Master Thesis

Signal Processing for a Homodyne Michelson Interferometer

Markus Kern

Institute of Electrical Measurement and Measurement Signal Processing
Graz University of Technology
Head of the Institute: Univ.-Prof. Dipl.-Ing. Dr. techn. Georg Brasseur

Advisor: Dipl.-Ing. Dr. techn. Markus Brandner

April 2012

Kurzfassung

Ein Weißlichtinterferometer kann verwendet werden um die Oberflächengeometrie eines Prüfkörpers mit einer Auflösung im Nanometerbereich zu rekonstruieren. Für einen Aufbau, welcher am Institut für Messtechnik und Messsignalverarbeitung entwickelt wurde, wird ein Homodyn-Michelson Interferometer eingesetzt um die zeitabhängige Verschiebung des Prüfkörpers in zwei Detektorsignale umzuwandeln. Diese Arbeit beschäftigt sich mit dem Entwurf und der Validierung einer Signalverarbeitungseinheit, welche diese Detektorsignale verarbeitet, um die Verschiebung des Prüfkörpers in Echtzeit zu schätzen.

Diese Arbeit ist in zwei Teile gegliedert: Im theoretischen Teil werden Grundlagen der interferometrischen Längenmessung erklärt und das Signalmodell des eingesetzten Aufbaues präsentiert. Des Weiteren werden verschiedene Einflussfaktoren welche das Messergebnis beeinträchtigen und die erforderlichen Korrekturmaßnahmen diskutiert. Ausgehend von einem Ursachen-Wirkungs-Diagramm wird die Messunsicherheit des Messergebnisses abgeleitet und in einem Messunsicherheitsbudget zusammengefasst. Der praktische Teil der Arbeit befasst sich mit der Entwicklung und Umsetzung einer eigenständigen Signalverarbeitungseinheit. Das entwickelte Prototypensystem führt die erforderlichen Berechnungen in Echtzeit durch und bietet eine Mensch-Maschinen-Schnittstelle sowie eine Schnittstelle zum Weißlichtinterferometer. Systemrelevante Parameter wie der Datendurchsatz und die Richtigkeit der Datenverarbeitungsalgorithmen werden experimentell bestätigt. Abschließend werden die erzielten Resultate diskutiert und Vorschläge zur Verbesserung der Leistungsfähigkeit des Messsystems präsentiert.

Abstract

A white light interferometer can be used to reconstruct the geometry of test samples with a resolution in the nanometer range. For a setup developed at the Institute of Electrical Measurement and Measurement Signal Processing an additional homodyne Michelson interferometer is used to convert the time-dependent displacement of the test sample into two detector signals. This thesis is concerned with the development and experimental validation of a signal processing unit used to process these detector signals in order to provide displacement estimates in real-time.
The thesis is organised in two parts: The theoretical part introduces the interferometer principle and presents the signal model of the used setup. Various influencing factors which affect the measurement result and the necessary correction methods are discussed. Based on the cause-effect diagram the measurement uncertainty of the displacement estimates is derived and summarized by the measurement uncertainty budget. The practical part of the thesis addresses the development and realisation of the stand-alone signal processing unit. The developed prototype system performs the required computations in real-time and provides both a human-machine interface and an interface to the white light interferometer system. Relevant parameters of the system such as the data throughput and the correctness of the signal processing algorithms are validated experimentally. The thesis closes with a discussion of the obtained experimental results and suggestions for further performance improvements of the overall measurement system.

Statutory Declaration

I declare that I have authored this thesis independently, that I have not used other than the declared sources / resources, and that I have explicitly marked all material which has been quoted either literally or by content from the used sources.

