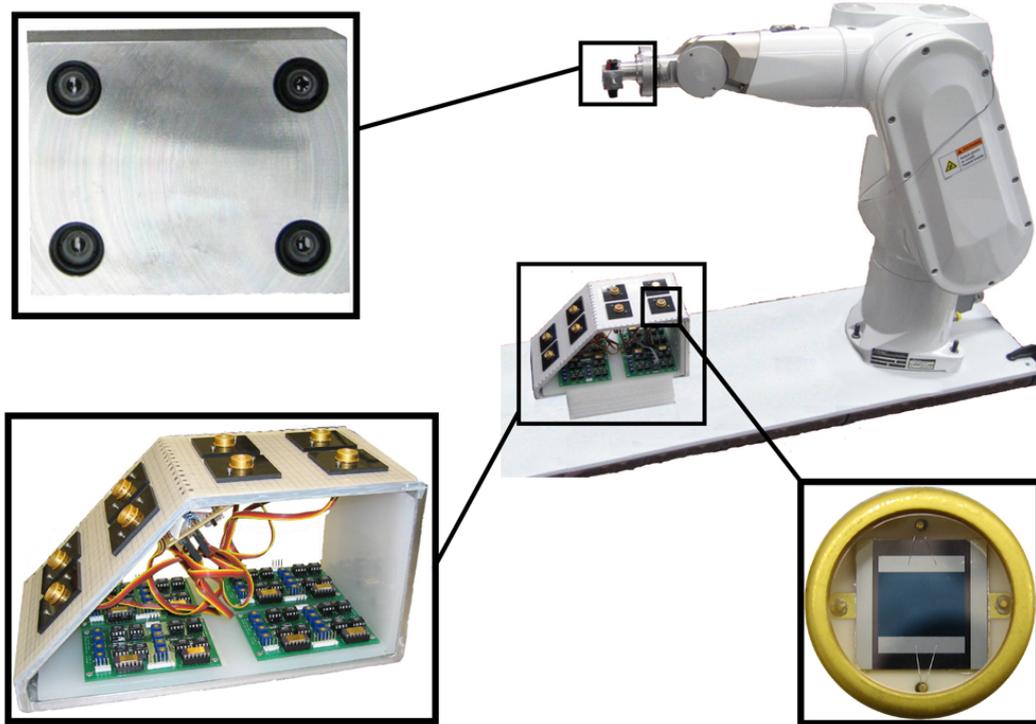


ROBOTICS RESEARCH LABORATORY
DEPARTMENT OF COMPUTER SCIENCE
UNIVERSITY OF KAISERSLAUTERN

Student Research Project



Robot Calibration based on
Position Sensitive Devices

Sebastian Blank

January 29, 2007

Robot Calibration based on Position Sensitive Devices

Student Research Project

The Robotics and Automation Laboratory
Department of Electrical and Computer Engineering
Michigan State University

Sebastian Blank

Topic issued : 08/28/2006
Day of release : 12/15/2006

Supervisors : Prof. Dr. Ning Xi, Dr. Yantao Shen
Author : Sebastian Blank

I hereby declare that I have self-dependently composed the student research project report at hand. The sources and additives used have been marked in the text and are exhaustively given in the bibliography.

January 25, 2007 – Kaiserslautern

(Sebastian Blank)

Acknowledgements

I would like to take this opportunity to thank my supervising professor Dr. Ning Xi for the chance to perform this research project at the Robotics and Automation Laboratory at Michigan State University. I would also like to thank my supervisor Dr. Yantao Shen for the time and knowledge he devoted to support my research and his contribution to this project. Furthermore I would like to thank all of my lab-mates and especially M.Sc. Quan (Mike) Shi for his help and advice. Last but not least I would like to show my appreciation for the staff involved in the exchange program at Michigan State University and University of Kaiserslautern for their support before and during my stay at MSU.

Contents

1	Introduction	7
1.1	Motivation	7
1.2	Calibration Task	7
1.3	Project Objectives	9
2	Position Sensitive Devices	11
2.1	Fundamentals	11
2.1.1	Segmented PSDs	13
2.1.2	Lateral PSDs	14
2.2	PSD Output Processing Board	15
2.2.1	Board Design	15
2.2.2	Board Output	15
2.2.3	Parameter Tweaking	18
2.3	Experimental Results	20
2.3.1	Lateral PSD Results	20
2.3.2	Segmented PSD Results	26
2.3.3	Result Evaluation	27
3	Robot Control	31
3.1	Feedback Mapping	31
3.2	Feedback Test	33
3.3	Feedback Evaluation	35
3.4	Controller Design	36
3.4.1	PD plus Gravity Control Law	36
3.4.2	PSD-Servoing	38
3.5	Controller Simulation	42
4	The Calibration System	45
4.1	Concept	46
4.1.1	Virtual Intersection Points	47
4.1.2	Laser Beam Intersection points	49
4.1.3	Coordinate Transformation	51
4.1.3.1	General Coordinates Transformation	51
4.1.3.2	Quaternion-Based Algorithm	52
4.2	Calibration Algorithm	55

4.2.1	Overview	55
4.2.2	Positioning	56
4.2.3	Calibration	57
4.2.4	Result Test	58
4.3	Calibration System Hardware	59
4.3.1	Laser fixture	59
4.3.2	Calibration Fixture	60
4.4	Error Consideration and Discussion	62
4.4.1	Sources of Error	62
5	Conclusion & Outlook	65
5.1	Conclusion	65
5.2	Outlook on future Work	65
A	Table of Abbreviations	67
B	Drafts	69
B.1	Calibration fixture	69
B.2	Laser fixture	71
B.3	PSD output processing board	71
C	Tables and Figures	73
C.1	Board Resistor Value Determination	73
	References	74

1. Introduction

1.1 Motivation

Calibration of robots has become a research field of great importance over the last decades especially in the field industrial robotics. The main reason for this is that the field of application was significantly broadened due to an increasing number of fully automated or robot assisted tasks to be performed. Those applications require significantly higher level of accuracy due to more delicate tasks that need to be fulfilled (e.g. assembly in the semiconductor industry or robot assisted medical surgery). In the past, (industrial) robot calibration had to be performed manually for every single robot under lab conditions in a long and cost intensive process. Expensive and complex measurement systems had to be operated by highly trained personnel. The result of this process is a set of measurements representing the robot pose in the task space (i.e. world coordinate system) and as joint encoder values. To determine the deviation, the robot pose indicated by the internal joint encoder values has to be compared to the physical pose (i.e. external measurement data). Hence, the errors in the kinematic model of the robot can be computed and therefore later on compensated.

These errors are inevitable and caused by varying manufacturing tolerances and other sources of error (e.g. friction and deflection). They have to be compensated in order to achieve sufficient accuracy for the given tasks. Furthermore for performance, maintenance, or quality assurance reasons the robots may have to undergo the calibration process in constant time intervals to monitor and compensate e.g. ageing effects such as wear and tear.

In modern production processes old fashioned procedures like the one mentioned above are no longer suitable. Therefore a new method has to be found that is less time consuming, more cost effective, and involves less (or in the long term even no) human interaction in the calibration process.

1.2 Calibration Task

To solve this dilemma a new kind of calibration system is needed. In this project an optical approach based on PSDs (i.e. *Position Sensitive Devices*) was chosen

to improve the calibration process in respect to time, reliability, and costs aspects. Another important facet of improvement is to reduce or even eliminate (in the long term) the need for human participation in the process. This is due to the fact that it is both expensive and error-prone.

As shown in [Shiakolas 02] industrial robots in general feature good repeatability but rather poor accuracy. This is derived from the fact that the major share of error contribution ($> 90\%$) is of non-dynamic nature such as e.g. the joint zero position error. These static errors could be avoided by initially calibrating the robot. For this purpose a system has to be designed and manufactured that allows it to determine the robots individual positioning inaccuracy. The idea of the optical system proposed is to obtain a robot specific representation (joint encoder values) for a given set of points in the robot's task space. Since these points are already given in an external representation (i.e. world coordinate system [WCS]) the robot base frame can be related to the task space (i.e. world frame). Hence a higher positional accuracy can be achieved using this method. This is based on the fact that the desired pose in the task space can be directly transferred into joint variables in the robot's kinematic model. The system designed for this project consists of two parts: A fixture to hold the laser pointers at the robot arm (referred to as "laser fixture") and a fixture that is positioned in the task space, housing the PSDs (referred to as "calibration fixture").

As a first step, the four laser beams mounted in the proximity of the TCP of the robot are aligned with one surface area of the fixture device at a time. At least three out of the four beams have to be centered on the four PSDs mounted on one surface at a time to be able to perform the calibration. This requirement originates from the fact that the problem dealt with here involves 6 degrees of freedom (i.e. 3 DOF Cartesian position, 3 DOF rotation). When a position is reached that satisfies the criteria, the current TCP position (based on the robots joint encoder values and the robot's kinematic parameters) is stored and the procedure continues with the next surface area. This is done until all the areas of the fixture device are processed.

After that the matching points in either representation (robot base frame [RBCS] and WCS) have to be constructed. Therefore lines based on the recorded data are constructed and intersected to receive the matching intersection points that will be employed to relate both representations to each other and therefore perform the calibration. For our purposes, a laser beam can be modeled as a straight line with the position of the respective laser device mounted in the proximity of the TCP as origin (known in the laser fixture coordinate system [LFCS]) and the TCP orientation (stored orientation) as beam orientation. Hence, the virtual laser beam intersection of two beams can be determined by intersecting two lines at a time to compute the respective laser beam intersection point in TCP coordinate frame. On the other hand all distances and angles of the fixture device are known. Following the same pattern, the virtual intersection points of the PSDs can be determined as well. This time the center of the PSD's active area serves as the line origin and the orientation can be determined by the known angles in between the surface area parts of the fixture. Therefore two representations for the same (intersection) points are now obtained.

Since the target is to relate the robot base to the task space, the laser beam intersection points have to be transferred from the laser fixture frame into robot base

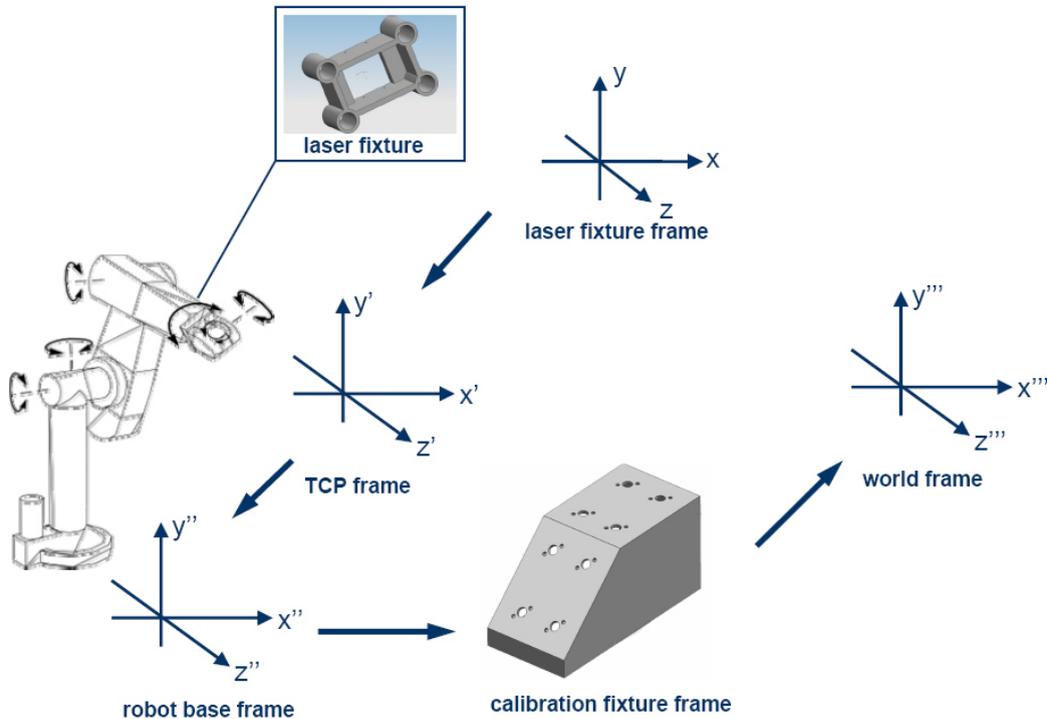


Figure 1.1: Schematic of the calibration system developed in this project. The laser fixture is mounted to the robot arm close to the TCP and can hold up to four laser pointers. The laser beams are projected on the calibrations fixture housing the PSDs. The robot arm needs to be repositioned based on the data acquired from the calibration fixture feedback (PSD positional feedback) until at least three beams are centered on the PSD surfaces. After this is done, the gathered data is processed and the resulting points are transferred from the laser calibration frame to the world frame.

frame. Once this transformation is performed the relation between the robot base and the task space can be established and hence the calibration operation is performed successfully.

1.3 Project Objectives

This section is intended to give the reader an idea about the tasks of this first stage of the robot calibration project and to eliminate any misconceptions in the first place. First of all let's point out that this report only covers the work of the first phase of this project. It is particular important to understand that the presented system does not at all claim to be a finished product, that is "ready to use". In fact it is intended to be a technical feasibility evaluation performed by Michigan State University's Robotics and Automation Laboratory in cooperation with Asea Brown Boveri (ABB).

The long term motivation for this is to be able to replace calibration technologies used today that have some undesirable side effects (discussed in section 4). In this particular case an alternative to an old fashioned but reliable system using physical

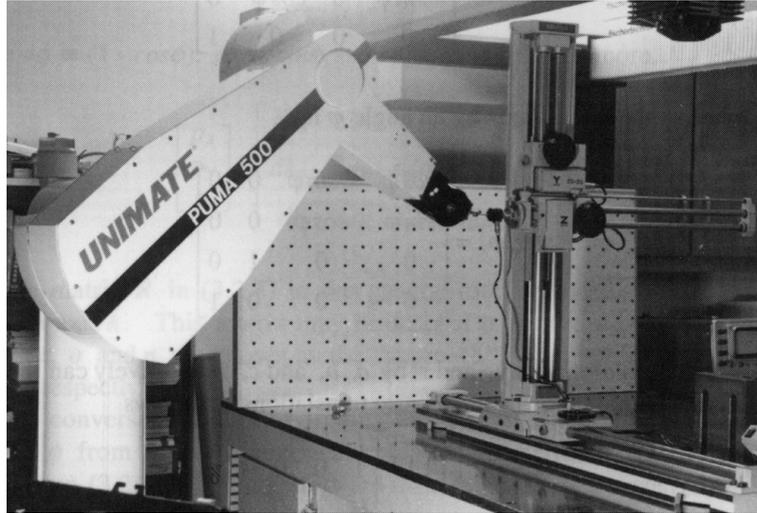


Figure 1.2: *Image taken from: [Zhuang 96, page 68]:* Example of the classical calibration fixture as still in use today. The robot arm has to be moved manually into a position that allows the metal pin attached to the TCP to touch a second metal pin fixed in the robot's task space. This has to be performed multiple times using different joint configurations to achieve the required precision. This method is both time consuming and involves a lot of manual interaction by a trained operator.

contact as shown in figure 1.2 is to be found. The optical approach using PSDs was chosen since it seems to promise at least equal if not better precision without any physical contact or the need for human involvement in the calibration process. The expectations towards this first stage of the project are to design and manufacture a system consisting of PSDs (calibration fixture) and laser devices (laser fixture) and show that calibration can be performed using this method. However it is evident, that the results will not live up to the full potential of this technology since if it proves to be successful additional research has to be performed to improve the outcome since no research performed by other groups on a system like the one proposed here could be found anywhere. Therefore the obtained results have to be analyzed and evaluated to give hints on what parts of the process need to be enhanced.

The main topic of this report is the initial design stage from scratch on. Chapter 2 will focus on PSDs. Basic properties and working principles will be discussed and the hardware designed to process will be presented. Further the experiments using the PSD sensor and the designed signal processing board will be presented and discussed. In chapter 3 the controller moving the robot arm during the calibration process will be deduced and the principle of visual servoing will be transferred to be used in connection with the PSD. In chapter 4 the focus will be directed to the calibration system itself. Here the basic concept is going to be presented in detail and occurring problems will be discussed.

2. Position Sensitive Devices

2.1 Fundamentals

A position sensitive device (PSD) is a sensor capable of tracking the location of a light beam on its surface. It basically consists of either one or two resistive layers (one-dimensional or two-dimensional PSD) placed on the surface of a high-resistive substrate (see figure 2.1 for basic PSD schematics)¹.

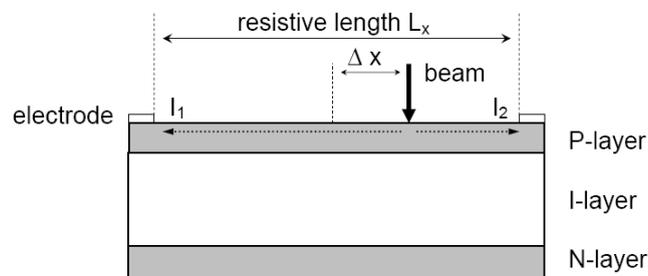


Figure 2.1: Schematics of a one dimensional PSD chip. The device consists of three semiconductor layers of which only the top one is used to determine the position. In two dimensional devices the bottom layer is used in an analog manner to gather positional information.

The operation principle of a PSD is quite simple: If the top P-layer is stimulated with a beam emitted from a light source, an electric charge is generated that is proportional to the light intensity. This formed potential in the resistive layer causes photo-currents to flow between the spot of stimulation and the two electrodes on either end of the layer. Due to the uniformity of the resistive layer I_1 and I_2 are

¹ More information on PSD than provided in the next sections (section 2.1.1, section 2.1.2) can be found in [Corporation 06], [Optronics 06], and [Photonics 03].

inverse proportional to the distance between the location of the potential (i.e. spot of light) and the respective electrode:

$$U = R \cdot I \Leftrightarrow I = \frac{U}{R} \quad (2.1)$$

Since the resistive layer is as mentioned uniform the following equation holds true ²

$$R = \varrho \cdot \frac{l}{A} \Rightarrow R \sim l \quad (2.2)$$

using equations 2.1 and 2.2 we can conclude:

$$\Rightarrow I \sim l \quad (2.3)$$

- I : occurring current [A]
- U : generated potential [V]
- R : resistance of the photo-active area [Ω]
- l : distance between light spot and respective electrode [mm]
- A : cross-sectional area of the resistive layer [mm²]
- ϱ : specific electrical resistance of resistive layer material [$\Omega \cdot m$]

Hence, the relation between the location of the beam and the occurring photo-currents can be expressed as ([Photonics 03]):

$$I_1 = \frac{\frac{L_x}{2} - \Delta x}{L_x} \cdot (I_1 + I_2) = \frac{\frac{L_x}{2} - \Delta x}{L_x} \cdot I_0 \quad (2.4)$$

$$I_2 = \frac{\frac{L_x}{2} + \Delta x}{L_x} \cdot (I_1 + I_2) = \frac{\frac{L_x}{2} + \Delta x}{L_x} \cdot I_0 \quad (2.5)$$

$$\frac{I_1 - I_2}{I_1 + I_2} = \frac{2\Delta x}{L_x} \Leftrightarrow \frac{I_1}{I_2} = \frac{L_x - 2\Delta x}{L_x + 2\Delta x} \quad (2.6)$$

- I_1, I_2 : Output currents of resistive layer [A]
- I_0 : total photo-current ($I_1 + I_2$) [A]
- L_x : total length of active area (resistive layer) in between electrodes [mm]
- Δx : distance of light stimulated spot from center of the PSD [mm]

² Assuming uniformity of resistive layer: $\rho = const$ and $A = const$

Two basic types of PSDs are produced by various manufacturers worldwide:

- lateral PSDs
- segmented PSDs

Both types are produced in both one- and two-dimensional versions. However they have some things in common that are unique and superior compared to other optical tracking devices: They offer outstanding positional resolution for a wide spectral range of light used to stimulate the PSD. They also respond to changes almost without delay even without any additional biasing efforts that can be performed to reduce the delay. In this project experiments will be conducted with two-dimensional sensors of both types to find out which kind of chip is better suited for our specific application of robot calibration.

2.1.1 Segmented PSDs

Segmented PSDs are common substrate photo-diodes that are divided in segments and separated by a gap (see figure 2.2). This gap also referred to as “dead region” is a section of the chip that is not affected by any form of light stimulating it. The gaps are necessary to electrically isolate the PSD segments. Segmented PSDs are produced with either two (one-dimensional) or four (two-dimensional) segments.

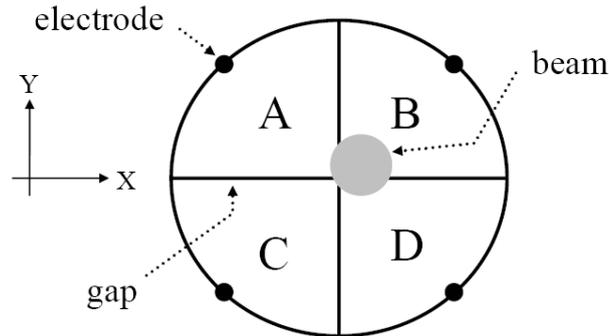


Figure 2.2: Schematic illustration of a two-dimensional segmented PSD. When light hits the active surface photo-currents occur in each segment. Those currents can be measured at the respective electrodes attached to each segment to determine the position.

