+" \n")
            file_list.append(dwelltime+str(round(y2))+" "+str(round(-x2+shift))+" \n")

        x1 -= stepsize*math.cos(angle)
        y1 -= stepsize*math.sin(angle)
        x2 -= stepsize*math.cos(angle)
        y2 += stepsize*math.sin(angle)

    #parallel part
    while x1 > ld*math.cos(alpha) + curve:

        if var == 'n':
            file_list.append(dwelltime+str(round(x1))+" "+str(round(y1))+" \n")
            file_list.append(dwelltime+str(round(x2))+" "+str(round(y2))+" \n")
        elif var == 'r90':
            file_list.append(dwelltime+str(round(y1))+" "+str(round(x1))+" \n")
            file_list.append(dwelltime+str(round(y2))+" "+str(round(x2))+" \n")
        elif var == 'r90flip':
            file_list.append(dwelltime+str(round(y1))+" "+str(round(-x1+shift))+" \n")
            file_list.append(dwelltime+str(round(y2))+" "+str(round(-x2+shift))+" \n")

        x1 -= stepsize
        x2 -= stepsize

    #second curve
    for angle in np.flip(angles, 0):

        if var == 'n':
            file_list.append(dwelltime+str(round(x1))+" "+str(round(y1))+" \n")
            file_list.append(dwelltime+str(round(x2))+" "+str(round(y2))+" \n")
        elif var == 'r90':
            file_list.append(dwelltime+str(round(y1))+" "+str(round(x1))+" \n")
            file_list.append(dwelltime+str(round(y2))+" "+str(round(x2))+" \n")
        elif var == 'r90flip':
            file_list.append(dwelltime+str(round(y1))+" "+str(round(-x1+shift))+" \n")
            file_list.append(dwelltime+str(round(y2))+" "+str(round(-x2+shift))+" \n")

        x1 -= stepsize*math.cos(angle)
        y1 += stepsize*math.sin(angle)
        x2 -= stepsize*math.cos(angle)
        y2 -= stepsize*math.sin(angle)

    #second sloping part
    while x1 > 0:

        if var == 'n':
            file_list.append(dwelltime+str(round(x1))+" "+str(round(y1))+" \n")
            file_list.append(dwelltime+str(round(x2))+" "+str(round(y2))+" \n")
        elif var == 'r90':
            file_list.append(dwelltime+str(round(y1))+" "+str(round(x1))+" \n")
            file_list.append(dwelltime+str(round(y2))+" "+str(round(x2))+" \n")
        elif var == 'r90flip':
            file_list.append(dwelltime+str(round(y1))+" "+str(round(-x1+shift))+" \n")
            file_list.append(dwelltime+str(round(y2))+" "+str(round(-x2+shift))+" \n")

        x1 -= stepsize*math.cos(alpha)
        y1 += stepsize*math.sin(alpha)
        x2 -= stepsize*math.cos(alpha)
        y2 -= stepsize*math.sin(alpha)

    return file_list
# %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% #
# Visualisation of a streamfile
def vis_streamfile(filename):

    # get lines from file, and skip first few
    file = open(filename,"r+")

    lines = file.readlines()[4:]
    file.close()
    x_coords, y_coords = [],[]
```

```
    # extract x and y coords from file lines
    for line in lines:
        line = line.split()
        x_coords.append(int(line[1]))
        y_coords.append(int(line[2]))

    # plot coords using colormap
    plt.figure()
    plt.grid()
    plt.title(f"Visualisation of {filename}")
    plt.xlabel('x')
    plt.ylabel('y')
    plt.axis("equal")
    color = np.arange(0,len(x_coords))
    plt.scatter(x_coords, y_coords, c=color, cmap = plt.get_cmap('cividis').reversed())
    plt.colorbar(ticks = [0,len(x_coords)-100]).ax.set_yticklabels(['start','end'])
    plt.show()

# create a file list based on given command
def create_filelist(structure, params = None):
    file_list = None

    # call appropriate function to create file list with given parameters
    if structure == "wire":
        file_list = wire(params = params)
    if structure == "parallel_squid":
        file_list = parallel_squid(params = params)
    if structure == "parallel_squid_3":
        file_list = parallel_squid_3(params = params)
    # add new structures here

    if file_list == None:
        raise Exception("Structure name is not recognised")

    return file_list

# create a streamfile from a filelist
def create_streamfile(file_list):
    datenow = datetime.datetime.now()
    M = datenow.strftime("%m")
    D = datenow.strftime("%d")
    Y = datenow.strftime("%y")
    H = datenow.strftime("%H")
    Min = datenow.strftime("%M")

    loops = 1
    bits = 16
    fieldx = 2498
    fieldy = 2218

    filename = str("StreamfileImport_time"+str(H)+"_"+str(Min)+"_D"+str(D)+"_M"+str(M)+"_Y"+str(Y)+".str")
    print("\nFilename: \t \t \t \t"+str(filename))
    file1 = open(filename,"w+")
    file1.write( "s"+str(bits)+" \n")
    file1.write(""+str(loops)+" \n")
    rows_number =  len(file_list)+1
    file1.write(str(rows_number)+" \n")
    file1.write(str(bits)+" "+str(fieldx)+" "+str(fieldy)+"\n")

    for i in range(len(file_list)):
        file1.write(str(file_list[i]))

    file1.close()

    filename_read = filename
    file_read = open(filename_read,"r+")
    file_r = file_read.readlines()
    file_read.close()
    if (len(file_r)-3) == rows_number:
        print("Length file: \t \t \t \t"+str(rows_number-1))

    "calculation build time"

    time_calc = 0
    weaklink = 0
    for i in range(0, len(file_list)):
        time_r = file_list[i].split()
        time_calc = float(time_calc) + float(time_r[0])
        #print(float(time_calc))
        if int(time_r[0]) == int(4000):
            weaklink = 1
```

```
    time_sec_total = time_calc/10000000
    time_total_min = math.floor(time_sec_total/60)
    time_sec = (time_sec_total/60 - time_total_min)*60
    print("Expected building time: \t"+str(time_total_min)+":"+str(round(time_sec)))

    print('\nfile is closed')

    return filename

def main():
    # check for proper usage
    if len(sys.argv) != 3 and len(sys.argv) != 4:
        print("Usage:run SQUID_code [structure] [parameters]")
        print("\t parameters: use -d for default values, or provide a dict \n\t with name value pairs for parameters (see documentation)")
        print("\t add -v at end to create visualisation \n\t of the structure")
        sys.exit()

    # extract structure name from command line
    structure = sys.argv[1]

    # extract parameters from command line
    if sys.argv[2] == '-d':
        params = None
    else:
        params = ast.literal_eval(sys.argv[2])

    # create a file list of correct structure type with parameters
    file_list = create_filelist(structure, params = params)

    # create a stream-file from file list
    filename = create_streamfile(file_list)

    # create visualisation if specified
    if len(sys.argv) >= 4 and sys.argv[3] == "-v":
        vis_streamfile(filename)

main()
```