Place	Date	Signature

Contents

1 Introduction

Proceeding miniaturisation of components in various areas in industry needs increasingly demanding measurement methods with higher standards concerning their resolution, measurement range and measurement uncertainties. For non-contact and non-reactive measurements over long measurement ranges laser interferometric devices are important. Laser interferometric devices are employed in precision measurements and positioning tasks since they provide a means for attaining high metric resolution and precision. The versatility and broad applicability of laser interferometers are unattainable using any other metrological method [23]. The main advantage of optical displacement measurement is the achievable resolution of sub-nanometers even over long measurement ranges [3], [24] and [29]. Such high resolutions are attainable because the wavelength λ of coherent light provides the reference in interferometric displacement measurement systems. Nevertheless, to reach resolutions in sub-nanometer range, interpolation methods and therefore extensive signal processing is necessary [17], [28].

1.1 Motivation

For high-resolution, non-contact determination of a test sample's surface geometry, a White Light Interferometer (WLI) is developed at the Institute of Electrical Measurement and Measurement Signal Processing. Figure 1.1(a) shows a schematic assembly of a WLI. It basically consists of a white light source, a beam splitter, a reference mirror, an image sensor, and a sample to be tested. The emitted light, which is characterised by a short coherence length, is split into two beams. The measurement beam goes through the beam splitter, is reflected at the surface of the test sample and is again reflected by 90 degrees on its return path at the beam splitter in direction of the image sensor. The reference beam originates from the reflection at the beam splitter, is reflected by the reference mirror in direction of the image sensor and goes through the beam splitter. This leads to the superposition of the measurement and reference beams at the image sensor, where we can observe an interference pattern. Interference takes place only if the Optical Path

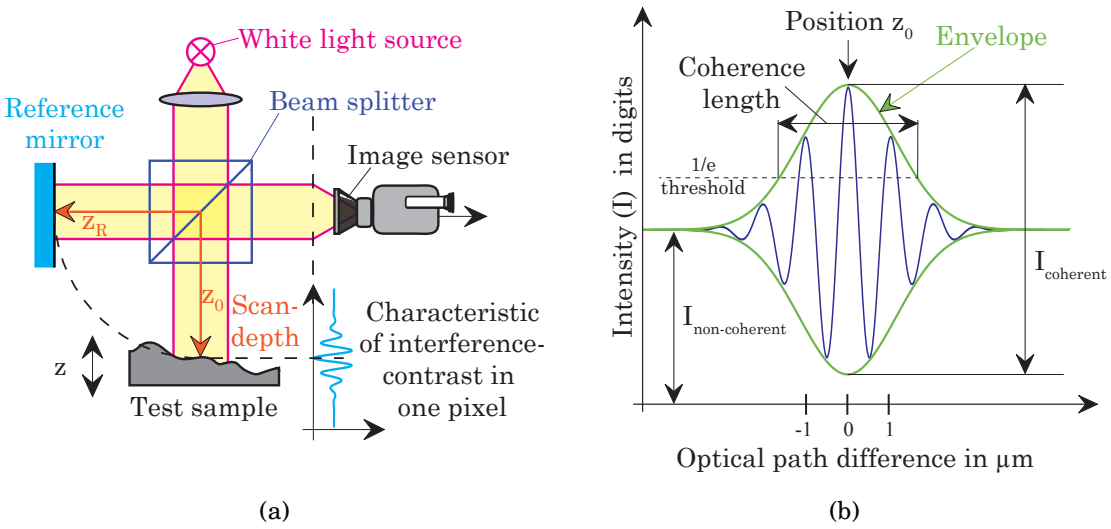


Figure 1.1: White light interferometer to determine the surface geometry of a test sample (figure from [30]). (a) schematic assembly (b) modulation of the intensity of one pixel (interferogram)

Difference (OPD) is smaller than the coherence length of the light source as illustrated in figure 1.1(b). This effect is used to determine the surface geometry by changing the scan depth z_0 during observation of pixel intensities at the image sensor. The WLI components and the developed software to read out the image sensor are described in the master thesis of Dums [11]. A complete determination of the surface geometry requires a precise indication of the scan depth z_0. Therefore Auer [6] developed a miniaturised Homodyne Michelson Interferometer (HMI). In order to make it applicable for the WLI system a real-time signal processing unit has to be developed which provides information about the displacement z.