The operation principle again is rather simple. When a beam of light hits the surface of the chip photo-currents are generated as explained in section 2.1. Again the currents are proportional to the intensity of the light the segment is exposed to. Therefore the relative position of the beam can be expressed as:

$$\begin{aligned}
 X &= \frac{(I_B + I_D) - (I_A + I_C)}{I_A + I_B + I_C + I_D} \\
 Y &= \frac{(I_A + I_B) - (I_C + I_D)}{I_A + I_B + I_C + I_D}
 \end{aligned}
 \tag{2.7}$$

X	:	relative beam position on X-axis [mm]
Y	:	relative beam position on Y-axis [mm]
I_A, I_B, I_C, I_D	:	photo-currents measured in PSD segment noted in the indices (see figure 2.2) [A]

As one can see, the position in X and Y direction can easily be calculated based on the photo-currents that can be measured in each segment. The achievable resolution (i.e. minimal detectable change in beam position) with this type of PSD is approximately $0.1\mu m$. However there are several restrictions that need to be fulfilled in order to get correct results:

- the beam has to overlap all segments at all times
- the diameter of the focused beam has to be larger than the gap in order to reach the active area and generate an output
- the beam's intensity distribution must be uniform since photo-currents in the respective segments are proportional to intensity

2.1.2 Lateral PSDs

The second type of PSD tested for this project is usually referred to as lateral PSD. This type is also available in one- and two-dimensional realizations. The one-dimensional version is shown in the previous section (figure 2.1). The chip used for 2D position detection is manufactured in a similar way. The only difference is that the bottom-layer is also equipped with two electrodes. This second layer works exactly the same as the top layer. The only difference is that the electrodes mounted on the bottom layer are aligned in a 90° angle relative to the top layer to represent the Y -axis (see figure).

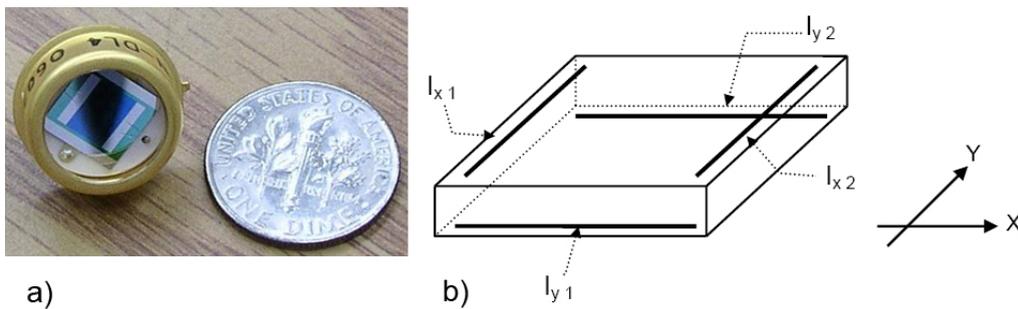


Figure 2.3: a) Photograph of lateral effect PSD (OSI Optoelectronics[®] DL-4) next to a dime. b) Schematic illustration of a two-dimensional lateral PSD. The top resistive layer is used to determine the beam location in X -axis direction. Therefore two electrodes are mounted on the left and right end of the active area to gather the occurring photo-currents. Bottom layer is equipped in the similar way to determine location in Y -axis direction.

In the lateral PSDs the relative two-dimensional position on the active surface of the chip can be expressed as:

$$\begin{aligned} X &= \frac{I_{x1} - I_{x2}}{I_{x1} + I_{x2}} \\ Y &= \frac{I_{y1} - I_{y2}}{I_{y1} + I_{y2}} \end{aligned} \quad (2.8)$$

$$\begin{aligned} X &: \text{relative beam position on X-axis [mm]} \\ Y &: \text{relative beam position on Y-axis [mm]} \\ I_{x1}, I_{x2}, I_{y1}, I_{y2} &: \text{photo-currents measured (see figure 2.3) [A]} \end{aligned}$$

The major advantage of this type is that the accuracy of the output is not affected by the spot profile of the beam or its intensity distribution. The positional resolution is lower than the one offered by the segmented type. The achieved resolution of $0.5\mu\text{m}$ however is still more than sufficient for the given application of robot calibration. Another outstanding property of this type is the position linearity over the whole active surface of the chip. This is important for our task since it allows us to keep the error at a low level during the mapping and compensation process. Furthermore the area of the chip that offers high resolution is a lot bigger than the one of the segmented type as the results in section 3 will show.

2.2 PSD Output Processing Board

2.2.1 Board Design

Before the actual calibration task could be approached, a signal processing circuit had to be designed and set up in order to test the properties of the chosen PSD chips (i.e. position linearity over active chip area, positioning sensitivity etc.). Since the active chip area of the PSD is comparatively small (i.e. $4\text{mm} \times 4\text{mm}$ for the used type) the measurements had to be performed applying high precision tools to position the laser beam over the surface. Therefore a Signatone CAP-945[®] high precision probe was used to acquire sufficient accuracy to calibrate the chip.

The purpose of this board is simply to process the PSD sensor raw output to be able to determine the relative 2D position of the beam on the chip's surface. The design can be structured into three functional stages. In the first stage the output signal of the four PSD electrodes is amplified using operational amplifiers. The second and third stage are used to perform the computations needed to be able to determine the spot position. In the second stage summing amplifiers and differential amplifiers are occupied to generate the input signals for the divider ICs (third stage).

2.2.2 Board Output

To perform the operations required in this context operational amplifiers are applied in various operation types. For the first stage inverting amplifier circuits are used. The input voltage U_{in} is amplified to the output voltage according to the following

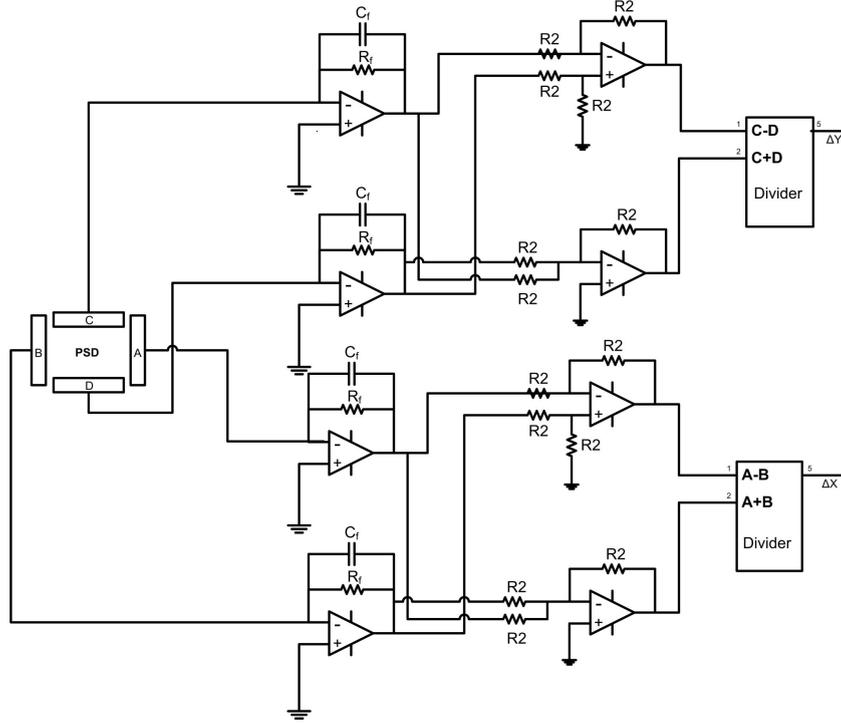


Figure 2.4: Schematics of the test board circuit used to experiment with lateral PSDs. The design can be structured into three functional stages: Amplification, Subtraction/Adding, Divider.

equation that can be deduced from [Hans 02]. Since the capacitor C_f (used for high frequency noise reduction) can be disregarded in case of DC stimuli (as provided by the PSD) the amplification is only depending on the feedback resistor R_f .³ Due to the fact that the output for both coordinate axis are processed in an equal manner, only the x-position will be deduced here. The position in X direction ($pos_x(A, B)$) however only involves the output signals A and B the equation for the y-position can be deduced by replacing A and B with C and D.

$$U_{A_{amp}} = \frac{-R_f}{R_1} \cdot U_A \Rightarrow U_{A_{amp}} \sim R_f \Rightarrow A_{amp} = -v \cdot U_A \quad (2.9)$$

- $U_{A_{amp}}$: amplifier circuit output voltage [V]
- U_A : amplifier circuit input voltage [V]
- R_f, R_1 : resistors [Ω]
- v : amplification factor [1]

Once the PSD output is amplified the second stage is needed to perform some calculation realized in analog operational amplifier circuits. For the prototype board two standard applications are used: summing amplifiers and differential amplifiers. In

³ $R_1 = const$ is the output resistance of the PSD. Assuming $R_f \gg R_1$ a sufficient amplification ratio ($v \gg 1$) can be achieved.

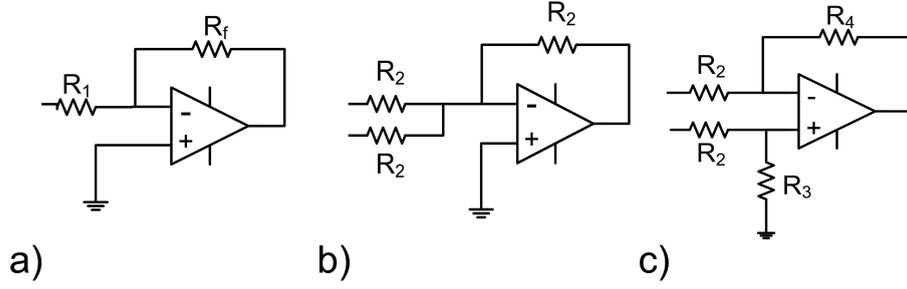


Figure 2.5: Schematics of the operational amplifier circuits as used in the test board. a) Inverting operational amplifier circuit used to amplify the PSD output signal. b) Summing amplifier circuit with identical resistors (i.e. $v = 1$). c) Differential amplifier used to subtract the pre-amplified signals in second stage. The potentiometers R_3 and R_4 are used to determine the amplification of this stage.

both cases the amplification ratio (v) was set to approx. $v = 1$ in order to maintain a good signal to noise ratio. For the summing amplifier the following equation can be deduced

$$\begin{aligned}
 U_{A_{amp}+B_{amp}} &= -R_2 \cdot \left(\frac{U_{A_{amp}}}{R_2} + \frac{U_{B_{amp}}}{R_2} \right) = \frac{-R_2}{R_2} \cdot (U_{A_{amp}} + U_{B_{amp}}) \\
 &= -1 \cdot (U_{A_{amp}} + U_{B_{amp}}) \quad (2.10)
 \end{aligned}$$

$U_{A_{amp}+B_{amp}}$: amplifier circuit output voltage [V]
 $U_{A_{amp}}, U_{B_{amp}}$: amplifier circuit input voltages $U_1 + U_2 = U_{in}$ [V]
 R_2, R_4, R_4 : resistors [Ω]

Whereas for the differential amplifier circuit these equations apply

$$U_{A_{amp}-B_{amp}} = \frac{R_4}{R_3 + R_4} \cdot \frac{R_1 + R_2}{R_1} U_{A_{amp}} - \frac{R_2}{R_1} \cdot U_{B_{amp}} \quad (2.11)$$

Assuming $R_1 \approx R_3$ and $R_2 \approx R_4$ we can simplify equation 2.11 to: ⁴

$$U_{A_{amp}-B_{amp}} = \frac{R_2}{R_1} \cdot (U_{A_{amp}} - U_{B_{amp}}) \Rightarrow U_{A_{amp}-B_{amp}} = U_{A_{amp}} - U_{B_{amp}} \quad (2.12)$$

$U_{A_{amp}-B_{amp}}$: amplifier circuit output voltage [V]
 $U_{A_{amp}}, U_{B_{amp}}$: amplifier circuit input voltages [V]
 R_1, R_2, R_3, R_4 : resistors [Ω]

⁴ Since in this case $R_1 = R_2$

The divider IC output as used in this circuit equals:

$$U_{out} = 10 \cdot \frac{-1 \cdot U_{in1}}{U_{in2}} \quad (2.13)$$

After the signal is processed in the third stage (Divider) the resulting output can be determined using equations 2.9, 2.10, 2.2.2, and 2.13. This allows it to calculate the resulting x-position as:

$$\begin{aligned} U_{pos_x} &= 10 \cdot \frac{-1 \cdot U_{A_{amp}-B_{amp}}}{U_{A_{amp}+B_{amp}}} = 10 \cdot \frac{-1 \cdot (U_{A_{amp}} - B_{amp})}{-1 \cdot (U_{A_{amp}} + U_{B_{amp}})} \\ &= 10 \cdot \frac{(-v \cdot U_A) - (-v \cdot U_B)}{(-v \cdot U_A) + (-v \cdot U_B)} = 10 \cdot \frac{U_A - U_B}{U_A + U_B} \end{aligned} \quad (2.14)$$

Hence, the resulting Y-position is

$$U_{pos_y} = 10 \cdot \frac{U_C - U_D}{U_C + U_D} \quad (2.15)$$

The total resulting output of all stages is equivalent to the equation that can be found in paragraph 2.1. Therefore this test board can be used to compute the relative 2D position of the light beam, the PSD is exposed to.

2.2.3 Parameter Tweaking

To determine the best suited values for the components (i.e. resistors and capacitors), multiple factors have to be considered. This is for example the time constant (cutoff frequency) and the noise to signal ratio just to name a few examples. Hence some initial test values were determined by setting up a prototype board and performing extensive tests for various settings. In order to reduce the complexity (i.e. number of alterable components) for the test with the final board produced in PCB technology two out of three variable values were chosen to be fixed. To determine the best suited value the conditions of use have to be considered. Since the board output is intended to be used as feedback for the robot to position the laser on the PSD center the signal rise time (10% to 90%) must not be too high. Therefore the maximum reasonable rise time was assumed to be about 0.1 sec. The rise time however is dependent on the time constant of the first circuit stage:

$$\tau_{rise} \approx 2.2 \cdot \tau \Leftrightarrow \tau \approx 0.05s \quad (2.16)$$

With the value set for τ one can now easily estimate the remaining component values. C_f set to $1\mu F$ yields:

$$\tau = R \cdot C \stackrel{!}{=} 0.05s \Leftrightarrow R = \frac{5 \cdot 10^{-2}s}{10^{-6}F} = 5 \cdot 10^4 \cdot \frac{kg \cdot m^2}{s^3 \cdot A^2} = 50k\Omega \quad (2.17)$$

Therefore we can conclude that the resistor used in the first stage must not be bigger than $50k\Omega$. To receive good results, the resistors of the second level should have the

same magnitude. For calculation convenience the value for R_2 was set to $10k\Omega$ since this value has proven to deliver good results in the initial experiments. This value applies to all but one of the resistors in the second stage. In each amplifier circuit one resistor was chosen to be adjustable to be able to control the gain. The most promising results (in the aspect of noise to signal ratio etc.) were achieved using an Analog Devices AD534[®] divider IC and a OPA37 low noise Burr-Brown[®] operational amplifier (see figure 2.4).

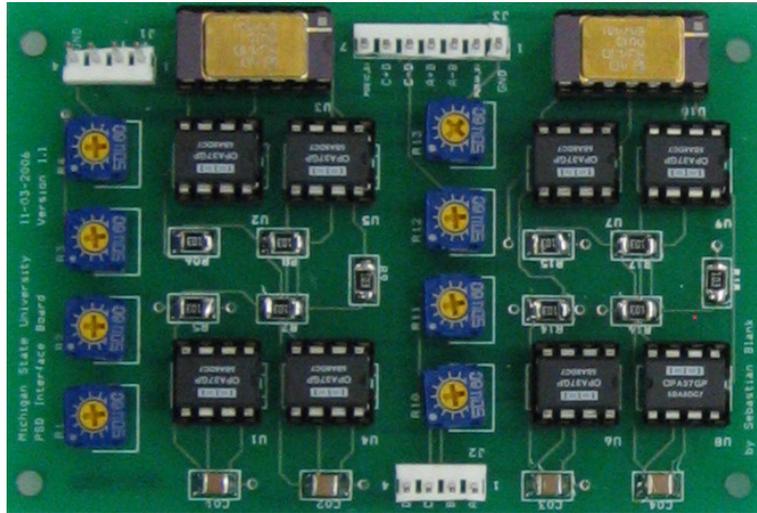


Figure 2.6: Photograph of the manufactured board, designed to compute the PSD positional output. The size of the two layer is PCB-board is approx. $6.4cm \times 8.9cm$.

To be able to finally determine the quality of the processed PSD sensor output signal the best setup for the adjustable resistors in the 1st and 2nd stage has yet to be found. Therefore another experiment was performed in order to find the best fitting value combination given the constraints above (i.e. $R_2 \leq 50k\Omega$). Hence, the PSD sensor was connected to the board and the processed signal was measured using an oscilloscope. Initially the two first stage potentiometers were set to the desired value ($\in [5k\Omega, 50k\Omega]$). After that the laser beam was positioned in 4 locations on the PSD surface (top & bottom each far left & right) that had proven to produce most likely the highest output (as experienced in previous experiments). For each setup the second stage potentiometers were adjusted. The target was to maintain a preferably high signal amplitude just below the operational amplifiers saturation level ($-11,9 V$, $+11,4 V$). This is especially important to guarantee that board output is deformed as little as possible during the signal processing. Since multiple factors had to be optimized the adjustment had to be performed in various cycles to get closer to the final result. Therefore the two potentiometers of the 2nd stage were adjusted until the board produced a zero output signal as long as no beam was emitted to the PSD. After this was achieved, the laser was moved to the four specific locations mentioned above and the value was altered to a sufficient level of amplification. Subsequently the cycle the laser was removed again to check the zero-input signal. This procedure was repeated until satisfying results were obtained.

A table showing the experimental results for various resistor values can be found in the appendix section C.1. The achieved signal amplitude in this experiment was approximately $A_{signal} = 22V$ with a measured noise contribution of $A_{noise} = 0.025V$. The signal-to-noise ratio can be calculated as:

$$SNR(dB) = 10 \cdot \log \frac{P_{signal}}{P_{noise}} = 20 \cdot \log \frac{A_{signal}}{A_{noise}} \approx 135.59dB \quad (2.18)$$

For the further experiments the setup of $20k\Omega$ for the first and $49k\Omega$ was chosen. This is due to the fact, that this setup seems to offer the best trade-off of response time vs. signal consistency for our purposes. This manifests in a fast reaction to beam movement while maintaining a steady signal when the beam remains at the same location.

2.3 Experimental Results

Now that the board is designed and the component values are set, the first experiments concerning the PSD sensor can be performed. For this purpose, the same type of laser pointer used in the laser fixture on the actual robot was fixed on a Signatone CAP-945[®] computer aided probe (CAP) perpendicular to the PSD surface. This device allows it to position the laser in 3 DOF (x,y,z) over the PSD chip surface with high precision (accuracy $\pm 2.5\mu m$, repeatability $\pm 1\mu m$). The PSD chip was fixed on a floating, shock absorbing workbench together with the probe to eliminate influence of e.g. operator movement and vibrations. After the setup was prepared two different experiments were conducted with both the lateral and the segmented PSDs. The first experiment is intended to determine if the positional output of the board is dependent on the direction the position is reached from. Therefore the laser was moved in a straight line over the PSD surface in both directions. First from bottom to top and then vice versa. The second experiment deals with the linearity of the output signal of the two positional channels (X,Y). In order to achieve results with sufficient positional resolution the laser was moved over the chip surface in two cycles. In the first cycle the probe was moved on a straight line parallel to the y-axis with a line spacing of approx. 0.2 mm. Therefore pseudo-continuous y-positions with discrete x-locations were recorded. In the second cycle the y-position remained discrete in 0.2 mm steps while the x-position was altered continuously. The result of this process will be presented in the following sections.

2.3.1 Lateral PSD Results

In the following section the results of the experiments described above will be presented and analyzed. Before one is able to evaluate the signal properties, it is important to understand the influence of the PSD's physical surface design on the signal. Therefore before the actual experimental results are discussed the focus will be directed to the correlation of the chip design and the emitted signal. One of the features of the lateral PSD is that the structures emitting the output signal for x- and y-position are physically separated from each other. Hence one signal can be analyzed without the need for considering the other because they don't interfere with each other in this context. This however is not necessarily true for the mapping. This will be discussed in the respective paragraph.

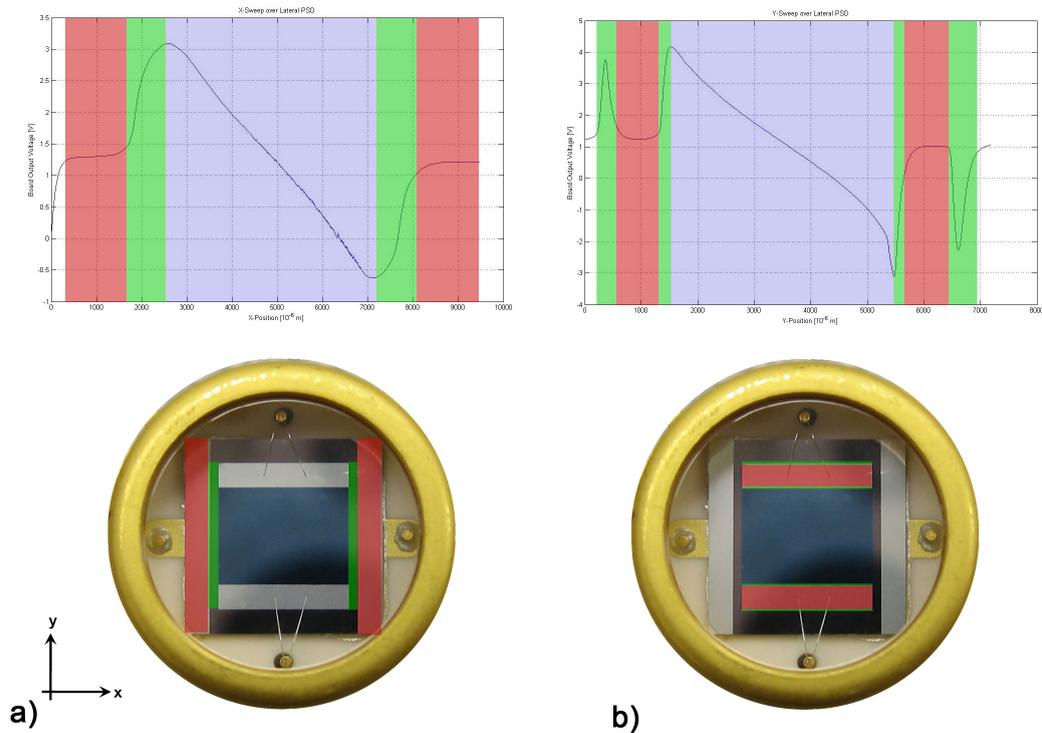


Figure 2.7: Diagram showing the correlation between the lateral PSD's physical surface layout and the respective signal recorded during sweep. *a) Top:* Plot of the signal during a sweep with constant y -position along the x -axis. *Bottom:* Photograph of a lateral PSD surface. The signal section highlighted in red corresponds with the laser passing the optical non-active metallic electrodes (also marked red on the chip). In the same way the influence of the gap (green area) is marked. The blue marked section limited by the two gaps corresponds with the active area of the PSD. As one can see the output changes mostly linear with the position of the laser beam in this section. *b) Top:* Plot of the signal during a sweep with constant x -position along the y -axis. *Bottom:* Photograph of a lateral PSD surface. The influence of the layout on the signal is presented in the same way. Since there is no gap this time, the transition between electrode and active area is highlighted green on either side. Again the change of position on the active area results in a predominantly linear output.

The signal representing the x -output can be recorded while sweeping the laser beam along a straight line that is parallel to the x -axis of the chip at various fixed y -positions. The output for one of these sweeps is presented in figure 2.7 a) on the top. The segments of the graph highlighted in green, blue, and red corresponding with the physical structures (shown on the bottom) of the chip design. At the beginning and end of the signal representation, the most outer parts of the chip can be found: the electrodes (red). Since they don't react on stimulation in form of light the signal shows a flat line that is parallel to the x -axis for the time that laser beam traverses this section. In between the active area and the electrodes a rather big gap (highlighted in green) can be found that is also not actively responding to

light stimuli. As the laser passes this section and gradually moves its full beam profile to the active area, a constant rise in the signal can be observed until the gap is completely passed (signal peak). After this peak the beam travels over the actual sensor surface (blue). A rather linear output in respect to the change in position can be observed in the signal. This is very favorable to our task of precision position determination, since the output can be easily approximated by a straight line as necessary for the calibration process, to allow position determination without a significant amount of deviation from the signal. As the laser leaves the active sensing area the same signal characteristics as before occur.

The signal for the y-channel is presented in a similar manner in 2.7 *b*). Here however, the signal outside of the active area shows differences to the x-channel caused by the differences in the chip design. In the y-signal a very narrow spike precedes the already familiar signal that shows a steeper incline for the region corresponding with the gap section of the signal. The additional first and last peak can also be explained with the chip design. Since the electrodes are now positioned directly on the active area instead of next to it (as seen in x-direction) a small part of the active area overlaps the outer side of the electrode from underneath. Hence, the laser passes this region first before it reaches the neutral electrode and generates a massive change in signal. For the same reason the signal incline is much steeper since now there is no more gap and therefore no soft transition of active and non active material as before. The physical design details of the chip can be seen in figure 2.8. Here a microscope was utilized to take some magnified photographs of details that influence the signal output. In part *a*) of this figure the edge of the active area causing the first and respective fourth peak in the y-axis signal, is visible as tiny glimmering section in the middle separating the chip's neutral base material (top, dark gray) from the electrode (bottom, light gray). In *b*) the gap in x-direction enclosed by the electrode (left, silver structure) and the active surface (right, darker section) can be identified. The third magnified image presented in *c*) shows the part of the surface used in the y-direction that corresponds with part *b*) in the x-direction. As mentioned before there is no gap in between the two areas this time.

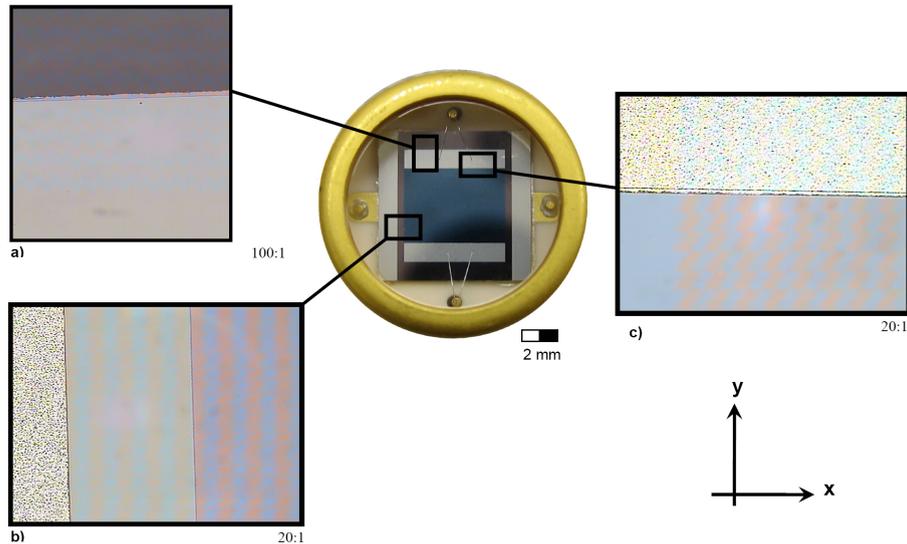


Figure 2.8: Magnified photographs of a lateral PSD surface taken with a microscope at two levels of magnification (20:1,100:1). *a*) Outer edge of the electrode (light gray area, bottom) in y -direction. The small glimmering section separating the electrode from the neutral base material (dark gray area, top) is sensitive to light stimuli. *b*) Non-photo-active gap (light gray, middle) separating the electrode (metallic surface, left) from the active area (darker gray, right). *c*) The y -direction electrode (top, silver) and the active area (bottom, darker gray) are visible. In opposition to the x -direction no gap in between those two structures can be found.

Now that the impact of the chip design on the signal emitted is discussed the focus can be redirected back to the results gathered for the experiments introduced in the previous section. Before the results for the two (x,y) full surface sweeps are presented, the results of an experiment focusing on the correlation of the current output relative to the previous position of the laser beam i.e. hysteresis effects. This is especially important for all later experiments because if it can be shown that only negligible little or even no hysteresis effects occur, this will dramatically reduce the amount of data that has to be gathered and evaluated since the output for each position on the surface has just to be recorded once instead of several times (reaching the position from various directions). To investigate this matter let us take a look at the result of the experiment presented in figure 2.9.

To be able to record the signals for the x -direction (2.9 *a*)) the laser beam steadily moved over the surface parallel to the x -axis with a y -position of 0.6 mm below the upper left edge in both ways. First it was moved from the zero position in direction of inclining x -values (blue graph) and then backward along the same way (red graph). It shows that the signals nearly correspond with each other except for minor deviations caused by measuring tolerances of the equipment used and change in conditions (temperature etc.). The only part that shows mentionable deviations is the transition from the electrode to the active surface and vice versa. This phenomenon occurs even clearer in the y -signal (2.9 *b*)). Since there are more and sharper significant changes in signal (peaks) one can observe that the effect

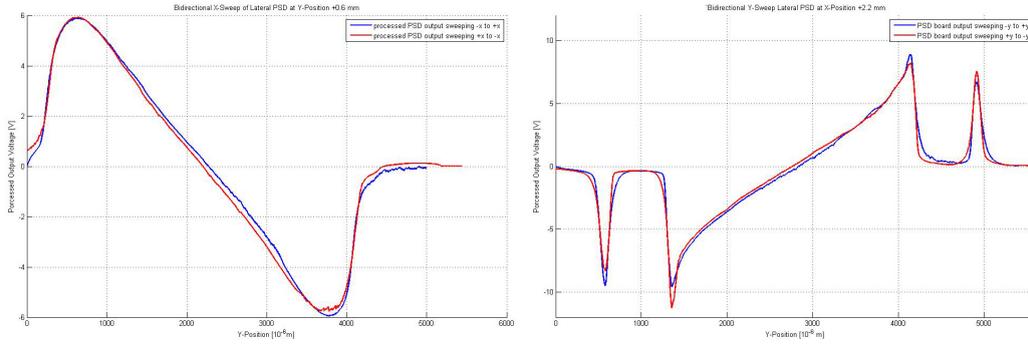


Figure 2.9: Plot of the recorded signal for a line sweep over the PSD surface. *right:* Plot of processed PSD signal for a sweep parallel to the x-axis with a fixed y-position (0,6 mm away from the top left edge of the active area). The blue curve indicates the recorded signal for a left-right sweep, while the red curve represents a right-left sweep. *left:* Plot visualizing the same issue for a sweep parallel to the y-axis with fixed x-position (+ 2.2mm). Again the same signal behavior can be observed.

seems to be symmetrically occurring in both the front and the back directed sweep. While the blue graph shows a higher amplitude in the first and third peak, the red shows this behavior at the second and fourth peak. At the two inner peaks (2nd, 3rd) even a symmetry in sharpness of the peak is traceable. The signal with the higher amplitude shows a sharper, narrower peak than the one with the lower amplitude. The possibility of an effect based just on random changes in signal becomes even more unlikely when the curvature of the graphs in the transition between active area and electrode (y-positions 1200 to 1800 μm and 4700 to 5300 μm) is examined. Depending on the direction of the sweep the graph first shows a rapid change when the outer edge of the electrode is passed (blue y-position 1200 μm) but a way less steep decline when the inner edge is traversed (blue y-position 1800 μm). This can be observed in an inverted manner between the third and fourth peak for the red signal. This specific behavior can be explained however with the non-homogeneous energy distribution of the used laser beam. The energy density is higher at the front end of the beam than at the rear end. Considering these conclusions it can be assumed that the lateral PSD chip used in this case does not show significant hysteresis effects (maximum signal deviation 0.16 V at with a signal amplitude of about 24 V).

In the following paragraph the results of the two full surface sweeps (x- and y-direction) is going to be presented.

In figure 2.10 the results of the experiments concerning the x-position of the processed PSD output are presented. To gather the data (approx. 90,000 points on a 5.2mm \times 4.5mm surface) the laser beam was moved to discrete y-positions on the chip and parallel to the x-axis (both relative to the CAP) to receive data that can almost be assumed to be continuous (y-spacing 200 μm line to line, x-spacing $< 2\mu\text{m}$ point to point). After the data was collected using the computer aided probe as described in the previous passages, it was processed. Therefore MatLab[®] was used to adjust the (non-steady) sampling rate to a common base and reduce the

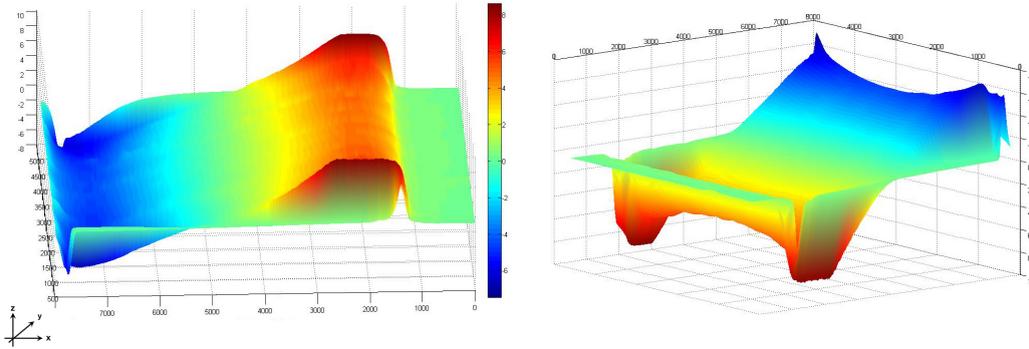


Figure 2.10: 3D plot of the data of all sweeps parallel to the x-axis combined generated using MatLab[®]. The y-spacing used is 0.2 mm between two sweeps. To cover the whole chip 23 lines were used resulting in approx. 90,000 points recorded. The graph clearly shows the linear characteristics of the output over the full active area (plane within the two peaks) for all y-positions (0 to approx 4500 μm). Furthermore the location of the electrodes (neutral green plane next to the peaks) is clearly visible.

noise caused by the recording equipment.

The graph presented shows the board output voltage (z-axis) for a sweep over the full active area (x-, y-axis) of the lateral PSD. The results observed for a single sweep presented earlier also holds true in case of the whole surface. The properties of the signal of the active area (minimum peak to maximum peak) obviously prove to be very close to linear. Furthermore it is noticeable that the output seems to be axially symmetric in x-direction along the middle of the active area ($x \approx 4500\mu\text{m}$) and y-direction ($x \approx 3000\mu\text{m}$). It is also noticeable that the maximum peak heights in y-position seem to form a parabola (approx. $0.0166 \cdot x^2 - 0.356 \cdot x^1 + 6,544$) with a minimum at the y-direction middle of the surface. The same holds true for the minima of the inverted direction (parabola opens towards negative z-axis direction). Accordingly the lines of equal potential on the whole surface follow the peak behavior with declining curvature while moving towards the middle (middle is a straight line). Therefore the most outer line of equal potential forms a cushion shape while the most inner one forms a perfect square. The data gathered also proves that the dimensioning of the processing board components was correct since no sign of amplification saturation level exceeding, which would manifest as a plateau shape (parallel to the x-y-plane) at the peaking areas, can be found.

In figure 2.11 the result of the full surface sweep in y-direction is presented. As already seen in the x-direction surface sweep before, the data resembles the signal properties of the y-direction output discovered in the previous sections. Again the two peaks enclose the very linear appearing output of the active surface in the middle. On the other hand one is also able to identify the signal representation of the non-photo-active electrodes as planar surfaces with almost zero potential (approx. y-positions: 500 to 1000 μm and 4300 to 4800 μm).

This time the peaks do not seem to follow a continuous change in height but show slight variation in the maximum output level reached. Except for these influences

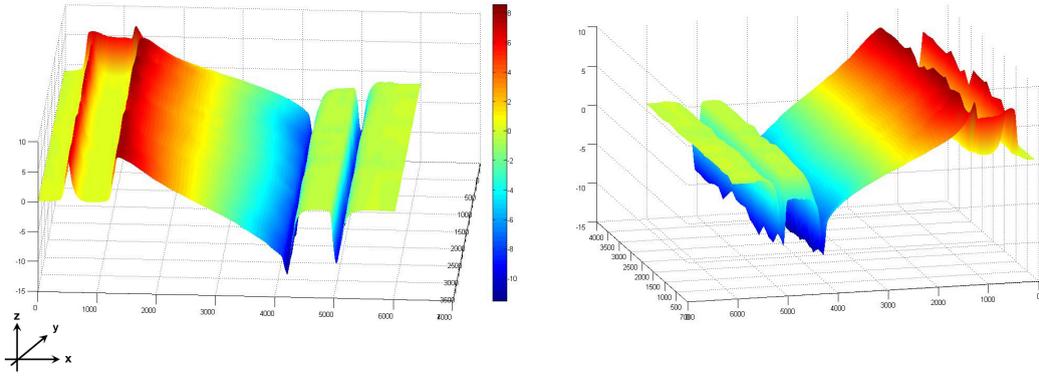


Figure 2.11: 3D plot of the data of all sweeps parallel to the y-axis combined generated by MatLab[®]. Again the spacing in between the sweep lines is 0.2 mm. This results in 20 lines and therefore includes approx. 80,000 points of recorded data. The light sensitive active area proves to deliver linear output once more. The two green planes at approx. zero potential can be identified as the electrodes framed by two peaks caused by two gaps on the chip that respond to stimulation by the laser beam.

the peaks seem to form a steady straight line over the whole surface examined. The simple variation in height are most likely caused by irregularities in the electrode edge structure (see figure 2.8 a),b). The limited (but considerably high) resolution of $2\mu m$ contributes to this outcome since the recorded signal (smoothed by the capacities used on the board) does not show the precise output at a point in time but rather an average over the sampling period. Therefore the fine structures of the electrode (-edge) variations manifest as “dents” in the line.

Except for these effects the signal shows again a two-dimensional axial symmetry along x-direction (approx. line of zero output at $2500\mu m$) and y-direction (approx. $2000\mu m$). Again the lines of equal potential form a cushion shape on either side of a straight line in the middle of the active area. The curvature this time however is less grave than observed in the x-sweep.

2.3.2 Segmented PSD Results

In this paragraph the experimental results for the segmented will be presented. Since the first experiments already delivered sufficient information about the signal not all experiments performed for the lateral type were also conducted for the segmented type. Before discussing the segmented PSD output in detail however, again the general properties of the output will be discussed related to the physical chip layout.

As presented in figure 2.12 the changes in signal can again be traced back to the design features of the sensor. The datasets presented were gathered while performing a y-axis sweep with a x-location slightly right of the middle gap. In the first part a steady and almost linear incline in output voltage can be observed (position 0 to $800\mu m$). This can be associated with the sensor approaching the photo-active area from the outside and slowly moving in positive y-direction towards the middle. Once the peak is reached the signal almost steadily maintains at nearly the same

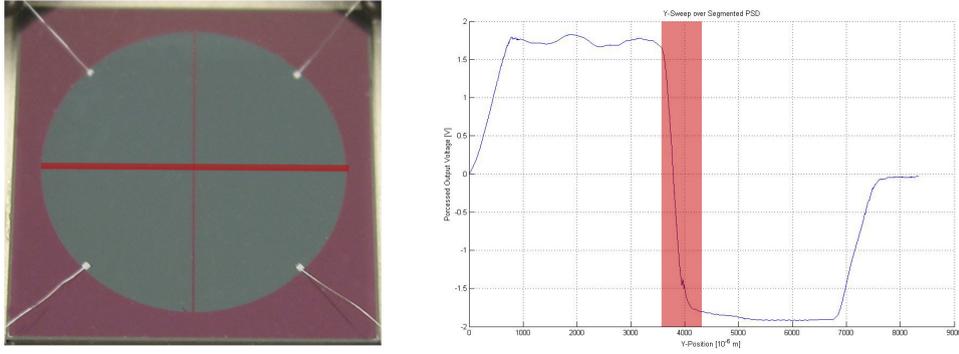


Figure 2.12: Diagram showing the correlation between the segmented PSD's physical surface layout and the respective signal recorded during sweep. The almost linear inclining section at the beginning (0 to 800 μm) and end (6800 to 7600 μm) of the sweep are caused by the laser beam moving in and out of the photo-active area. Except for the section close to the middle gap (3700 to 4200 μm) the sensor shows a very non-linear output in respect to the change of beam position.

level while the location of the laser beam is steadily altered. Therefore it can be determined that the relation between position and output is strongly non-linear for this section. As the beam traverses the middle gap section (3700 to 4200 μm) the output starts to decline again. In this section a very linear change of output to position can be observed. Since the change in amplitude of this region is considerable high, the positional resolution can be assumed as very good for this part. Once the beam passes the section, the same pattern as before occurs: The output maintains at almost the same level before declining as the laser beam reaches the edge of the active area (6800 to 7600 μm).

Now that the general signal properties of the segmented PSD output are discussed one can proceed with analyzing more detailed signal characteristics. A very important one again is the occurrence of hysteresis effects. Therefore the same experiment as already presented for the lateral type was conducted. The results are presented in figure 2.13.

For high precision area around the middle gap no significant signal deviation of the two recorded outputs (one moving in negative-, the other one in positive y-direction over the same part of the surface) are noticeable. Once this region is left however, the signal recorded moving in negative y-direction (red) shows a steady offset in the section close to the edge of the active area (2000 to 0 μm). Therefore it can be stated that hysteresis effects occur in this type of chip.