1.2 Problem Description and Requirements

This thesis is concerned with the development and realisation of a stand-alone Signal Processing Unit (SPU) to measure displacement in sub-nanometer resolution by using signals obtained from an HMI. The necessary signal processing to estimate displacement has to meet real-time conditions to make it applicable for the WLI. Figure 1.2 shows the concept of the SPU with the required interfaces. To integrate the SPU to the WLI system it has to provide a communication interface to a Personal Computer (PC) which can be used to exchange data and control system parameters. Additionally a human machine

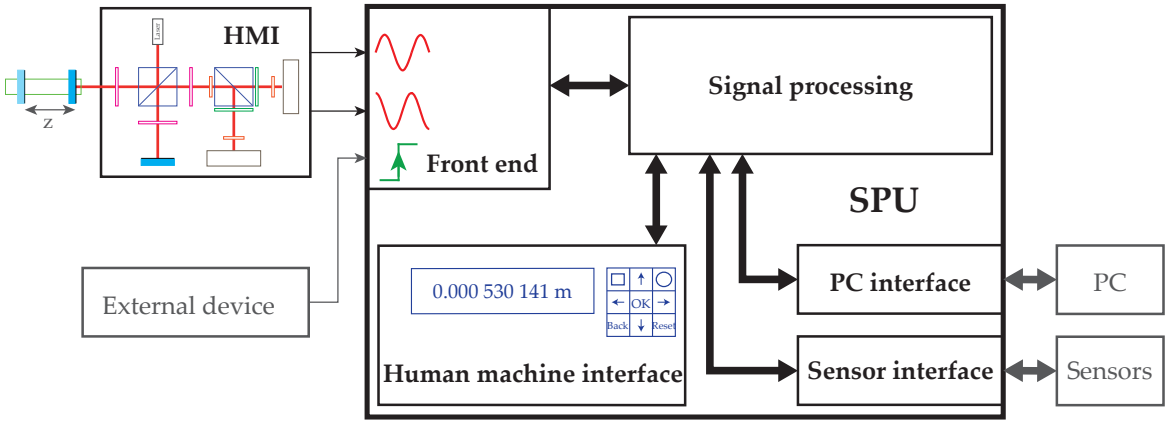


Figure 1.2: Concept of the SPU with the required interfaces which has to be developed. The HMI provide the necessary signals to determine the displacement z. The human machine interface displays measurement results and allows the user to control the device. Additionally, sensors measuring environmental parameters and a PC, to exchange data, can be attached. The SPU also provides a trigger input to be able to synchronise measurements with external devices.

interface has to be implemented to alternatively use the SPU stand-alone without any other peripheral devices. To visualise the measurement results and to inform the user about important parameters a graphical output device has to be integrated into the SPU.
The determination of a test sample's surface geometry with the WLI requires a spatial resolution of 1 nm. The required data processing performance of the SPU depends on the maximum velocity of the test sample in the WLI. Currently the WLI software developed by Dums, moves the test sample with a speed of $0.002\,\frac{\mathrm{mm}}{\mathrm{s}}$ to determine the surface geometry. However the limit of the test sample's velocity in the WLI is actually specified with $100\,\frac{\mathrm{mm}}{\mathrm{s}}$. The maximum displacement range z in the WLI is given by 20 mm.
To determine the surface geometry of a test sample, the WLI moves the test sample along a trajectory and wants to know the position at specific points in time. For this reason the WLI generates trigger impulses. The SPU has to provide a trigger input channel to synchronise the measurement results with the smallest possible time base error and memorise these measurement results for further processing. The WLI system reads the stored measurement results from the SPU after a certain number of trigger events. For practical reasons the SPU has to be able to set a reference point at an arbitrary position to which it refers all measurement results to.
An HMI uses coherent radiation and its interference properties to relate displacements of a mirror to intensity changes on photo detectors. The intensities on the photo detectors and therefore the photo currents depend mainly on the source of radiation in the HMI, applied apertures in the optical path, and ambient light. It has to be expected, that the

amplitudes and offsets of the photo currents may vary by large. For that reason, a smart front end has to be developed which can handle and compensate different input currents without human intervention.
Systematic errors in interferometric displacement measurement arise from unbalance of the photo currents concerning their amplitudes and offsets, and from phase delays between the photo currents unequal to 90° [58]. This property of a detector system is called the lack of quadrature and needs to be compensated in order to minimise systematic measurement errors.
The reference in interferometric displacement measurement is the wavelength λ of the coherent radiation source in ambient air. The wavelength λ is a function of the vacuum wavelength λ_0 of the radiation source and the refractive index of air. The latter depends on temperature, pressure, water vapour, and carbon dioxide concentration. Because the knowledge of the wavelength λ is essential for the SPU to determine displacements, we have to estimate the prevalent refractive index of air.
Measurement results are incomplete unless accompanied by a statement of the associated uncertainties. For this reason the measurement uncertainty, with focus on the signal processing part of the measurement system, needs to be evaluated incorporating signal preconditioning and data processing.

Summary of requirements for the SPU to be developed:

- The SPU has to offer a human machine interface with a graphical output device, to be able to control and show parameters and measurement results without other peripheral devices.
- The SPU has to provide an interface to communicate with a PC to exchange data.
- The SPU has to automatically adapt amplification and offset of the input signals.
- The processing performance of the SPU has to be able to handle a maximum velocity of $100\,\frac{\mathrm{mm}}{\mathrm{s}}$ in real-time, with a spatial measurement resolution of 1 nm.
- The SPU has to offer a trigger input to synchronise measurements with external devices and keep triggered measurement results in memory.
- The SPU has to be able to estimate the prevalent refractive index of air with knowledge of ambient temperature, barometric pressure and humidity in order to calculate the wavelength λ.
- The SPU has to offer an interface for sensors which measure the environmental parameters.

2 Related work

This chapter describes the principle of interferometric displacement measurement with an HMI in order to better understand the relationship between displacement of the measurement mirror and the signals obtained from an interferometer. Further the miniaturised, symmetrical, one-beam HMI developed and realised by Auer [6] is explained briefly to illustrate the known issues in processing quadrature signals. Finally current approaches for measurement uncertainty evaluation are presented.

2.1 Displacement Measurement with a Homodyne Michelson Interferometer

An Homodyne Michelson Interferometer is an assembly of optical components which can be used to determine the quantity displacement with high resolution. In an HMI linearly polarised radiation is spilt into two beams that interfere when being superimposed after propagation along different optical paths. Suitably positioned polarisation filters and retarder plates in the optical path generate two detector signals with a phase delay of ideally 90°. By continuous evaluation of these detector signals the direction and the magnitude of the displacement can be determined.

Figure 2.1 illustrates a schematic assembly of an HMI. As source of radiation, a single frequency laser with 45° linear polarisation and a wavelength of λ, is used. The laser beam is split into a reference and measurement beam with the Non Polarising Beam Splitter (NPBS). Afterwards the reference and measurement beam are propagated along different optical paths, reflected at the reference and measurement mirror, respectively, and superimposed. At the NPBS the beam is again split into two partial beams, where one is directed towards the beam trap and the other is reflected towards the Polarising Beam Splitter (PBS). Thereby the horizontally and the vertically polarised component are directed towards the photo detectors 1 and 2, respectively. If we denote the phase delay of the signal at detector 1 by Φ_0 and omit the $\frac{\lambda}{4}$-retarder plate, the signal at detector 2 would have a phase delay of $\Phi_0 + \pi$.