2.3.3 Result Evaluation

Now that all the relevant data is collected and analyzed, a decision has to be made concerning the type of chip best-fitted to the demands of the task of optical robot calibration. In this case it is especially important to discuss the special advantages and disadvantages observed in the experiments in both types of PSD sensors and make a decision based on the impact of these properties on the calibration process. In the segmented PSD the area around the middle gap seems to offer a good (and

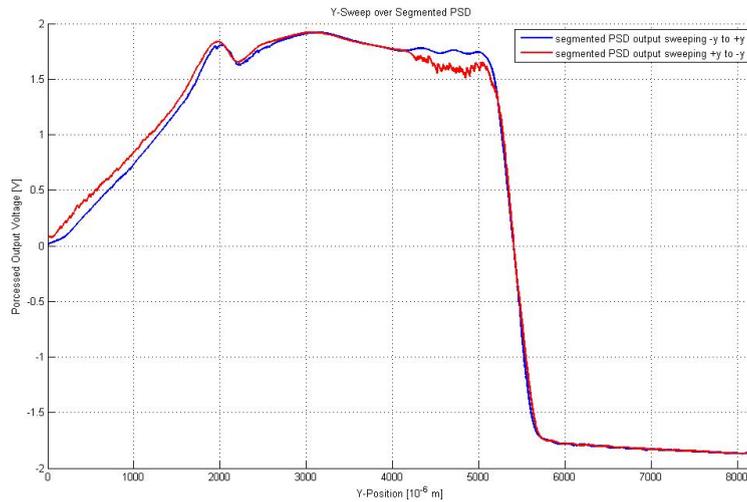


Figure 2.13: Plot of two sweeps at identical positions on the segmented PSD but with opposite directions. Minor hysteresis effects can be found especially in the first section (y -position 0 to 2000 μm).

for this task sufficient) positional resolution with an almost linear characteristic. The linearity aspect is very important since this will make the process of finding the proper feedback for the controller actually centering the robot over the PSD a lot easier and allows it to reach a higher level of precision. Another positive aspect of this type is the big active area ($\approx 78.5mm^2$) of the chip that makes the process of initial preliminary positioning easier. However, the major disadvantages of this type are the low overall signal amplitude ($\Delta U \approx 4V$) which results in a poor overall positional resolution outside of the middle section and the absence of change in signal with positional change of the beam over a wide section of the surface ($\approx 70\%$ of the active surface). The discovered properties will not allow it to generate a proper feedback for the controller later on. Therefore if this technology is chosen, the controller output will not only be dependent on the current input but also has to depend on knowledge of multiple previous states reached due to occurring hysteresis. This will make the calibration process a lot more complicated if not impossible.

On the other hand the properties of a lateral PSD sensor were analyzed. This particular type shows some major advantages over the segmented PSD chip in multiple ways. The mapping of the emitted signal and the laser beam position is in opposition to the segmented type unique for the whole active area. Therefore the control process can be significantly simplified, since the controller output here only depends on the current input of the emitted signal. Again, the signal for beam positions within the active area shows strong linear characteristics and is therefore easy to map in order to estimate the current position. Another positive property of the lateral sensor is the high output amplitude (due to the solid design of the processing board) of approximately 24 V allowing a high positional resolution and therefore a

high level of accuracy. The only disadvantage compared to the segmented type is the smaller active area ($\approx 23mm^2$) this however is only a minor trade-off for the significant benefits associated with the lateral type. Therefore this type was chosen to be used in the calibration process.

3. Robot Control

The subject to be discussed in this chapter is amongst the most crucial elements of the calibration system: the controller. During the calibration process it supposed to move the robot arm to a centered position over the PSD sensor automatically. In this case however the feedback for the controller can not directly be obtained from the plant (i.e. PSD). This is caused by the fact, that the PSD only provides output voltage that has to be mapped to a position in order to be used for the controller that has to position the robot arm. This problem is known in the field of robotics as *visual servoing*. A good introduction to this topic can be found in [Hutchinson 96]. To perform the design of the controller however we first need to take care of the feedback the controller needs in order to be able to control the robot movement. After adequate feedback is provided and tested in the first two sections, the controller design will be discussed in the third section.

3.1 Feedback Mapping

The need for feedback mapping is obvious since as mentioned the data gather about the plant by an“external” observer (PSD) has to be “translated” into a feedback the controller can deal with (i.e. position). Therefore a mapping function has to be found that performs this task. During the process of finding a suited function, several requirements the function has to meet have to be considered. On the one hand it is important that the mapping is as accurate as possible since this is decisive for the level of precision of the whole system. On the other hand the function must be invertible in a unique way so that all positions on the PSD surface can be uniquely mapped to an output voltage. Considering the results found for the PSD output earlier it is obvious that this can be achieved by using a polynomial with the degree of $n = 1$ for each of the respective coordinate directions. To find the proper function MatLab[®] was used to fit an average polynomial for the earlier recorded PSD data:

$$pos_x(u) := -0.0042u + 5.5001V \quad (3.1)$$

for the given PSD output u [Volts] and the position in μm . The respective polynomial for the y-direction is:

$$pos_y(u) := -0.0087u + 7.945V \quad (3.2)$$

The resulting graph comparing the recorded PSD signal to the respective mapping function can be seen in figure 3.1.

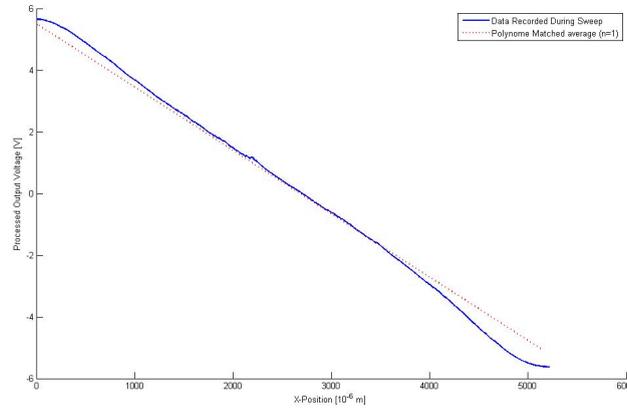


Figure 3.1: Plot of the recorded sweep data along x-axis (*blue*) and the estimated matching polynomial. The middle region shows an almost perfect match but significant deviation occurs close to the edge on both sides.

The graph shows significant deviation of the calculated function compared to the real data, but a rather good match when the location of the beam is close to the middle (1500 μm to 3500 μm). Outside of the very linear middle area the PSD output voltage changes slightly less linear (see figure 2.10). This result corresponds with the observations made during the recoding of the PSD data (section 2.3.1). The performance of the mapping function however is still more than sufficient since the only required robot arm movement for a position outside of the middle is to reposition the beam closer the center. This is ensured by this mapping function, so the deviation is bearable. As soon as the beam reaches the center section the positional mapping can be performed with high precision, so a good final position can be achieved.

After the mapping function was found the output mapping was tested for the first time. For this purpose a laser pointer was mounted to on a precision x,y-table (positional resolution $\leq 1 \mu m$). The orientation of the beam was adjusted till it hit the PSD surface with a 90° angle. Hence, one is able to move the laser beam with high precision over the surface to generate patterns and then compare the desired pattern to the computed mapping output. Using a simple line pattern along the x-axis and the y-axis, one was able to observe that a change in x-location will change the y-signal even if the y-position is maintained constant. This is based on the fact that the lines of equal potential show cushion-shape i.e. signal output for constant y-location while changing the x-location is represented by a parabola. Considering

this observation, it becomes obvious that the signals need to be decoupled first to achieve satisfying mapping results. Therefore a decoupling matrix D_A is employed:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = D_A \cdot \begin{bmatrix} x \\ y \end{bmatrix} \quad (3.3)$$

Now that the signals are decoupled, the mapping result has to be rotated around the origin $(0, 0)$ to map PSD axis to coordinate axis. Hence, a 2D-rotation matrix R is used:

$$\begin{bmatrix} x'' \\ y'' \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \cdot \begin{bmatrix} x' \\ y' \end{bmatrix} = R \cdot \begin{bmatrix} x' \\ y' \end{bmatrix} \quad (3.4)$$

This results in the final mapping position (x'', y'') .

3.2 Feedback Test

To test the mapping performance more complex patterns were used to determine the mapping functions quality. The results of these tests will be presented and discussed in the following section. The first patterns used were two straight lines along the coordinate axis. This pattern is best suited in this context to determine if the signal decoupling and rotation were sufficient. If this is the case the mapped output will follow the physical movement of them beam.

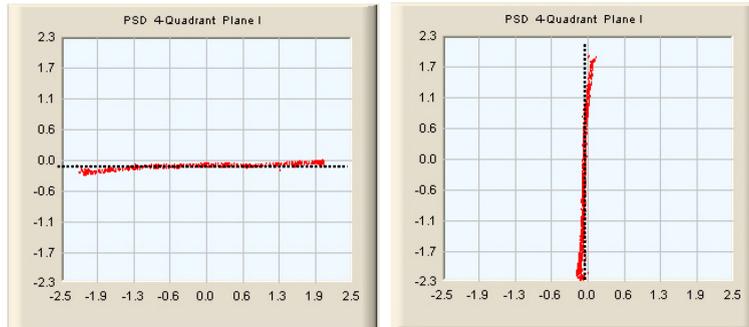


Figure 3.2: Screenshot of PC-interface software showing a trace of the beam movement (red) along a straight line (dashed black) in x-direction (*left*) and y-direction (*right*).

The mapping result for the first pattern is presented in figure 3.2. At the beginning and the end of the trace one is able to observe the already assumed deviation (approx. $\Delta_{max} = 0.2 \text{ mm}$) caused by the use of just a single mapping polynomial for the whole active PSD surface. In the main section around the chip center however, the trace follows exactly the real beam movement. These specific characteristics can be found in both the trace along the x-axis as well as the one along the y-axis, so it is most likely not just a random effect. Therefore it can be stated that the attempts to compensate the coupling can be considered successful.

To investigate the impact of the mapping function deviation on the mapping result

further other patterns were employed. In the second test a rectangular pattern was used to determine the difference in the mapping quality (i.e. deviation) in the center of the chip and close to the edge of the photo-active surface. The results are presented in figure 3.3.

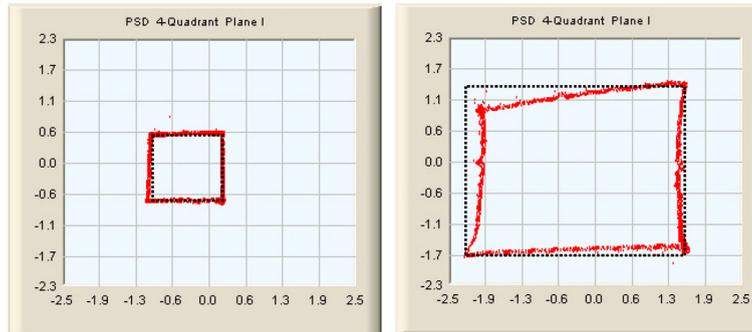


Figure 3.3: Trace of beam moving on PSD surface (red) in comparison to desired pattern (dashed black line). The small square trace (*left*) only covering the middle section of the active area almost perfectly follows the desired pattern (lines straight, 90° angles) while the trace of the bigger square covering the outside areas of the chip shows the expected curvature and deviation.

As expected and previously observed the mapping of the smaller pattern is significantly better than the result of the larger rectangle since the larger one predominantly covers the outer regions of the chip where the deviation of the mapping polynomial is bigger than close to the middle. Hence, the small rectangle shows perfect 90 degree angles with straight lines on either side, while in the bigger rectangle deviations in the angle and the curvature of the sides of the pattern are clearly visible.

To be able to determine the rate of change of the mapping quality when approaching the edge of photo-active area a meander and a triangle pattern was used for the next test run. Those two patterns are best suited for this purpose since they cover both the inner and the outer regions of the chip. The respective mapping results can be found in figure 3.4.

While the triangular pattern only shows the already familiar cushion shape in the lines that form the three sides, the meander pattern illustrates the steady change in deviation of the mapping from the actual beam position with increasing distance from the middle of the pattern. The most outer rectangular pattern resembles the same behavior as already presented in the big rectangular pattern before. The occasionally occurring “peak” in the signal (e.g. at $x = 0.9 \text{ mm}$, $y = -0.4 \text{ mm}$) are not caused by the mapping function but rather by movements caused by the human operator of the x,y-table. In such a complex pattern this is unfortunately inevitable since all patterns were generated by manual input.

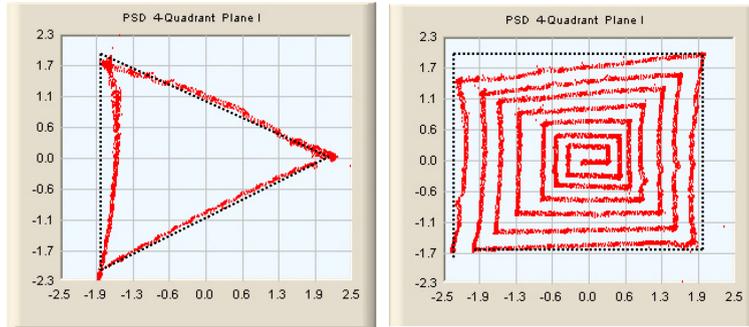


Figure 3.4: Additional traces of triangular (*left*) and meander pattern (*right*). Both patterns show the deviation of the signal from the actual beam movement when the outer regions of the chip are used.

3.3 Feedback Evaluation

Considering the results presented in the previous section the mapping results seem to be sufficient to perform a calibration task. The test showed that the characteristics that can be found in the mapping function (i.e. deviation at both ends, good in middle section) also hold true for the complete mapping result. The deviation of the mapped- from the actual location when the beam is positioned close to the outer edge of the active area is in this case acceptable. This is because the sole purpose of the controller is to position the laser close to the center of the chip. Therefore even the completely accurate mapping result will cause the controller (and thereby the robot arm) to reposition the laser towards the middle.

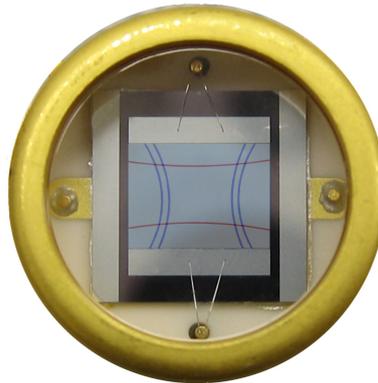


Figure 3.5: Schematic of a lateral PSD with lines separating the segments with different interpolation polynomials. The line for the fitting in y-direction are indicated in red while the respective lines in x-direction are marked blue. The resulting number of segments is 15. The middle segment is the largest (most linear area) with about 4mm^2 . This equals about 25% of the chips surface. The curvature of the lines shows the already familiar cushion pattern.

To broaden then field of application of the calibration system and increase the performance of the controller, it might become necessary to improve the mapping even

in those remote regions of the chip. In general there are two ways to do so: increase the degree of the polynomial to adapt to the curvature or increase the number of segments that have a separate mapping function. Since the first solution is no feasible due to the restrictions made for the mapping function earlier, the second option can be considered here. Tests indicated that the optimal trade-off of increase in complexity to increase in mapping quality can be found with approx. five regions in x-direction and three in y-direction resulting in 15 total regions on the chip. Figure 3.5 illustrates the position of the lines (polynomial with degree $n = 2$) separating the regions from each other. For each region a mapping function can be found in the same way as used before for the single mapping function. This way the global mapping is still unique for any pair of x- and y-output voltage.

3.4 Controller Design

In this paragraph the design process for the controller intended to move the robot arm in a way that the laser beams are centered over the PSD surfaces will be presented. First the control law describing the controller has to be deduced. Once the control law is obtained, the controller has to be altered to be able to utilize the PSD feedback to control the robot position (PSD-servoing).

3.4.1 PD plus Gravity Control Law

First of all, one has to decide what kind of control scheme will be best-suited for this specific task. In this case the computed-torque controller as presented in [Lewis 93, chapter 3.4] seemed to be the most promising. From this controller family the PD+G controller was chosen for this project.

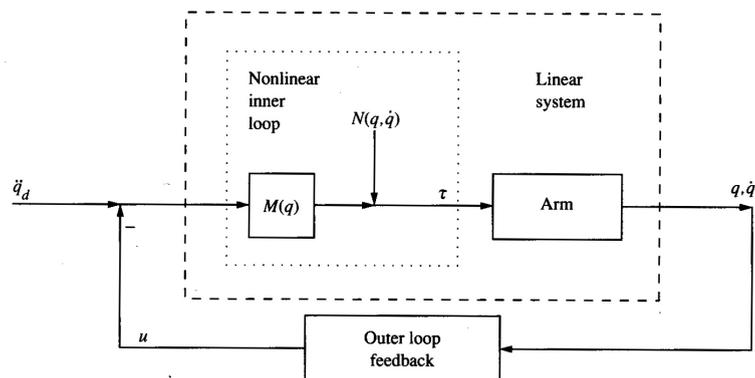


Figure 3.6: Graphic illustrating the control-scheme of computed-torque based approaches taken from [Lewis 93, page 127]. Note the inner non-linear loop that is nested in the outer linear loop including the robot arm.

In order to begin with this task first the general dynamics of a robot arm have to be discussed. A detailed derivation can be found at [Lewis 93, pages 64 ff.]. The general robot arm dynamics are given as (for symbol explanation see table below):

$$M(q)\ddot{q} + V(q, \dot{q}) + F_v\dot{q} + F_d + G(q) + \tau_d = \tau \quad (3.5)$$

This rather complex term can be simplified for writing convenience by merging the non-linear terms into the vector $N(q, \dot{q})$

$$N(q, \dot{q}) = V(q, \dot{q}) + F(\dot{q}) + G(q) \quad (3.6)$$

This yields

$$M(q)\ddot{q} + N(q, \dot{q}) + \tau_d = \tau \quad (3.7)$$

The tracking error can be defined as the deviation of the actual trajectory from the desired trajectory:

$$e(t) \stackrel{!}{=} q_d(t) - q(t) \quad (3.8)$$

Differentiating equation 3.8 and solving 3.7 for \ddot{q} will lead to

$$\ddot{e} = \ddot{q}_d + M^{-1}(N + \tau_d - \tau) \quad (3.9)$$

Defining the control input u as

$$u \stackrel{!}{=} \ddot{q}_d + M^{-1}(N - \tau) \quad (3.10)$$

yields the computed-torque control law:

$$\tau = M(\ddot{q}_d - u + N) \quad (3.11)$$

q	:	joint variables
$M(q)$:	inertia matrix
$V(q, \dot{q})$:	Coriolis / centripetal vector
F_v, F_d	:	viscous friction matrix and dynamic friction vector
$G(q)$:	gravity vector
τ	:	input vector
τ_d	:	disturbance vector
$e(t)$:	tracking error
$q(t), q_d(t)$:	actual- and desired trajectory

The control law characterizing the class of computed-torque-like controllers can be denoted as:

$$\tau = \widehat{M}(\ddot{q}_d - u + \widehat{N}) \quad (3.12)$$

The choice of \widehat{M} and \widehat{N} characterize the specific type of controller of this family. Since a PD plus gravity controller was chosen, they are set to $\widehat{M} = I$ and $\widehat{N} = G(q) - \ddot{q}_d$. This yields:

$$\tau_c = I(\ddot{q}_d - u) + G(q) - \ddot{q}_d = -u + G(q) \quad (3.13)$$

Selecting PD feedback for u will lead us to:

$$\tau_c = -K_v\dot{q} - K_pq + G(q) \quad (3.14)$$

with K_v and K_p denoting the positive scalar constant controller gains.