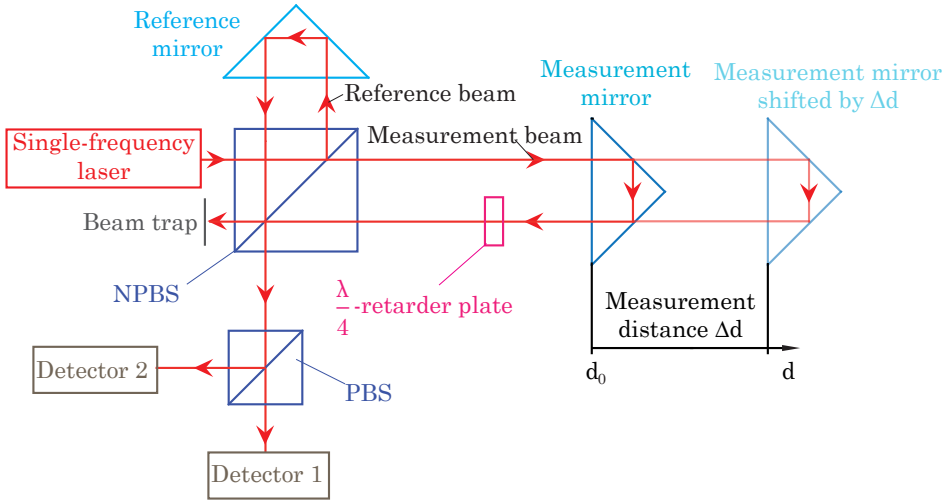


Figure 2.1: Schematic assembly of a homodyne Michelson interferometer (figure from Auer [6]).

The $\frac{\lambda}{4}$-retarder plate in the measurement arm transforms linearly polarised to circularly polarised optical radiation. Consequently, a phase delay of the intensity signal at detector 2 of $\Phi_0 + \pi + \frac{\pi}{2}$ is obtained. The intensities at the detectors are proportional to the phase delay Φ_0. The detectors convert these intensity signals to electrical photo currents which are described by equations 2.1.

$$\begin{aligned} I_1 &\sim 1 + \cos(\Phi_0) \\ I_2 &\sim 1 + \sin(\Phi_0) \end{aligned} \tag{2.1}$$

If the measurement mirror is shifted by the distance Δd, an additional phase delay on detector 1 and detector 2 is observed. Equations 2.2 describe the resulting photo currents.

$$\begin{aligned} I_1 &\sim 1 + \cos\left(\Phi_0 + \frac{4\pi \cdot \Delta d}{\lambda}\right) \\ I_2 &\sim 1 + \sin\left(\Phi_0 + \frac{4\pi \cdot \Delta d}{\lambda}\right) \end{aligned} \tag{2.2}$$

If we plot the photo current signal I_1 on the abscissa and the photo current signal I_2 on the ordinate and shift the measurement mirror we obtain a Lissajous figure in the shape of a circle as illustrated in figure 2.2. This circle is described by a phasor of constant length and a phase angle Θ. One revolution of the phasor corresponds to a measurement mirror

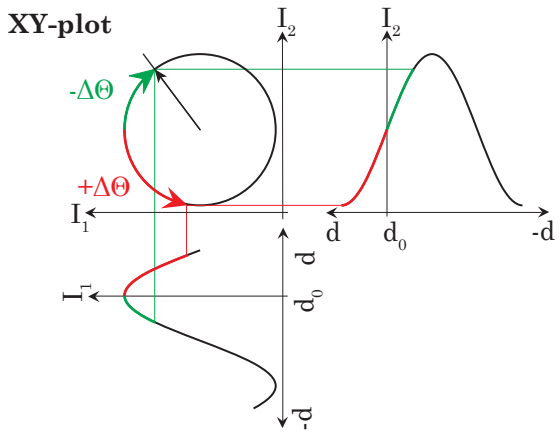


Figure 2.2: Ideal characteristic of the photo currents. The orthogonal projection of the photo currents shows a circle when shifting the measurement mirror (figure from Auer [6]).