3.4.2 PSD-Servoing

Since the PSD feedback can't be used directly for the controller, one needs to “translate” the feedback from *sensor space* (voltage) to *robot space* (position) to achieve sufficient controller performance. This task is as mentioned before usually referred to as *visual servoing* and is a well known problem in robotics. In [Shen 02] and approach similar to the one required here can be found for uncalibrated visual servoing using cameras. Therefore this concept is adapted here to be used with PSD sensor output instead of a camera image. In this context we are dealing with two different coordinate frames: the end-effector frame (indicated by the index C) and the PSD frame (indicated by P). Hence, let us define the position of the end-effector relative to the PSD as ${}^C X_P$:

$${}^C X_P = (X, Y, Z, \gamma, \rho, \phi)^T \quad (3.15)$$

with X, Y, Z denoting the position and γ, ρ, ϕ the orientation as roll, pitch, and yaw. Applying forward kinematics, the velocity of the end-effector relative to the robot base frame can be expressed depending on the Jacobian matrix of the robot ($J(q)$) and a matrix expressing the rotation between the two coordinate frames A . Note that A consists of the unknown rotation matrices R (since system is uncalibrated).

$${}^C \dot{X}_P = \begin{bmatrix} R & 0 \\ 0 & R \end{bmatrix} \cdot J(q)\dot{q} = A \cdot J(q)\dot{q} \quad (3.16)$$

Since the end-effector is moved on a plane parallel to the PSD surface the change in position of the TCP (X_c, Y_c) will result in the same change in position of the laser beam location on the PSD (x, y). Hence:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} X_c \\ Y_c \end{bmatrix} \quad (3.17)$$

Therefore we can define the beam location on the PSD $\beta = [x, y]^T$ as

$$\beta = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \end{bmatrix} \quad (3.18)$$

Since the PSD is fixed relative to the TCP, the end-effector movement can be described as ${}^C \dot{X}_P = [T_x, T_y, T_z, \omega_x, \omega_y, \omega_z]^T$ where T denotes the three components of the translational velocity vector ${}^C L_P$ and ω denotes the components of the angular velocity vector ${}^C \Omega_P$. Therefore the velocity of a point $X = [X_x, X_y, X_z]^T$ fixed on the end effector can be expressed as:

$${}^C \dot{X} = {}^C \Omega_P \times {}^C X + {}^C L_P \quad (3.19)$$

Introducing the unit vectors pointing in the respective coordinate axis direction $\vec{i} = [1, 0, 0]^T$, $\vec{j} = [0, 1, 0]^T$, $\vec{k} = [0, 0, 1]^T$ one can write:

$${}^C\Omega_P \times {}^C X = \det \begin{bmatrix} \vec{i} & \vec{j} & \vec{k} \\ \omega_x & \omega_y & \omega_z \\ X_c & Y_c & Z_c \end{bmatrix} \quad (3.20)$$

Replacing the robot location by the respective TCP location (relation given in equation 3.17) will result in:

$${}^C\Omega_P \times {}^C X = \det \begin{bmatrix} \vec{i} & \vec{j} & \vec{k} \\ \omega_x & \omega_y & \omega_z \\ x & y & Z_c \end{bmatrix} = \omega_y Z_c \vec{i} + \vec{j} \omega_z x + \omega_x y \vec{k} - x \omega_y \vec{k} - \vec{j} \omega_x Z_c - \vec{i} \omega_z y \quad (3.21)$$

Substituting equation 3.18 in 3.19 and using the result found in 3.21 leads to:

$$\begin{aligned} \dot{X}_c &= \omega_y Z_c - \omega_z y + T_x \\ \dot{Y}_c &= \omega_z x - \omega_x Z_c + T_y \\ \dot{Z}_c &= 0 \end{aligned} \quad (3.22)$$

Notice that Z_c is an unknown but fixed constant since system is uncalibrated. The equation above can be rewritten using 3.18:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = J_P \begin{bmatrix} T_x \\ T_y \\ T_z \\ \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \quad (3.23)$$

Solving this equation for the Jacobian of one point in the end-effector yields:

$$J_P = \begin{bmatrix} 1 & 0 & 0 & 0 & Z_c & -y \\ 0 & 1 & 0 & -Z_c & 0 & x \end{bmatrix} \quad (3.24)$$

In order to be able to solve the 6 DOF calibration problem at least three points have to be observed. The required Jacobian matrix can now be constructed by simply stacking the Jacobian matrices of the three points. Therefore we get:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{y}_1 \\ \dot{x}_2 \\ \dot{y}_2 \\ \dot{x}_3 \\ \dot{y}_3 \end{bmatrix} = J_S \begin{bmatrix} T_x \\ T_y \\ T_z \\ \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \Leftrightarrow \dot{\xi} = J_S \cdot^C \dot{X}_P \quad (3.25)$$

Hence, this leads us to

$$J_S = \begin{bmatrix} 1 & 0 & 0 & 0 & Z_{c1} & -y_1 \\ 0 & 1 & 0 & -Z_{c1} & 0 & x_1 \\ 1 & 0 & 0 & 0 & Z_{c2} & -y_2 \\ 0 & 1 & 0 & -Z_{c2} & 0 & x_2 \\ 1 & 0 & 0 & 0 & Z_{c3} & -y_3 \\ 0 & 1 & 0 & -Z_{c3} & 0 & x_3 \end{bmatrix} \quad (3.26)$$

Notice that all values for T_z in this matrix (third column) are equal to zero. A change in these values however does not affect the relation between beam point velocity and the TCP velocity in their respective frames. Based on this fact we can change the values in this column to values $k_i, i \in [1, \dots, 6]$ with $k_i \neq 0$ and $k_i \neq k_j$ for $i \neq j$ to avoid the singularity of J_S .

$$J_{S'} = \begin{bmatrix} 1 & 0 & k_1 & 0 & Z_{c1} & -y_1 \\ 0 & 1 & k_2 & -Z_{c1} & 0 & x_1 \\ 1 & 0 & k_3 & 0 & Z_{c2} & -y_2 \\ 0 & 1 & k_4 & -Z_{c2} & 0 & x_2 \\ 1 & 0 & k_5 & 0 & Z_{c3} & -y_3 \\ 0 & 1 & k_6 & -Z_{c3} & 0 & x_3 \end{bmatrix} \quad (3.27)$$

Combining equation 3.25 and 3.16 yields

$$\dot{\xi} = J_{S'} A J(q) \dot{q} \quad (3.28)$$

Using the fact that $\det(J_{S'}) \neq 0$ (i.e. $J_{S'}$ is non-singular according to the weak condition) and solving the equation above for \dot{q} results in

$$\dot{q} = J^+(q) A^T J_{S'}^+ \dot{\xi} \quad (3.29)$$

$$\widehat{A}^T = \begin{bmatrix} \widehat{R}^T & 0 \\ 0 & \widehat{R}^T \end{bmatrix} \quad (3.30)$$

Considering the earlier found control law for the PD plus gravity controller 3.13 and introducing the PSD beam location deviation in the PSD frame $\Delta\xi = \xi - \xi_d$ the control law including PSD servoing can be rewritten as:

$$\tau_c = G(q) - K_v \dot{q} - K_p J^T(q) \begin{bmatrix} \widehat{R}^T & 0 \\ 0 & \widehat{R}^T \end{bmatrix} \widehat{J}_{S'}^T \Delta\xi \quad (3.31)$$

Substituting equation 3.31 into the equation describing the general arm dynamics (3.5) while disregarding the terms describing friction and disturbance yields:

$$M(q)\ddot{q} + V(q, \dot{q})\dot{q} = -K_v\dot{q} - K_p J^T(q) \widehat{A}^T \widehat{J}_{S'}^T \Delta\xi \quad (3.32)$$

Using *proposition 1* from [Shen 02, page 583] on the right side results in:

$$\begin{aligned} M(q)\ddot{q} + V(q, \dot{q})\dot{q} &= -K_v\dot{q} - K_p J^T(q) A^T J_{S'}^T \Delta\xi - K_p J^T(q) \left(\widehat{A}^T \widehat{J}_{S'}^T - A^T J_{S'}^T \right) \Delta\xi \\ &= -K_v\dot{q} - K_p J^T(q) A^T J_{S'}^T \Delta\xi - K_p J^T(q) Y(\Delta\xi, x, y) \Delta\Theta \end{aligned} \quad (3.33)$$

$Y_1(J(q)\dot{q}, x, y)$ denotes a regression matrix (independent of any unknown elements). Further Θ is defined as a $m \times 1$ (m is number of unknown variables) vector combining the unknown elements in A^T and $J_{S'}^T$. The respective estimation of Θ is denoted $\widehat{\Theta}$. To achieve the desired result an adaptive law becomes inevitable to improve the estimations of \widehat{A} and $\widehat{J}_{S'}$. Using equation 3.28 one can state:

$$\begin{aligned} \widehat{J}_{S'} \widehat{A} J(q) \dot{q} - \dot{\xi} &= \left(\widehat{J}_{S'} \widehat{A} - J_{S'} A \right) J(q) \dot{q} \\ &= Y_1(J(q)\dot{q}, x, y) \Delta\Theta \end{aligned} \quad (3.34)$$

The adaptive law can be denoted as:

$$\begin{aligned} \Delta\dot{\Theta}^T &= - \left[\widehat{J}_{S'} \widehat{A} J(q) \dot{q} - \dot{\xi} \right]^T B_2 Y_1(J(q)\dot{q}, x, y) + \frac{1}{B_1} \dot{q}^T K_p J^T(q) Y(\Delta\xi, x, y) \\ &= \Delta\Theta^T Y_1^T(J(q)\dot{q}, x, y) \cdot B_2 Y_1(J(q)\dot{q}, x, y) + \frac{1}{B_1} \dot{q}^T K_p J^T(q) \\ &\quad Y(\Delta\xi, x, y) \end{aligned} \quad (3.35)$$

Here B_1 and B_2 are positive gains. Using *theorem 1* ([Shen 02, page 583]) it can be stated that the properties of the proposed controller guarantees stability and convergence as also shown in the simulation results of the controller including servoing in the next paragraph.

3.5 Controller Simulation

Before the controller can be used on the real robot it is reasonable to simulate the behavior first before actually applying it to the real robot hardware. Not only is it easier to tweak the test different controller setups (i.e. gains) for various initial conditions (e.g. initial positional error), but also is it safer since a wrong setup could cause damage or even destruction of the robot arm. Hence, MatLab code simulating the controller and a simplified (planar) robot arm was used for this purpose. In this case basically two variables are of interest: the positional error and the joint velocity error.

In figure 3.7 the positional error (i.e. deviation of the beam from the center position) is presented. The performance of the controller at this point is crucial since it will directly influence the level of accuracy of the calibration system decisively. Here the simulated deviation of the beam position from the PSD center is illustrated for a given initial positional error $\Delta x = 0.2 \text{ mm}$ and $\Delta y = 0.6 \text{ mm}$.

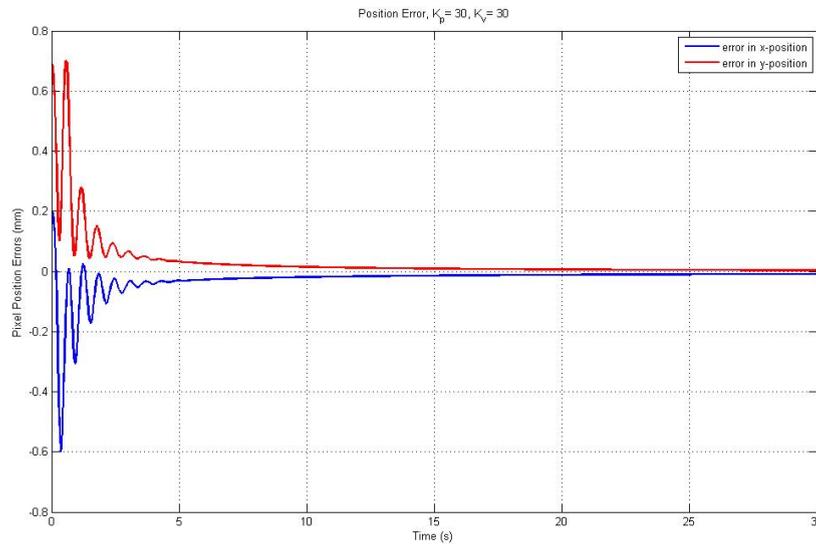


Figure 3.7: Simulation result for the proposed controller depicting the 2D resulting positional error of the controller guided system. The given initial position (e.g. $\Delta x = 0.2 \text{ mm}$ $\Delta y = 0.6 \text{ mm}$) is corrected by the controller to ensure a resulting centered position. As the graph shows, the static error decreases rapidly ($28 \mu\text{m}$ at $t \leq 5 \text{ sec}$) with increasing time.

At the very beginning the controller causes the plant (i.e. robot arm) to perform significant oscillation in order to reduce the positional error of both the x- and y-axis. These oscillations decay after a short period of time ($t \leq 3 \text{ sec}$) and a steady convergence toward zero can be observed. The speed of convergence can be estimated considering the absolute deviations at certain points in time. While at $t = 1 \text{ sec}$ the deviation is still approx. $120 \mu\text{m}$, it drastically declines. At $t = 5 \text{ sec}$ it is only approx. $28 \mu\text{m}$ and at $t = 30 \text{ sec}$ it nearly reached zero ($\leq 7 \mu\text{m}$).

Figure 3.8 illustrates the joint velocity error of a simulated two jointed planar robot for a controller simulation with the same parameters as before.

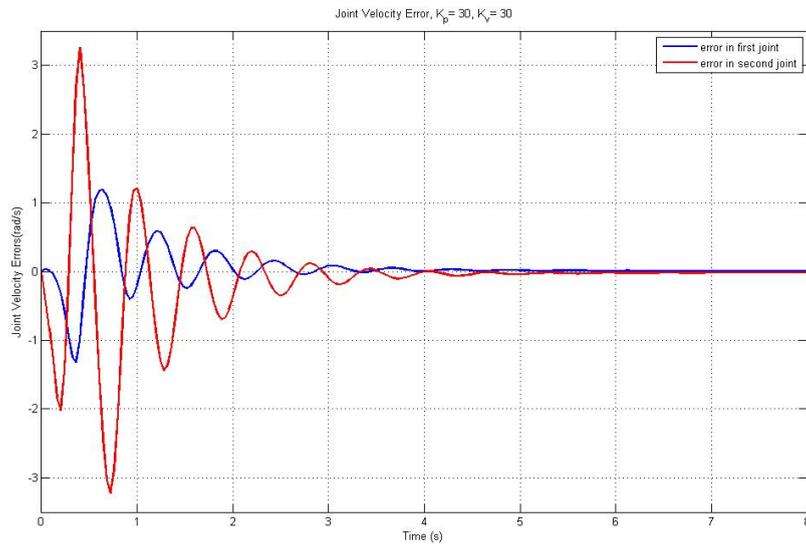


Figure 3.8: Graph representing the simulation of the joint velocity error of a planar robot with two joints. The initial significant oscillation rapidly decreases and converges to approx. zero for $t \geq 6$ sec.

In correspondence with the positional error the joint velocity error initially shows heavy oscillation ($t \leq 3$ sec) changing into the already familiar decline towards zero. Therefore it can be remarked that a steady state has been reached.

Reviewing the results on can state that for a reasonable settling time, the simulated robot arm asymptotically shows the desired behavior (positional error $\rightarrow 0$ for $t \rightarrow \infty$ with joint velocity error $\rightarrow 0$). Since the calibration task is not time critical, the selected positioning time of 30 seconds seems reasonable and can even be extended if the level of accuracy needs to be improved beyond $5 \mu m$ for this part.

4. The Calibration System

No matter what principles is that calibration system are base on (e.g. optical, mechanical etc.),they all have the same basic idea in common: An object or point in the robots task space with known properties (i.e. position, orientation, size etc.) is used to perform a calibration of the robot system. This is achieved by comparing the known properties with the data measured for this object or point by the robot. The data considered for this depends on the specific kind of system used. A camera system for example could be used to measure the size and position of a certain known object in the camera's field of view. This can then be used to compare the computed position (extracted from the captured image) with the robot's position given by the joint encoder values and forward kinematics. Based on this comparison the deviation of the measured position from its intended position can be determined. If this is done for several points, the pose error of the robot can be determined. By modifying the robot controller software one is then able to compensate the error and achieve better positional accuracy in the future. Some examples of various approaches in the past that use methods that are somehow related to the one that is presented in this report, can be found in [van Albada 93], [Meng 01], and [Gong 00]. However a system that uses the same principle could not be found in the reviewed literature. A general overview and introduction to robot calibration can be found in [Elatta 04], [Shiakolas 02], and [Zhuang 96]. All these calibration systems however share several major disadvantages if one considers their application in the field of industrial robotics:

- high complexity
- high costs
- time consuming

Furthermore, almost all of the systems proposed by various authors require lab conditions (i.e. expensive instruments, highly skilled personnel etc.). This however is opposed to the scenario of a calibration solution for lager quantities of robots

as used in industrial production processes. Hence, from an economic point of view, almost all currently available calibration methods are highly inefficient and therefore not suitable for any company. This is because it will dramatically decrease the profit margin. Therefore a new system is needed that satisfies the following criteria:

- applicable to perform on-the-shop-floor calibration
- low acquisition and operating costs
- ensure low robot downtime
- robust to external disturbance commonly found in production lines (e.g. magnetic fields)
- easy to operate for user (minimize human interaction in calibration process)
- portable from one robot workcell to another

To make sure we meet the given requirements the most common forms of calibration systems (i.e. camera or gaging equipment based systems) have to be discarded since they are expensive, slow, and too complex. They are suited for research related applications since these usually only require the calibration procedure to be performed once and only on very few robots. In an industrial production environment however, they would increase downtimes and production costs dramatically. This is the reason why an approach based on PSDs (as introduced in section 1.2) was chosen for this project. In the following sections the components designed for this system (i.e. laser fixture, calibration fixture housing the PSDs) are introduced (section 4.3.2) and the theoretical background, the system is based on will be provided (section 4.1).

4.1 Concept

As described in the previous section the basic idea of calibration is the comparison of robot external (world coordinates) and robot internal (joint encoder values) representation of a certain location in the task space to determine the error. In order to perform calibration, the external representation has to be known a priori. One of the guiding aspects in the design process was that the system has to be portable. This indicates however, that it is not feasible to fix the calibration fixture to the workcell floor and measure its location (also error-prone). Therefore an alternative way to determine the points relative to the world coordinate system without a fixed location in the workcell has to be found. Therefore it is necessary to put a lot of thought into the design process.

Let's assume a local coordinate system is established on the lower right edge of the calibration fixture (see figure 4.1). Since as discussed above one can not establish a fixed connection of the device in the external world coordinate system one must consider other options. The world coordinate system (WCS) can basically be placed anywhere since neither the origin nor the orientation are restricted in any way. In this case, the best and also most convenient location for the WCS would be the origin of the calibration fixture coordinate system (CFCS). This way further

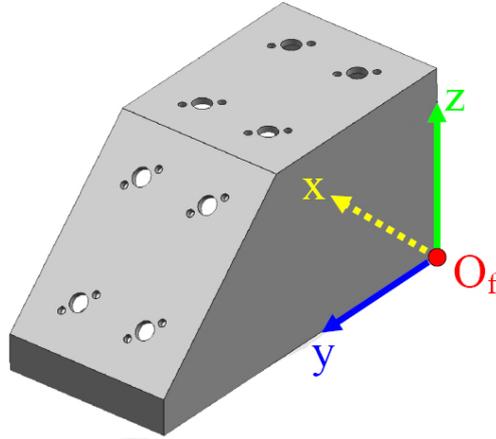


Figure 4.1: Sketch of the calibration fixture used for experiments holding the PSDs. One can see the front plane (angled down) and the top plane (parallel to ground) with the drill holes that will house the PSDs. On the bottom right one can see the calibration fixture coordinate system (CFCS). The x,y -plane forms the bottom of the fixture, while the y,z -plane limits the fixture to the right.

complex transformations from CFCS to WCS are not required. Another enjoyable side-effect of this WCS position is that this source of positional error can therefore be eliminated in advance. The trade-off for this however is that one has to come up with a way to determine the PSD center position relative to the calibration fixture coordinate system's (CFCS) origin O_f and its coordinate axis $\vec{x}, \vec{y}, \vec{z}$ (see figure 4.1). Furthermore, a way to relate this to the robot has to be found, too. This can be simultaneously achieved using the calibration system designed for this project.

The basic idea of this system is to match four laser beams (mounted on the robot arm near the TCP) to the centers of the PSDs fixed to the calibration fixture in groups of four. This is done in order to be able to relate the robots internal point representation (i.e. laser beam intersection point, see section 4.1.2) to the external calibration fixture coordinate system representation (i.e. virtual intersection point, see section 4.1.1). The position of the each PSD is fixed and known (after the device is designed, manufactured, and initially calibrated) in CFCS coordinates. Since as mentioned before the CFCS and the WCS share the same origin and orientation of the coordinate axis, the location is also know in WCS.

4.1.1 Virtual Intersection Points

To construct the virtual intersection points let us assume that the center of each PSD can be considered the origin of a virtual emitted beam (represented by a vector) that is perpendicular to the PSD surface and aims towards the fixture's bottom plane (see figure 4.2).

A virtual beam can be mathematically described as an ordinary line that follows the general equation below with the restriction that i -th line has to intersect with the center of the i -th PSD C_i . Furthermore the line must be perpendicular to the surface plane of the PSD.

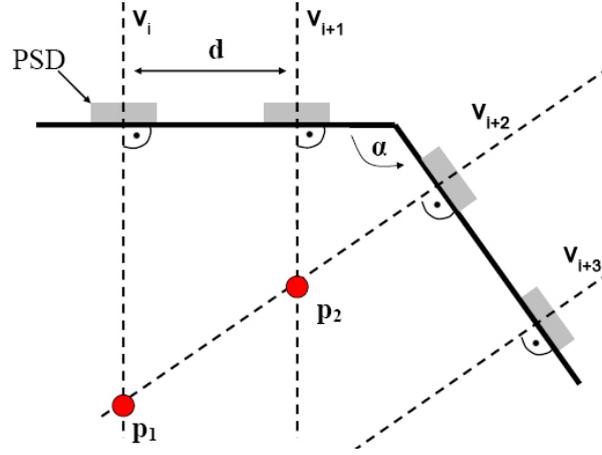


Figure 4.2: Schematic lateral view of the calibration fixture with the top and side plane (facing down). The virtual lines trough the PSD centers are marked as dashed lines. Two intersection points p_1, p_2 of the beams v_i, v_{i+1}, v_{i+2} are shown exemplarily to illustrate the concept of virtual intersection points. The fixed distance between the PSDs is denoted as d while the angle between the two PSD planes is marked as α .

$$\vec{v}_i := \vec{m}_i \cdot x + \vec{t}_i \quad (4.1)$$

- v_i : i-th virtual beam
- m_i : gradient of the i-th beam
- x : variable $x \in \mathbb{R}$
- t_i : displacement component of the i-th beam

The virtual intersection point of two virtual beams (i.e. lines) can be found by equating the right sides of the beam equations (equation 4.1) and solving of the resulting system. Since that would be a problem involving 12 unknown variables (3 for each gradient and displacement vector) and due to the fact that additional knowledge of the fixture geometry (see figure 4.3) is available, one can simplify the process. The problem can be solved a lot easier by using trigonometric functions instead. The fact that a right-angled triangle is formed allows it to find the intersection point location p_w using trigonometric functions. In this case the angle β (see figure 4.3) and it's adjacent leg are known. Since the fixture device is symmetric, β is equal to $\frac{\alpha}{2}$ (see figure 4.2). Let us consider the top plane to be parallel to the bottom plane (as required before). In this case it is also parallel to the CFCS x,y-plane. Therefore one can calculate the three-dimensional location of the virtual intersection point according to:

$$p_w^x = \Delta x_{edge} \quad (4.2) \quad p_w^z = \Delta z_{edge} - d \cdot \tan \frac{\alpha}{2} \quad (4.4)$$

$$p_w^y = \Delta y_{edge} \quad (4.3)$$

- Δx_{edge} : x-location of the edge between top and side plane in CFCS
- Δy_{edge} : y-location of the edge between top and side plane in CFCS
- Δz_{edge} : z-location of the edge between top and side plane in CFCS
- d : distance between the PSD center and the edge of the plane
- α : angle between top and side plane (see figure 4.2)

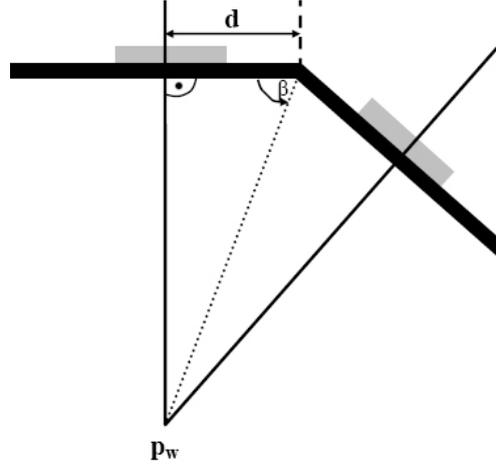


Figure 4.3: Zoom into figure 4.2. Again top and side plane are displayed. Since the length of side d and the angle $\beta = \frac{\alpha}{2}$ are known of this right-angled triangle, one can use trigonometric functions to find the location of p_w .