displacement of $\frac{\lambda}{2}$. By continuous detection of the intensity signals on detector 1 and detector 2, we can determine the resulting direction and magnitude of the measurement mirror displacement according to equation 2.3.

$$\Delta d = \frac{\Delta\Theta \cdot \lambda}{4\pi} \tag{2.3}$$

Adequate signal processing makes spatial resolutions of the displacement Δd in the sub-nanometer range possible. A crucial parameter in determining the displacement is the refractive index n of the propagation medium, which influences the wavelength λ of optical radiation by $\lambda = \frac{\lambda_0}{n}$, whereby λ_0 denotes the vacuum wavelength. The refractive index of air changes with the temperature, pressure, water vapour, and carbon dioxide concentration. Based on the work of Edlén [12] who derives the refractive index of air, Birch and Downs [7, 8] present a correction of the formulae, which is used in a modified form by the National Institute of Standard and Technology (NIST) [38]. The NIST slightly modifies the water vapour term to improve the results if the temperature differs significantly from 19.6°. We use the NIST approach to calculate the refractive index of air.

The previous outline describes the basic concepts of an HMI including the relationship between intensity changes on the detectors and the displacement Δd of the measurement mirror. However, Auer [6] realised a miniaturised, symmetrical, one-beam HMI in form of a prototype which is depicted in figure 2.3.
Auer uses a stabilised Linearly Polarised (LP)-single-frequency laser SL03 [33] from Sios as source of coherent optical radiation. A polarisation-maintaining fibre optics

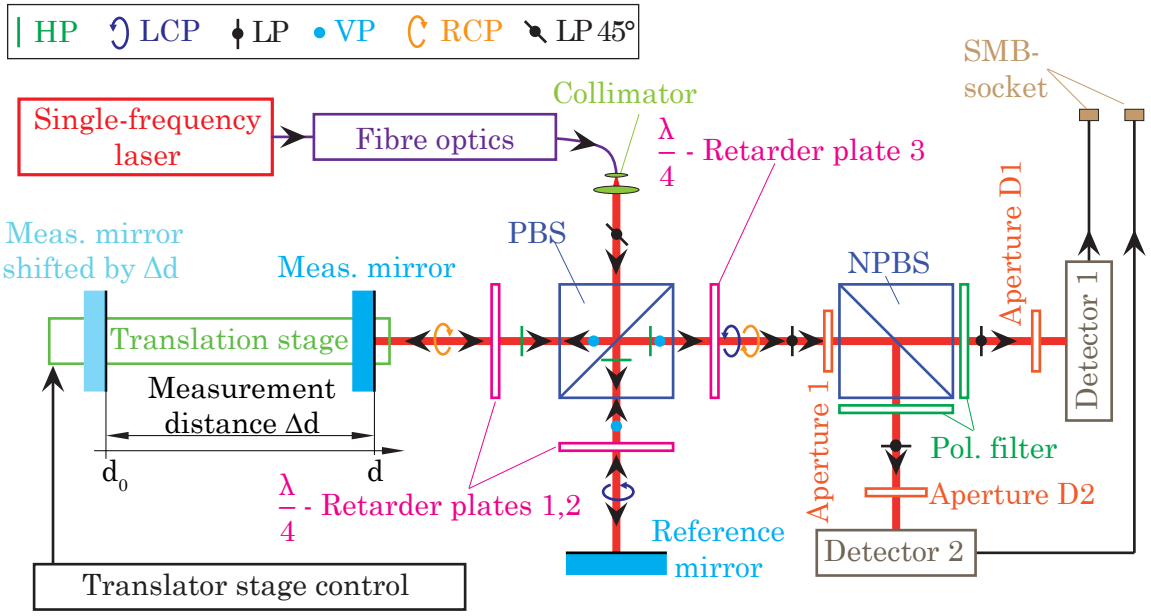


Figure 2.3: Theory of operation of the developed miniaturised HMI by Auer [6]. The polarisation angle of the linearly polarised beam, resulting from the $\frac{\lambda}{4}$-retarder plate 3, depends on the optical path difference between the measurement and the reference beam and contains the information about the measurement mirror position.