In order to get the specific x,y,z-location additional considerations have to be made. The plane edge can be modeled as a single line since its width can be disregarded for our purposes. After this a second line is introduced that connects the two closest PSD centers on the left and right side (top- and bottom plane) of the edge. Therefore the needed coordinates (i.e. $x_{edge}, y_{edge}, z_{edge}$) can be calculated by solving the intersection problem of these two lines.

4.1.2 Laser Beam Intersection points

Now that the external calibration points are obtained (see section 4.1.1) one has to determine a way to acquire the corresponding points in robot internal measurements called laser beam intersection points ¹. In this project this is achieved by using four laser beams emitted by laser pointers fixed in the proximity of the robot's TCP. The idea is to align them with the centers of the PSDs to acquire laser beams that correspond with the virtual beams introduced in the previous section. This is necessary to construct the intersection points. Hence, the laser beam position is given for each respective laser pointer in a local coordinates system called the laser fixture frame coordinate system (LFCS) originating from the lower right edge of the laser fixture (see figure 4.4).

¹ Of course these points are virtual intersection points again since the laser beam is either absorbed or reflected by the PSD surface. In order to reduce confusion, these points will be addressed as "laser beam intersection point". The term "virtual intersection points" will only be used for the calibration fixture.

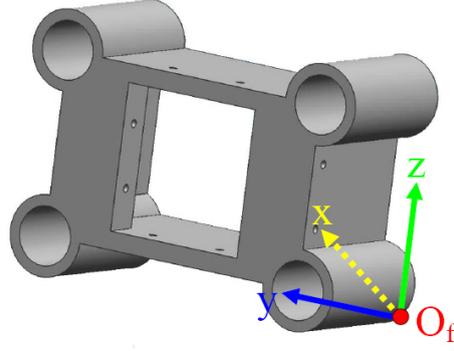


Figure 4.4: Draft of the fixture that holds the laser pointers. The fixture is mounted in the proximity of the TCP. The lasers will be mounted into the round cut-outs on either corner of the fixture. The fixture ensures that all laser beams are parallel to each other once they are put in place and calibrated. On the lower right the laser fixture frame coordinate system (LFCS) with its origin O_f can be seen.

In order to find the beam intersections one has to solve an intersection problem similar to the virtual intersection points of the calibration fixture (see figure 4.2). The laser beams can be mathematically modeled as straight lines since they meet the required properties. Attenuation and other laser related phenomena such as widening or aberration can be disregarded because of evanescently influence in this context. As line origin one can assume the center of the respective laser pointer fixed to the robot. The directional vector is also known because the emitted beam is parallel to the axis of the robot the fixture is mounted on. Hence, the line is fully defined and one can proceed with the computation of the beam intersection points. Let us assume two straight lines (similar to 4.1):

$$\vec{v}_i = \vec{m}_i \cdot x + \vec{t}_i \quad (4.5) \quad \vec{v}_{i+1} = \vec{m}_{i+1} \cdot x + \vec{t}_{i+1} \quad (4.6)$$

The intersection problem can be found by equating the two lines: $\vec{v}_i \stackrel{!}{=} \vec{v}_{i+1}$. Solving this problem results in the coordinates of the intersection point $p_b = (p_b^x, p_b^y, p_b^z)^T$ in the LFCS.

$$p_b^x = \frac{m_{i,x}t_{i+1,x} - m_{i+1,x}t_{i,x}}{m_{i,x} - m_{i+1,x}} \quad (4.7) \quad p_b^z = \frac{m_{i,z}t_{i+1,z} - m_{i+1,z}t_{i,z}}{m_{i,z} - m_{i+1,z}} \quad (4.9)$$

$$p_b^y = \frac{m_{i,y}t_{i+1,y} - m_{i+1,y}t_{i,y}}{m_{i,y} - m_{i+1,y}} \quad (4.8)$$

$$\begin{aligned} p_b^x, p_b^y, p_b^z &: \text{coordinates of laser beam intersection points} \\ m_{i,x} &: \text{x-coordinates of vector } \vec{m}_i \end{aligned}$$

Now that both the virtual intersection points and the laser beam intersection points are constructed they have to be related to each other to perform the calibration task. In order to be able to do so, they have to be transformed into a common coordinate

system (i.e. WCS). The virtual intersection points are already given in WCS so only the laser beam intersection points have to be transformed. This can be achieved using the coordinate transformation methods presented in the next section.

4.1.3 Coordinate Transformation

4.1.3.1 General Coordinates Transformation

In 3D Cartesian space a transformation from one coordinate system into another can sufficiently described using a 4x4 matrix, representing the rotation and displacement of the system's origin and axis relative to each other. Hence, the transformation can be performed by finding the proper transformation matrix T consisting of the rotation matrix R and the displacement vector \vec{d} . For a transformation ${}^{i-1}T_i$ from a coordinate system $i - 1$ into another system i one can apply the following homogeneous transformation equations that can be found in several books, papers, and articles (see [Lewis 93] for more details):

$$\begin{pmatrix} \vec{p}_{x_i, y_i, z_i} \\ 1 \end{pmatrix} = {}^{i-1}T_i \cdot \begin{pmatrix} \vec{p}_{x_{i-1}, y_{i-1}, z_{i-1}} \\ 1 \end{pmatrix} \quad (4.10)$$

$${}^{i-1}T_i = \begin{pmatrix} R_{3x3}(\alpha, \beta, \gamma) & d_{3x1}(x_{i-1}, y_{i-1}, z_{i-1}) \\ \vec{0} & 1 \end{pmatrix} \quad (4.11)$$

Wherein the resulting rotation matrix R can be decomposed into three single rotations along each of the coordinate axis:

$$R_{3x3}(\alpha, \beta, \gamma) = R_x(\alpha) \cdot R_y(\beta) \cdot R_z(\gamma) \quad (4.12)$$

with the components

$$\begin{aligned} Rot_x(\alpha) &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha \\ 0 & \sin \alpha & \cos \alpha \end{pmatrix} & Rot_z(\gamma) &= \begin{pmatrix} \cos \gamma & -\sin \gamma & 0 \\ \sin \gamma & \cos \gamma & 0 \\ 0 & 0 & 1 \end{pmatrix} \\ Rot_y(\beta) &= \begin{pmatrix} \cos \beta & 0 & \sin \beta \\ 0 & 1 & 0 \\ -\sin \beta & 0 & \cos \beta \end{pmatrix} \end{aligned}$$

The displacement on the other side can be expressed as a simple vector:

$$d_{3x1}(x_{i-1}, y_{i-1}, z_{i-1}) = \begin{pmatrix} d_x \\ d_y \\ d_z \end{pmatrix} \quad (4.13)$$

Considering equation 4.12 one can determine the rotation matrix $R_{3x3}(\alpha, \beta, \gamma)$ and finally the transformation matrix ${}^{LFCS}T_{TCPCS}$ (note that sin and cos have been

replaced with s and c in order to be able to display the complete matrix in a more compact way):

$$R_{3x3}(\alpha, \beta, \gamma) = \begin{pmatrix} c(\beta)c(\gamma) & -c(\beta)s(\gamma) & s(\beta) \\ s(\alpha)s(\beta)c(\gamma) + c(\alpha)s(\gamma) & -s(\alpha)s(\beta)s(\gamma) + c(\alpha)c(\gamma) & -s(\alpha)c(\beta) \\ -c(\alpha)s(\beta)c(\gamma) + s(\alpha)s(\gamma) & c(\alpha)s(\beta)s(\gamma) + s(\alpha)c(\gamma) & c(\alpha)c(\beta) \end{pmatrix}$$

$${}^{LFCS}T_{TCPCS} = \begin{pmatrix} & & d_x \\ & R_{3x3}(\alpha, \beta, \gamma) & d_y \\ 0 & 0 & d_z \\ & & 0 & 1 \end{pmatrix} \quad (4.14)$$

- \vec{p}_{x_i, y_i, z_i} : vector \vec{p} in coordinate system (x_i, y_i, z_i)
- ${}^{i-1}T_i$: transformation from coordinate system $(x_{i-1}, y_{i-1}, z_{i-1})$ to (x_i, y_i, z_i)
- $R_x(\alpha)$: matrix describing rotation around x-axis with angle α
- $R_y(\beta)$: matrix describing rotation around y-axis with angle β
- $R_z(\gamma)$: matrix describing rotation around z-axis with angle γ
- $R_{3x3}(\alpha, \beta, \gamma)$: resulting rotation matrix
- $d_{3x1}^{\vec{}}(x_{i-1}, y_{i-1}, z_{i-1})$: displacement vector
- d_x, d_y, d_z : displacement in x-, y-, and z-direction
- $s(\vartheta), c(\vartheta)$: $= \sin(\vartheta), \cos(\vartheta)$

4.1.3.2 Quaternion-Based Algorithm

In order to perform a transformation as presented in the previous section, all parameters (rotations and displacement vector) have to be known. Due to the fact that in this project a calibration procedure is to be performed, the simple approach of using an approximation for the parameters is not acceptable. This is due to the fact that the resulting accuracy level is highly dependent on those parameters. Therefore another approach offering more precision has to be found: A quaternion-based general transformation algorithm ([Zhuang 96] pp. 247-251). This algorithm is going to be presented in this section. Application of this method allows it to compute the proper transformation without any additional knowledge (e.g. rotation angles, point of origin, robot geometry, joint parameters etc.). The only input required is a sufficient number of (matching) measured point sets in the two coordinate systems in question. The minimum amount of points for this algorithm is three points. Here this condition can be satisfied by using the eight virtual intersection points of the calibration fixture that are given and the corresponding eight laser beam intersection points. The complete deduction is both too long and too complex, therefore only the most relevant steps will be discussed here.

To be able to fully understand the following equation transformations one has to be familiar with quaternions algebra. A brief overview can be found in [Zhuang 96] pp.220-222. Based on the general transformation equation as given in equation 4.10 one can conclude for a point p given in the coordinate systems i and $i + 1$ ($i \in \mathbb{R}$):

$$p^{\vec{i}+1} = R \cdot p^{\vec{i}} + \vec{t} \Leftrightarrow R \cdot p^{\vec{i}} = p^{\vec{i}+1} - \vec{t} \quad (4.15)$$

Using the equation $R \cdot p^{\vec{i}} = q \circ p^{\vec{i}} \circ q^*$ based on quaternion algebra yields:

$$\Leftrightarrow q \circ p^{\vec{i}} \circ q^* = p^{\vec{i}+1} - \vec{t} \Leftrightarrow q \circ p^{\vec{i}} - (p^{\vec{i}+1} - \vec{t}) \circ q = 0 \quad (4.16)$$

After expanding of the quaternion products above using the definition

$$a \circ b = \left[a_0 b_0 - \vec{a} \cdot \vec{b}, (a_0 \vec{b} + b_0 \vec{a} + \vec{a} \times \vec{b})^T \right]^T \quad (4.17)$$

and an alternative way to denote quaternions ($q = [q_0, \vec{q}]$) equation 4.16 can be written as

$$\Leftrightarrow [\vec{q} \cdot p^{\vec{i}}, q_0 \cdot p^{\vec{i}} - p^{\vec{i}} \times \vec{q}] = [\vec{q} \cdot (p^{\vec{i}+1} - \vec{t}), q_0 (p^{\vec{i}+1} - \vec{t}) - \vec{q} \times (p^{\vec{i}+1} - \vec{t})] \quad (4.18)$$

This however, is equivalent to the equation below considering $(p^{\vec{i}} + (p^{\vec{i}+1} - \vec{t})) \cdot \vec{q} = 0$

$$(p^{\vec{i}} + (p^{\vec{i}+1} - \vec{t})) \times \vec{q} = q_0 (p^{\vec{i}} - (p^{\vec{i}+1} - \vec{t})) \quad (4.19)$$

For the matrix R to be a rotational matrix the *Euler* parameters (\vec{k}, Θ) are introduced to the equation as $q = \left[q_0 = \cos\left(\frac{\Theta}{2}\right), \vec{q} = \sin\left(\frac{\Theta}{2}\right) \cdot \vec{k} \right]$. Note that \vec{k} is the rotation axis with the property $R \cdot \vec{k} = \vec{k}$ (i.e. k is not affected by the rotation) and Θ is the rotation angle. This yields

$$\tan\left(\frac{\Theta}{2}\right) \cdot (p^{\vec{i}+1} - \vec{t} + p^{\vec{i}}) \times \vec{k} = \vec{p} - p^{\vec{i}+1} + \vec{t} \quad (4.20)$$

Using a skew symmetric matrix $\Omega(\vec{\nu})$ one is able to avoid the computation of the cross product that is rather complex. Instead the operation can be simplified to a multiplication introducing:

$$\Omega(\vec{\nu}) = \begin{pmatrix} 0 & -\nu_z & \nu_y \\ \nu_z & 0 & -\nu_x \\ -\nu_y & \nu_x & 0 \end{pmatrix} \quad (4.21)$$

Therefore, equation 4.20 can be transformed to determine the vector \vec{x} : $\vec{x} = -\tan\left(\frac{\Theta}{2}\right) \vec{k}$ depending on the rotation parameters k, Θ

$$\Omega \left(p_j^{i+1} + p_j^i \right) \cdot \vec{x} - \Omega \left(\vec{t} \right) \cdot \vec{x} - \vec{t} = p_j^i - p_j^{i+1} \quad (4.22)$$

Let us assume that $j = 1, \dots, m$ measured point-sets are available and $a_j = p_j^{i+1} - p_{j+1}^{i+1}$, $b_j = p_j^i - p_{j+1}^i$. This will lead to

$$\Omega \left(a_j + b_j \right) \vec{x} = a_j - b_j \quad (4.23)$$

The resulting system $A \cdot \vec{x} = B$ can be solved if a minimum of three point-sets ($j, j+1, j+2$) is available. To improve the resulting transformation however it is beneficial to have more than three data sets to reduce the error caused by e.g. noise and improve the numeric stability.

$$A = \begin{pmatrix} \Omega(a_1 + b_1) \\ \Omega(a_2 + b_2) \end{pmatrix} \quad (4.24) \quad B = \begin{pmatrix} a_1 - b_1 \\ a_2 - b_2 \end{pmatrix} \quad (4.25)$$

The solution of the system will result in a vector \vec{x} . Therefore one can now determine values for \vec{k} and Θ using a least square fitting method:

$$\vec{k} = \frac{\vec{x}}{\|\vec{x}\|} \quad (4.26) \quad \Theta = 2 \arctan \left(\frac{x_{max}}{k_{max}} \right) \quad (4.27)$$

Using $s(\Theta)$, $c(\Theta)$ instead of $\sin(\Theta)$, $\cos(\Theta)$ and $\psi(\Theta)$ instead of $1 - \cos(\Theta)$ for writing and reading convenience, the rotation matrix R follows as :

$$R = \begin{pmatrix} k_x^2 \psi(\Theta) + c(\Theta) & k_x k_y \psi(\Theta) - k_z s(\Theta) & k_x k_y \psi(\Theta) + k_z s(\Theta) \\ k_x k_y \psi(\Theta) + k_z s(\Theta) & k_y^2 \psi(\Theta) + c(\Theta) & k_y k_z \psi(\Theta) - k_x s(\Theta) \\ k_x k_z \psi(\Theta) - k_y s(\Theta) & k_x k_z \psi(\Theta) + k_x s(\Theta) & k_z^2 \psi(\Theta) + c(\Theta) \end{pmatrix} \quad (4.28)$$

Note that this matrix is only dependent on one rotation angle with the respective rotation axis. This is sufficient however since any 3D rotation can be expressed as a single rotation around one axis with one angle in 3D space uniquely. This is expressed in Euler's theorem. Now that the rotational contribution to the transformation is known one still has to determine the displacement vector \vec{t} . With R known, equation 4.15 can be used to determine a solution using a least square method for m measured points:

$$\vec{t} = \frac{1}{m} \sum_{j=1}^m w_j \left(p_j^{i+1} - R p_j^i \right) \quad (4.29)$$

- R : matrix describing the rotational transformation
- \vec{t} : vector describing the displacement
- p_j^i : j -th measured point in coordinate system i
- q : quaternion variable that can be expressed as $q = [q_0, \vec{q}]$
- Θ : rotation angle in R
- \vec{k} : rotation axis in R

Now that one is able to extract all the information required to compose the transformation matrix (equation 4.11 using rotation matrix found in equation 4.28 and displacement vector found in 4.29), there is only one problem left not yet dealt with. Since the objective is to relate the RBCS to the WCS the laser beam intersection points have to be converted from LFCS into RBCS. Hence, the calibration will be performed in two cycles. The first cycle is used to relate the LFCS to the RBCS while the second one relates the RBCS to the WCS. Therefore one is also able to relate the LFCS directly to the WCS, as required for the calibration task that can now be performed.

4.2 Calibration Algorithm

In the following section an algorithm will be proposed and discussed that can be used to perform the calibration task employing the hardware and methods developed for this project. In the first paragraph a general overview is presented, while detailed information about the algorithm can be found in the following sections.

4.2.1 Overview

As presented in figure 4.5 the calibration process can be decomposed into two cycles. Each of these cycles consists of three parts that are identical for both runs: positioning, calibration, and the precision test. A detailed flow chart of each step can be found in figures 4.6 to 4.8.

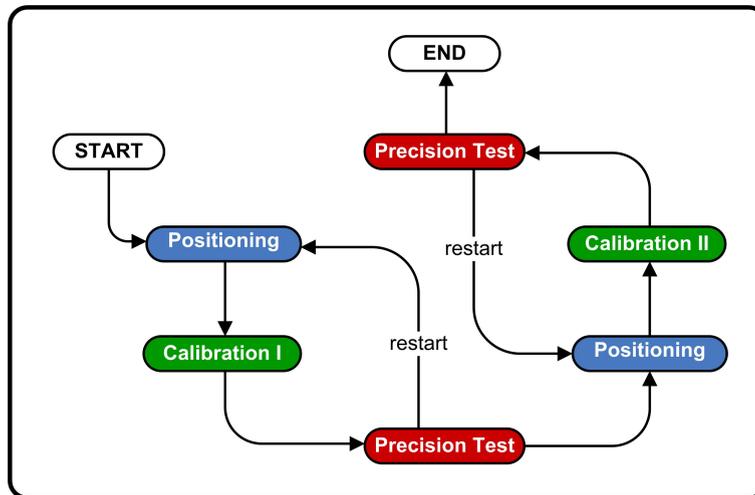


Figure 4.5: Diagram presenting the overview of the complete calibration algorithm consisting of two calibration stages. In the first stage the laser frame (TCP proximity) is calibrated relative to the robot base frame. After this is done, the second stage of the process is performed to calibrate the base frame relative to the world frame.

The first cycle is intended to establish a connection between the laser fixture frame and the robot base frame, while the second one correlates the base frame with world frame. At the beginning of each cycle the calibration fixture has to be set at the

specific location needed. For the first cycle this position is close to the robot base, while for the second one a location close to the future work piece in the task space is desirable. After the laser beams are roughly aligned with the PSDs manually, the rest of the calibration process can be performed fully automated without the need of user interaction. After the calibration process of either stage is finished the achieved level of accuracy is tested. If the result reflects the desired precision the cycle is finished. If that is not true, the positioning and calibration stage have to be restarted with the already acquired calibration information in order to increase the accuracy in the next run. After both cycles are finished successfully, a direct connection between the laser fixture frame (TCP proximity) and the world frame (task space) can be established in order to compensate the displacement.

4.2.2 Positioning

The first part of the algorithm is dealing with the required positioning of the laser beams on the PSDs and consists of two main parts: the manual preliminary positioning of the lasers and the automated centering of the beams on the sensor surface as presented in figure 4.6.

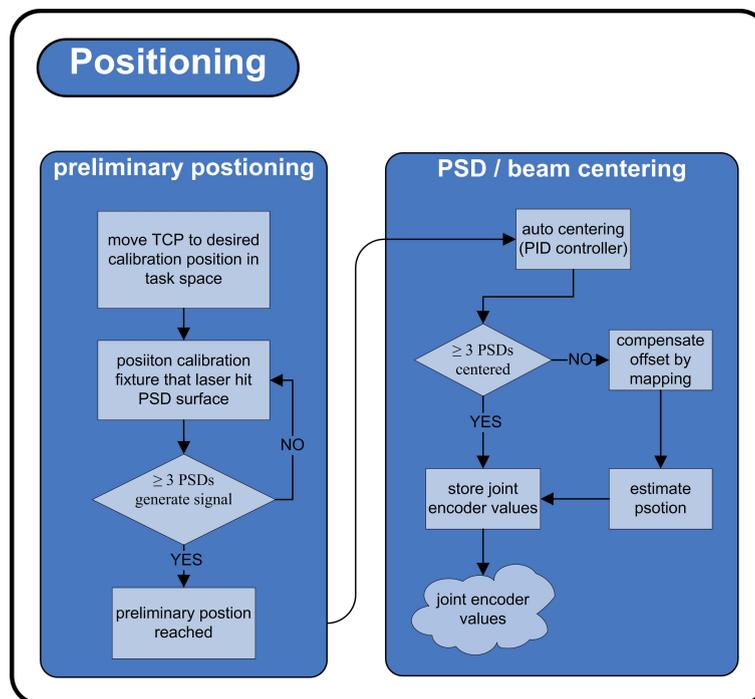


Figure 4.6: Diagram presenting the more detailed view of the positioning stage of the calibration algorithm (4.5). It consists of two functional sections: the preliminary positioning and the automated centering stage.

In order to be able to run the calibration process, a minimum of human interaction is needed in the beginning to position the laser beams on the photo-active surfaces of the PSDs. Optionally this manual interaction with the system can be eliminated when e.g. a camera system is used to guide the TCP and thus the laser beams to the PSDs. In this prototype version however this idea is abdicated to reduce the

complexity to the core problems. Once the system can detect positional information of three or more of the PSDs (i.e. lasers hit the active surface) the controller used for automatic centering of the beams over the respective TCPs can take over the control of the robot arm. After the centering of at least three beams was successful the joint encoder values are stored to be used later in the calibration process. Just in case the centering can not be completed for some reason (e.g. insufficient positional resolution of the robot arm) the earlier introduced mapping function can be used to estimate the deviation from the middle. The result achieved with the second method is a lot more error-prone than the more favorable first one. Therefore the center interpolation using the PSD mapping should be avoided if possible.

4.2.3 Calibration

In the following section the core of the calibration process is illustrated. In figure 4.7 the detailed flow chart of the second sub-algorithm dealing with the calibration itself is presented.

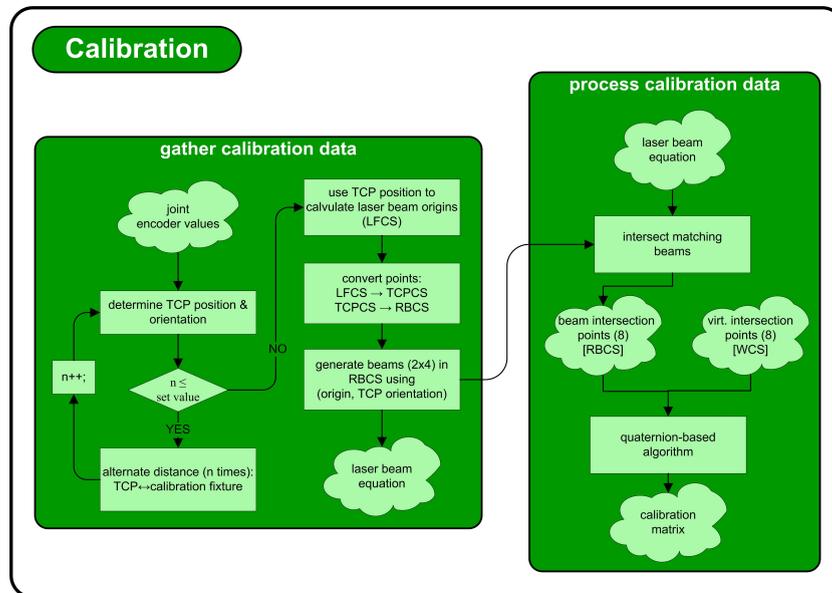


Figure 4.7: Diagram illustrating the calibration process itself. It is structured into two parts. The first part is designed to gather and process the required calibration data (i.e. laser beam equations). The second part however uses the beam equations to compute two sets of intersection points (virtual and laser beam).

After the laser beams were aligned with the PSD centers in the previous stage, the actual calibration stage of the algorithm can now be performed. In order to receive a better calibration result, the distance of TCP to calibration fixture surface (i.e. z-coordinate²) can be alternated to multiple setups. For each new distance the beam position on the PSD has to be readjusted to the center since deviations will occur. After this is done for a given number of distances the laser beam equations introduced in section 4.1.2 can be computed. For this purpose the locations of the beam origins

² Assuming the calibration fixture's top surface being parallel to the WCS x,y-plane

(given in LFCS) and the beam orientation (i.e. TCP orientation in LFCS) have to be converted into the respective coordinate system. After all beam equations are generated (two faces of the fixture with four beam equations each) the matching lines have to be intersected to receive the laser beam intersection points. Once this task is finished they are compared to the virtual intersection points that are already known (see section 4.1.1). Since these two set of points correspond with each other and are given in two different coordinate systems with unknown transformation between those systems, a quaternion based algorithm (introduced in section 4.1.3.2) is employed to establish a connection (i.e. transformation). In the first cycle this way the transformation between the LFCS and the RBCS is established while in the second one the connection between RBCS and WCS is set. Hence, one is able to receive a direct transformation from LFCS into WCS. Since the LFCS origin and orientation is known relative to the TCPCS, the needed connection between TCPCS and WCS can be derived. Therefore one is able to convert any desired TCP location in the task space into the matching robot arm input (considering the robot system inherent deviation using the found transformation). Hence, the desired position can be reached with enhanced precision in the future.

4.2.4 Result Test

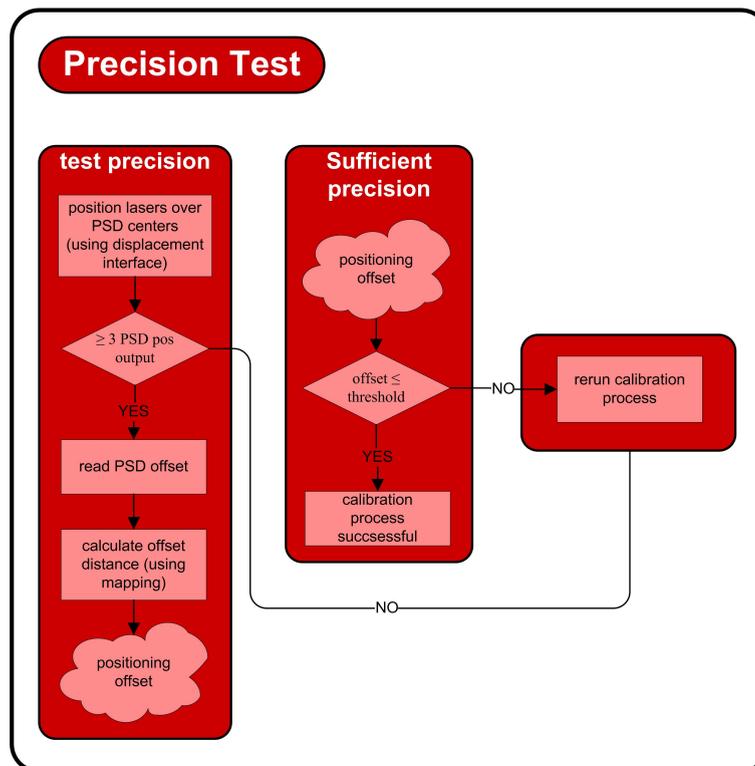


Figure 4.8: Flow chart of the last part of the calibration algorithm: the positional accuracy test. This segment of the algorithm determines if the accuracy level reached during the calibration process is sufficient or if the process has to be restarted.

To finish up the calibration process it is inevitable to find a way to determine the quality of the calibration procedure outcome. This is of special importance since

it is impossible to precisely predict the result of the process. Therefore the final stage of the calibration process presented in figure 4.8 is designed to give the needed feedback on the achieved precision.

To begin the test the laser fixture is automatically moved to the position where the centers of the PSDs (fixed on the calibration fixture) are supposed to be located. If a position is reached that satisfies the criteria specified (at least three PSDs represent beam location on surface) the positional outputs are evaluated. Hence, the deviation of the actual position from the desired position can be determined using the mapping function once more. If the new deviation is below the earlier specified threshold (i.e. desired level of accuracy) the process was successful. If the level of deviation is above this threshold or less than three PSDs show a beam location on the surface, the calibration cycle has to be restarted.

4.3 Calibration System Hardware

In following sections the focus will be directed to the hardware designed for the project. In the first section the laser fixture attached to the robot (housing the four laser pointers) will be presented. In the second section of this paragraph the calibration fixture will be introduced. The design drafts can be found in the appendix section B.

4.3.1 Laser fixture

As mentioned before the so called “laser fixture” is the device holding the laser in position. It is attached to the robot in the proximity of the TCP. This is necessary to be able to calibrate all of the robot’s joints. Being close to the TCP allows the lasers to follow all changes in pose (i.e. position and orientation) that the end effector can perform. It is mounted rigidly to the robot frame to ensure that neither position nor orientation changes during the calibration process. Otherwise the acquired results in the calibration process will be adulterated.

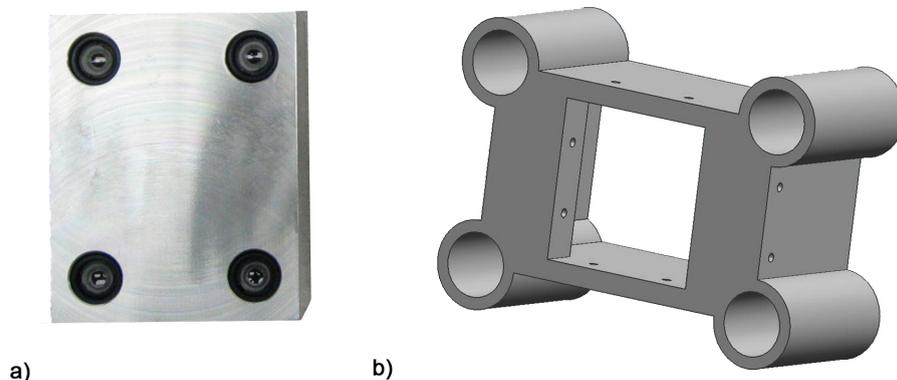


Figure 4.9: a) Photograph of the preliminary laser fixture used to attach the lasers to the robot and hold them in position. The lasers are mounted in the round cavities in each corner of the fixture. In order to reduce the weight the solid aluminum block will be cut later on. b) Draft of the final fixture generated with CAD software.

The fixture itself is designed rather simple. It basically consists of a rectangular metal frame with tube-shaped clearance on the corners to house the laser pointers. The diameter of these laser pointers is actually less than the diameter of the holes designated to hold them to be able to alter the laser pointer's optical path. This is necessary since it is absolutely crucial that all four beams are parallel to each other in order to receive good results during the calibration process. Before the laser fixture can actually be mounted on the robot a calibration of the beams has to be performed.

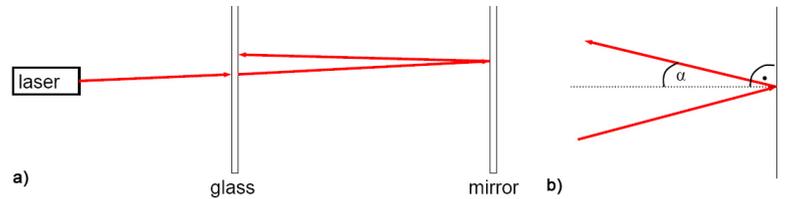


Figure 4.10: a) Illustration of the laser calibration set up. The laser beam is emitted from the laser pointer on the left. It then passes the semitransparent glass and is reflected back by the mirror. If the beam is not perfectly rectangular ($90^\circ \pm 0.006^\circ$) one will be able to detect a deviation in the point alignment on the glass. b) Magnification of the beam hitting the mirror surface.

For this process a rather simple but effective technique was chosen. The fixture is mounted to a straight plate that already holds a pane of semitransparent glass and a mirror. All three devices now have to be aligned in a reasonable distance to each other. The idea is to let the laser beam pass through the glass first. Due to the fact that the glass is not completely transparent a certain amount of attenuation will occur. This however is bearable in this case. After the glass is passed for the first time the beam will be reflected at the mirror surface and redirected back to the glass surface. If the alignment of the laser beam to the fixture front side is already rectangular only one spot will be visible on the glass screen. Otherwise the laser pointer orientation has to be altered until this condition is ensured. The bigger the distance between glass and mirror the more precise the alignment will be. The trade-off however is that the beam will actually widen up with distance. Since we consider the beam to be a single spot with negligible area the widening has to be kept as low as possible. Since the maximum application distance later will be approximately 100 cm, 50 cm will be a good calibration distance between glass and mirror since the beam travels bidirectional (i.e. optical path = 100 cm). When all the laser beams are aligned perfectly, the fixture can be mounted to the robot.

4.3.2 Calibration Fixture

The calibration fixture is the part the whole calibration process is based on. It consists of eight plastic panels holding one PSD each and a metal frame they are mounted on.

The base frame has two planes that are actually used during the calibration process: the top plane and the front plane. Those planes both hold four PSD panels each. The distance in between the sensors for each plane corresponds with the distance of

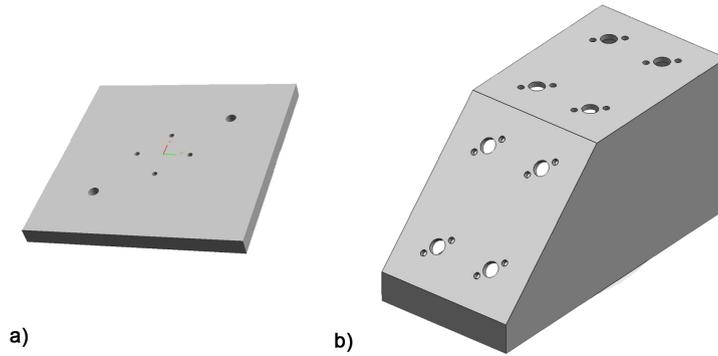


Figure 4.11: a) 3D-model of the PSD panel taken from the CAD software used for the design. The four small holes in the middle hold the PSD in place while the two bigger ones on the top right and bottom left are used to fix the panel to the frame using screws. b) Aluminum frame that holds the PSD panels. The eight sensor panels are aligned on the top and front plane. The big hole in the middle allows the PSD to move freely during the initial calibration while the small holes on either side are used to keep the pads in position using screws.

the laser beams on the laser fixture in order to be able to match the beams with the centers of the PSDs on one plane at a time. The idea behind having individual panels for each sensor is that this way one is able to move the PSDs relative to each other and thereby compensate deviations in the distances (manufacturing tolerances). The maximum movement of each panel is limited to $\pm 2mm$ on the x - and y -axis. Since manufacturing tolerances are more than one magnitude below the movement limit, the ability to achieve perfect alignment with the laser beams is ensured. Initially the PSD panel positions have to be calibrated once. Therefore the laser fixture is positioned over the respective plane with the panels. The positioning is to be performed in a serial way. The first panel is fixed in a neutral position ($x \pm 0, y \pm 0$) and the respective laser beam is to be aligned to the center of the PSD. The sensor output will indicate when the proper position is reached. Now the laser fixture is also fixed in the current position parallel to the plane. Now the second PSD is moved to the respective position until the center alignment is reached once more and then fixed in the current position. This calibration has to be performed for every single panel. Once the initial calibration is performed all the panels remain fixed in their position and don't have to be recalibrated again. The whole calibration fixture can therefore be regarded as fixed for the robot calibration process. The device shown in figure 4.11 is used for the further experiments in this project. The final version however will have more (total of 5) PSD holding side-planes. For this experimental stage the presented fixture is more than sufficient and a lot cheaper than the final version.

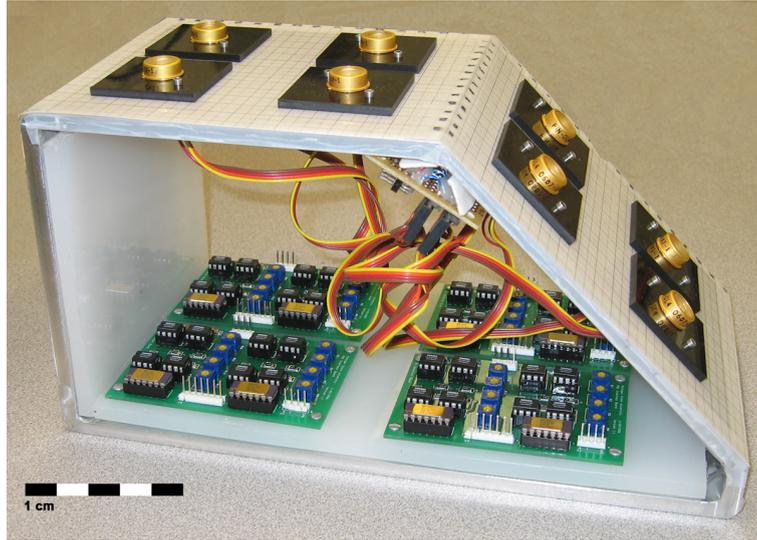


Figure 4.12: Photograph of the calibration fixture. On top of the aluminum frame the PSDs are mounted to PVC panels. Inside the frame the output processing boards are mounted and connected to the PSDs.

4.4 Error Consideration and Discussion

In this section the attention will be directed to the level of precision the calibration system is capable of ensuring (i.e. system inherent error). This information is crucial to evaluate the result and performance of the calibration process itself. In the following paragraph the error sources will be analyzed and their influence on the calibration result will be discussed.

4.4.1 Sources of Error

The calibration system error can be denoted as the 3D Cartesian offset vector $\vec{e}_i = (\Delta x_i, \Delta y_i, \Delta z_i)^T$ of the desired point (p^d) in the task space to the currently reached point of the robot's end-effector (p^c). This can be denoted as:

$$\vec{e}_i = \begin{pmatrix} p_x^d - p_x^c \\ p_y^d - p_y^c \\ p_z^d - p_z^c \end{pmatrix} \quad (4.30)$$

Using a robot with perfect accuracy, the positional reached would not deviate from the desired one. As a result of this, all components of the error vector would be equal to zero. In a real robot system however, this will never be the case since various error sources are always present. The target of the calibration process is to minimize the error vector. In order to do so, the positional error inherent in the system itself (i.e. minimal positional deviation detectable) has to be kept as low as possible. In order to reach this target it is crucial to identify the sources of error influencing the system's precision and their impact on the calibration result. After they are tracked down, efforts should be made to reduce or even eliminate the error contributions. Hence, in the following section various sources of error will be

discussed, their impact will be evaluated, and ways to avoid them will be presented if they are available. The error sources can be roughly structured in two groups: dynamic error contributions and non-dynamic error contributions. Using the calibration system one is able to specifically compensate the non-dynamic influences while only the mean dynamic error contributions (assuming a sufficient number of calibration points) can be eliminated. Since the remaining dynamic error shares can not be fully predicted (i.e. mathematically modeled with sufficient precision) only non-dynamic sources will be further discussed in the following lines.

Let us start with the most obvious error contribution: the position measurement error that occurs in the PSD position and laser pointer position. If this source could not be compensated, the resulting mean error based on the propagation of uncertainty would be greater than $250\mu m$ for each of the three vector components disregarding all other sources of error. Therefore it is crucial to eliminate this source of inaccuracy in advance. This can be achieved by initially pre-calibrating the calibration fixture to the laser fixture as described in section 4.3.1. Doing so will get rid of both the positing error of the PSDs as well as the Laser positioning errors. This will also compensate the influence of the orientation of the PSD since it might be slightly tilted relative to the fixture surface or rotated to the left or right. Hence, all deviations caused by inaccuracies during the manufacturing process can be disregarded for the further considerations since the remaining impact should be infinitesimal small.

Since the laser fixture allows it to change the laser's direction (within very low and reasonable limitations) to compensate manufacturing tolerances, the error in orientation will cause positional errors. The deviation from the earlier assumed 90° angle to the fixture surface will result in a deviation of the beams position on the PSD surface that will increase with growing distance in between the laser pointer (on the robot arm) and the PSD (on the calibration fixture). To take care of this problem the laser beam angles also have to be calibrated as describes in section 4.3.1. This ensures the direction to be perpendicular within a tolerance level of approx. $\pm 0.006^\circ$. Using basic trigonometric functions one is able to conclude the relation between positional deviation Δd , beam angle Θ , and distance between Laser fixture and PSD m as:

$$\sin(\Theta) = \frac{\Delta d}{m} \Leftrightarrow \Delta d = \sin(\Theta) \cdot m \quad (4.31)$$

Using the given angle tolerance and assuming a typical distance used during the calibration process of about $m = 250 \text{ mm}$ yields

$$\Delta d = \sin(0.006^\circ) \cdot 250 \text{ mm} \approx 0.025 \text{ mm} = 25 \mu m \quad (4.32)$$

The error can even be reduced later using addition geometric knowledge about the intersection points (e.g. equal distances on a single plane parallel to the coordinate frames of the WCS). For this purpose a least square fitting can be performed later on if necessary. This is however already done when the laser beam intersection points are transformed using the quaternion-based algorithm presented in section 4.1.3.2. Another important source of error is the shape of the laser beam used. For the calibration process it is considered to be of negligible spatial dimension. In reality

however, this is not completely true so instead of the center-point of the beam the PSD output might rather indicate the position of the left and right (respective top and bottom) edge of the beam. Furthermore the shape of the laser is not necessarily a perfect circle so that has to be taken into the consideration as well. In case of the chosen lateral-type PSD however, this will not be an issue. This is because PSD chips of this specific type indicate the center of the beam in their positional output and are not effected by deviations in the laser's energy distribution or deviations from the desired beam shape due to their unique design.