is used to couple the laser beam into the collimator. The collimator ensures parallel alignment of the optical radiation in direction of the PBS. At the PBS the 45°-LP optical radiation is split in a Horizontally Polarised (HP) measurement beam and a Vertically Polarised (VP) reference beam. Due to the reflection of the beams at the measurement and reference mirrors they pass the $\frac{\lambda}{4}$-retarder plates twice. Therefore the polarisation direction of the beams are rotated by 90° before they propagate towards the $\frac{\lambda}{4}$-retarder plate 3. The PBS guarantees that no optical radiation is reflected towards the collimator and enters the source. The $\frac{\lambda}{4}$-retarder plate 3 converts the HP and VP partial beams into a Left Circularly Polarised (LCP) and a Right Circularly Polarised (RCP) beam. The superposition of the LCP and the RCP beam leads to an LP beam. The polarisation direction of the resulting LP beam depends on the OPD between the measurement and the reference beam. When the measurement mirror is moved along the measurement distance Δd with the translator stage, the polarisation angle of the LP optical radiation changes.

As depicted in figure 2.2, two delayed signals are necessary to determine the direction of the measurement mirror movement. These signals are obtained by two analysers attached to the NPBS, which are shifted 45° to each other. Additionally, apertures facilitate a punctual observation of the interference pattern by the detectors. Two photo diodes of the

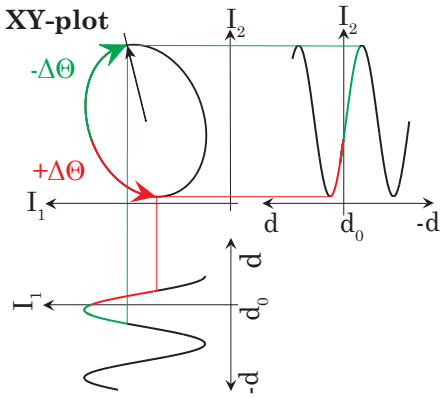


Figure 2.4: Real characteristic of the photo detector currents. The orthogonal projection of the photo currents shows a centre shifted, rotated ellipse when shifting the measurement mirror. This leads to systematic errors in the measurement result.

type BPX65 [20] are used to convert the intensity signals into electrical currents.
Due to different alignment and part variations concerning the photo diodes, non-identical apertures in the HMI, misalignment of the analysers, and other optical influences, the observed Lissajous figure is not a circle any more. The photo currents show different amplitudes and offsets and a phase delay unequal to 90°. Figure 2.4 illustrates the result of the orthogonal projected photo currents. The characteristic of the photo currents are described by the equations 2.4.

$$\begin{aligned} I_1(t) &= I_{1a} \cdot \left(1 + \cos\left(\frac{4\pi \cdot d(t)}{\lambda} + \Phi_1\right)\right) + I_{1o} \\ I_2(t) &= I_{2a} \cdot \left(1 + \sin\left(\frac{4\pi \cdot d(t)}{\lambda} + \Phi_2\right)\right) + I_{2o} \end{aligned} \tag{2.4}$$

This leads to systematic non-linearities in calculating the displacement from the phase angle Θ as given in equation 2.3. These commonly encountered systematic non-linearities in quadrature detector systems can be corrected by an elliptical fitting technique based on a least-squares approach [17]. Approaches to correct this systematic non-linearities are published by Heydemann [19] and Wu *et al.* [59].