The last source of inaccuracies is the minimal detectable change in beam position by the PSD. This physical resolution limit of $\pm 0.5 \mu m$ has to be considered as positional error as well. On the other hand, the positional resolution of the robot is limited as well. Since this limit is well above the one of the PSD the influence on the outcome can be disregarded. This limit however, has to be considered when the desired level of accuracy is set before the calibration process is started. This simply means that the desired level of accuracy set at the beginning of the process has to be physically achievable for the robot (i.e. maximum allowed deviation $>$ positional resolution). Concluding one can say that the identified error sources seem to allow it to reach a reasonable level of accuracy for the vast majority of industrial applications such as welding robots etc. If needed, each of these components can be further improved during the test phase of the complete prototype system.

5. Conclusion & Outlook

5.1 Conclusion

The task of this project was to design, manufacture, and implement a robot calibration fixture based on Position Sensitive Devices. For this purpose two types of PSD sensors were evaluated and the lateral-type PSD was chosen for its significant advantages over the segmented-type. After that, a circuit was designed and successfully tested to process the PSD output and compute the beam positions on the surface. The performance of this board was proven in the mapping of the board output to a position. Therefore a mapping function based on data acquired during excessive experiments with the PSDs and the processing circuit was found and experimentally verified. Once these tasks were finished successfully a PD plus gravity controller was designed including feedback based on visual servoing. The already developed method for a camera based system presented in [Shen 02] was adjusted to fit the PSD-based approach chosen for this project. To tweak the controller performance and confirm the desired behavior, the controller including the servoing modification was simulated. The simulation fully confirmed that the controller meets all expectations. After the hardware was designed and a calibration algorithm was designed the first stage of this project is now completed. It is still too early to provide a final conclusion covering the whole calibration system since the composition of all relevant subsystems is yet to be tested, but the results acquired so far seem very promising. All key problems have been solved and the results exceed the expectations of a prototype system in a feasibility study by far in every aspect. In order to be able to use the complete prototype system, additional tweaking and research will be inevitable.

5.2 Outlook on future Work

For the next couple of months the subsystems have to be combined into a finished prototype, that is ready to use. Therefore the algorithm and the controller have to be implemented and the actual performance has to be compared to the simulated results. After this phase the system has to be tested and weaknesses in the design

have to be analyzed and improved in order to be able to use it as a “black-box” system for automated robot calibration feasible to be used in industrial application on a large scale. The field of application for the technology used in the calibration system however is not restricted to this specific task. The acquired results show that it can be applied for various purposes that required accurate positional information of e.g. sensor probes or manipulators. Therefore it could be interesting to explore applications in the field of biology (e.g. cell manipulation) or the semiconductor industry (e.g. precision cutting) as well as mechanical engineering (e.g. displacement sensor). The list could be continued almost infinitely. This proves once the enormous potential of this technology that has yet to be explored and developed.

A. Table of Abbreviations

CAP	:	C omputer A ided P robe
CFCS	:	C alibration F ixture C oordinate S ystem
D-H	:	D enavit- H artenberg conventions for kinematics published in [Denavit 55]
DOF	:	D imension O f F reedom
IC	:	I ntegrated C ircuit
LFCS	:	L aser F ixture frame C oordinate S ystem
n-layer	:	semiconductor material doped with an donor material (surplus of electrons)
PSD	:	P ositional S ensitive D evice (see figure 2.1)
p-layer	:	semiconductor material doped with an acceptor material (lack of electrons)
RBCS	:	R obot B ase frame C oordinate S ystem
TCP	:	T ool C enter P oint of the robot's end effector
TCPCS	:	T ool C enter P oint C oordinate S ystem
WCS	:	W orld C oordinate S ystem

B. Drafts

B.1 Calibration fixture

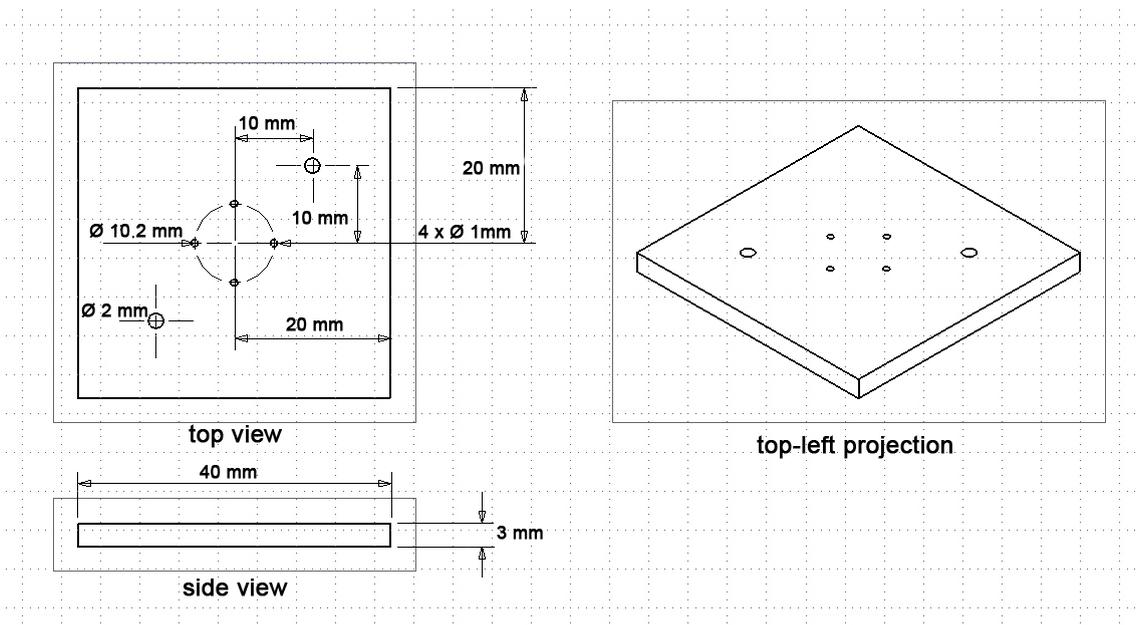


Figure B.1: Pad manufactured out of non conducting plastic that holds the PSD. In this project stage eight PSD pads are used for the calibration.

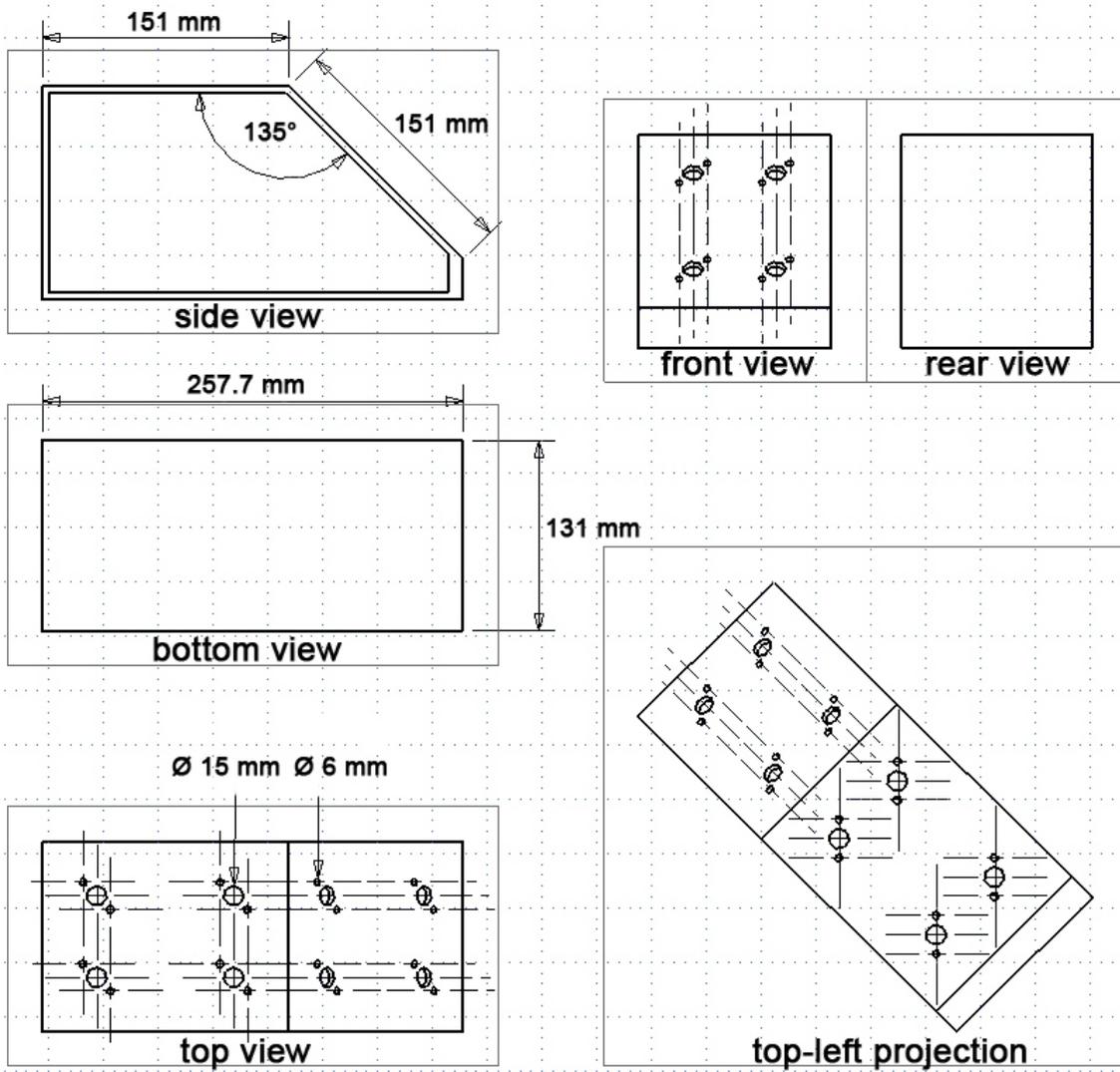


Figure B.2: Draft of the calibration fixture frame used to hold the PSD pads in position.

B.2 Laser fixture

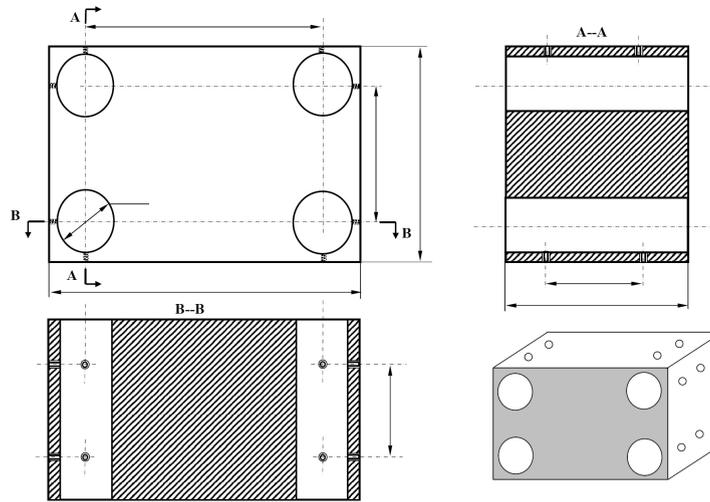


Figure B.3: Draft of the fixture designed to hold the four lasers and mount them to the robot. The laser pointers are mounted in the four cavities on either edge. Then screws are used to fix them in place and adjust deviations in orientation.

B.3 PSD output processing board

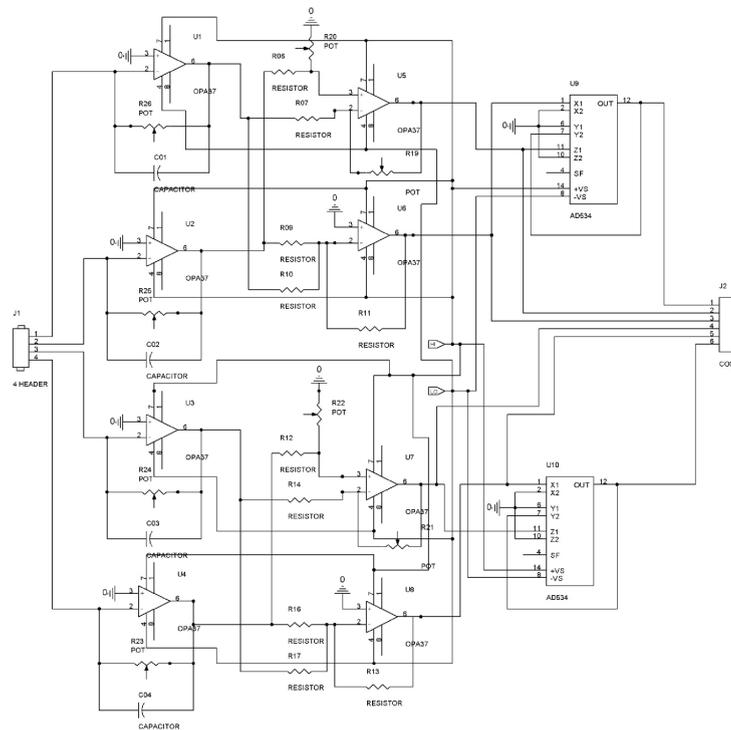


Figure B.4: Schematic drawing of the board used to process the PSD sensor output and calculate the respective x- and y-position of the beam.

C. Tables and Figures

C.1 Board Resistor Value Determination

desired value [k Ω]	1st level resistor [k Ω]	2nd level resistor [k Ω]	max Output [V]	min Output [V]
5	4,9	294,8	10,8	-11,85
5	5,02	297	11,05	-11,85
10	9,86	91,4	10,7	-11,68
10	9,84	92,6	11,1	-11,68
15	14,2	54,7	10,7	-11,5
15	14,2	54,7	10,5	-11,5
20	19,88	49,2	11,2	-11,8
20	20,35	49,6	11,3	-11,8
30	29,75	30,8	11,3	-11,7
30	28,5	31,3	11,3	-11,7
40	40,48	25,88	9,7	-11,2
40	40,78	27,8	9,6	-11,2

Figure C.1: Table representing the results of the experiment conducted to determine the proper values for the adjustable resistors of the PSD signal processing board in section 2.2.3.

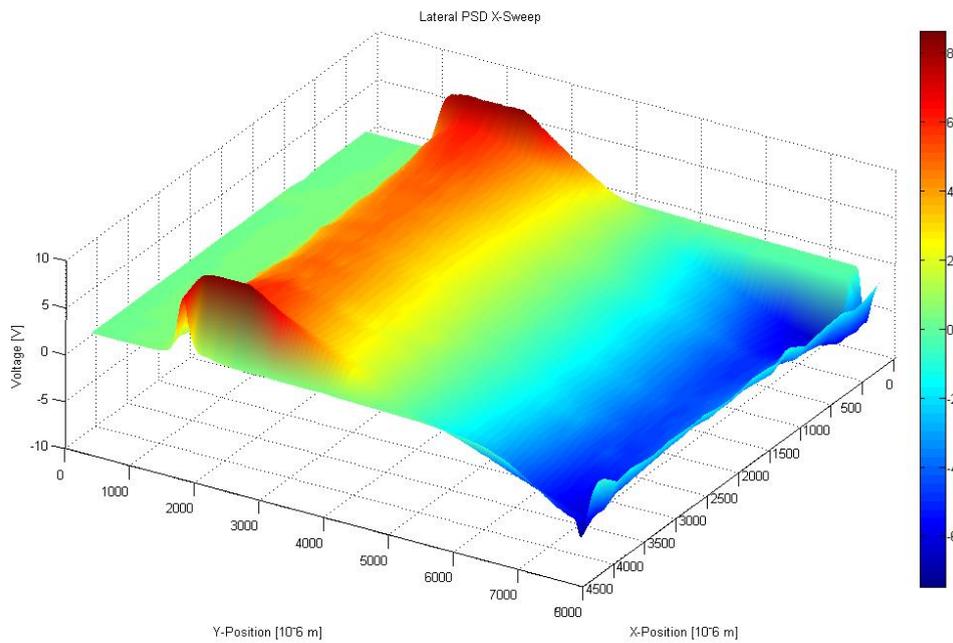


Figure C.2: Enlarged version of the 3D-Plot representing multiple sweeps in x-direction over the PSD surface at about 25 different y-locations.

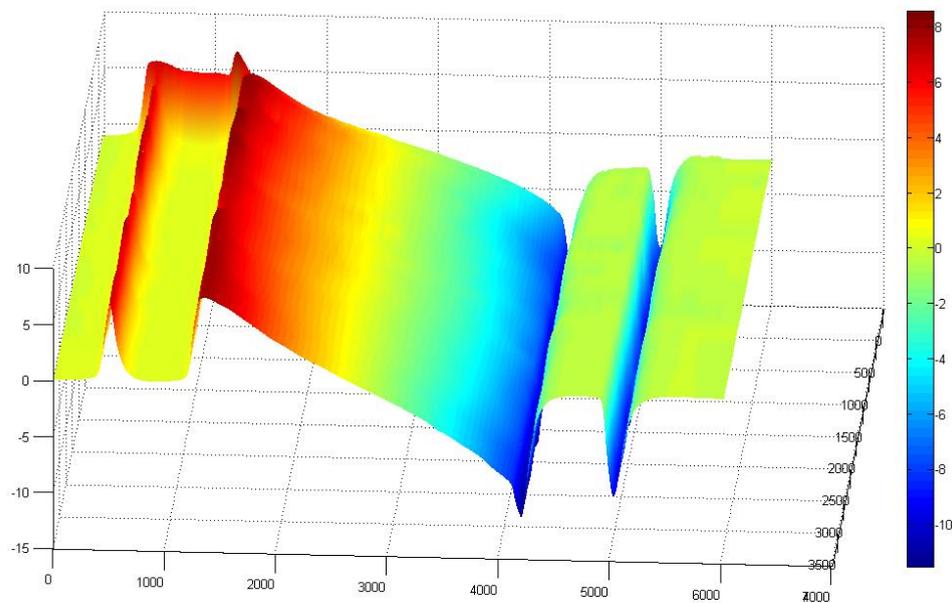


Figure C.3: Enlarged version of the 3D-Plot representing multiple sweeps in y-direction over the PSD surface at about 23 different x-locations.

References

- [Corporation 06] Hamamatsu Corporation. *Operating Principle and Features of PSD*. Hamamatsu Corporation, 360 Foothill Road P.O. Box 6910 Bridgewater, NJ 08807-0910, USA, 2006.
- [Denavit 55] J. Denavit, R.S. Hartenberg. A kinematic notation for lower-pair mechanisms based on matrices. *Journal of applied mechanics*, 22(2):215–221, 1955.
- [Elatta 04] A.Y. Elatta, Li Pei Gen, Fan Liang Zhi, Yu Daoyuan, Luo Fei. An Overview of Robot Calibration. *Information Technology Journal*, 3(1):74–78, 2004. ISSN 1682-6027.
- [Gong 00] Chunhe Gong, Jingxia Yuan, Jun Ni. A Self-Calibration Method for Robotic Measurement System. *Journal of Manufacturing Science and Engineering*, 122:174–181, 2000.
- [Hans 02] Dr. Volker Hans. *Elektrische Messtechnik I*, 2002.
- [Hutchinson 96] Seth Hutchinson, Greg Hager, Peter Cork. A Tutorial Introduction to Visual Servo Control. *IEEE Transactions on Robotics and Automation*, 12(5):651–670, 1996.
- [Lewis 93] F.L. Lewis, C.T. Abdallah, D.M. Dawson. *Control of Robot Manipulators*. Macmillan Publishing Company, New York, 1993. ISBN 0-0237-0501-9.
- [Meng 01] Yan Meng, Hanqi Zhuang. Self-Calibration of Camera-Equipped Robot Manipulators. *The International Journal of Robotics Research*, 20(11):909–921, November 2001.
- [Optronics 06] Duna Optronics. *Position Sensing and Alignment Engineering Guide*. Duna Optronics LTD., P.O. Box 3370, Nesher 20306, Isreal, 2006.
- [Photonics 03] Hamamatsu Photonics. *PSD (Position Sensitive Detector)*. Hamamatsu Photonics K.K., Solid State Division, 1126-1, Ichino-cho, Hamamatsu City, 435-8558, Japan, 2003.
- [Shen 02] Yantao Shen, Guoliang Xiang, Yun-Hui Liu, Kejie Li. Uncalibrated Visual Servoing of Planar Robots. In *IEEE International Conference on Robotics and Automation*, S. 580–585, May 2002.

- [Shiakolas 02] P.S. Shiakolas, K.L. Conrad, T.C. Yih. On the Accuracy, Repeatability, and Degree of Influence of Kinematic Parameters for Industrial Robots. *International Journal of Modelling and Simulation*, 22(3), 2002.
- [van Albada 93] G.D. van Albada, J.M.Lagerberg, Z. Zhong Wei. Portable Calibration Systems for Robots, 1993. published in "Robot Calibration", Eds. R.Bernhardt, S.Albright, Chapman & Hall, London, August 1993, ISBN 0-4124-9140-0, p.101-123.
- [Zhuang 96] Hanqi Zhuang, Zvi S. Roth. *Camera-Aided Robot Calibration*. CRC Press, 1996. ISBN 0-8493-9407-4.