2.2 Measurement Uncertainty

The outcome of a measurement, which is an estimate of the true value, is incomplete if not accompanied by a statement of the uncertainty of that estimate.
The Guide to the Expression of Uncertainty in Measurement (GUM) [21], developed by

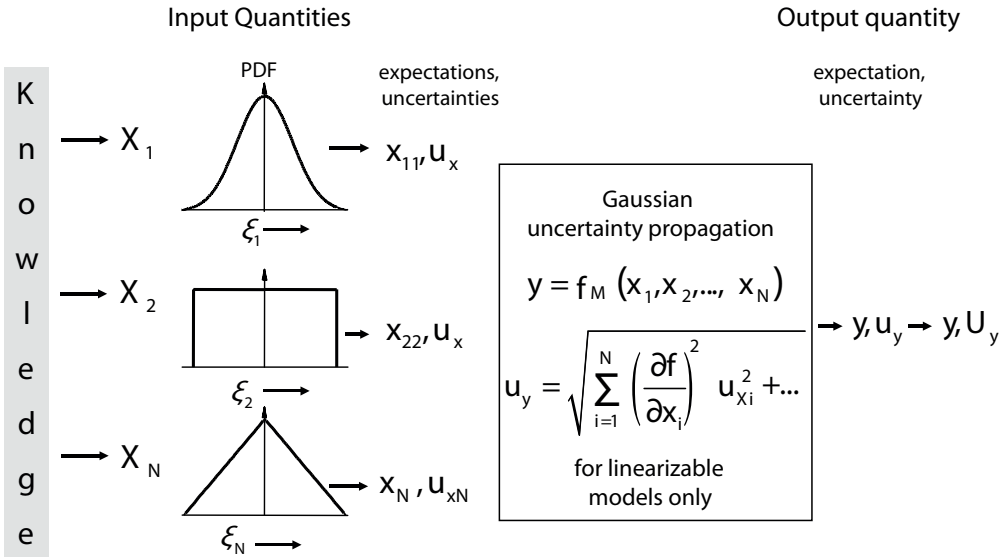


Figure 2.5: Illustrates the concept of GUM procedure (figure from [36]). Symbols: $X_1,...,X_N$ - input quantities; $\xi_1,...,\xi_N$ - realisations of the input quantities; $x_1,...,x_N$ - expected values of the input quantities with their associated standard uncertainty $u_{x1},...,u_{xN}$; Y - output quantity with its expected value $y = E[Y]$ and its associated combined standard uncertainty u_y; U_y - expanded measurement uncertainty

the Joint Commitee for Guides in Metrology (JCGM), is an approved and internationally accepted guideline to express measurement uncertainties. The GUM has first been released in 1993 and corrected and republished in 1995 and in 2008.

Figure 2.5, published by Sommer and Siebert [36], illustrates the concept of GUM procedure. It depicts how we obtain the expanded measurement uncertainty U_y from the input quantities $X_1,...,X_N$ regarding the model equation f_M. Each input quantity X_i is described by its expected value x_i together with its associated standard uncertainty u_{xi}. GUM uses two types of standard uncertainties based on how the information is found. A standard uncertainty is of type A, if it is estimated based on an observed frequency distribution. A type B standard uncertainty is obtained from an assumed Probability Density Function (PDF) and not by observation of repeated realisations. As an example for type B standard uncertainty, someone can use a value and its associated limit of the manufacturer specifications, assume the underlying PDF and estimate the standard uncertainty.

In practice, modelling of the measurement process and the derivation of the model equation, which relate the input quantities to the output quantity, can be difficult. Sommer

and Siebert [35] state a systematic approach how to derive the measurement equation. A detailed analysis of the measurement process is necessary to develop a cause-effect relation of a fictitious ideal measurement. Subsequent all imperfections and influences are introduced into the cause-effect relationship. Based on this knowledge the model equation $y = f_M(x_1, x_2, ..., x_N)$ is derived by inverting the cause-effect relationship.
In many cases the model equation is linear or can be approximated by a first order Taylor-series expansion about the best estimates x_i [36]. In doing so the combined uncertainty u_y of the output quantity Y is determined by equation 2.5 where x_i and x_j are estimates of X_i and X_j with their associated uncertainties u_{xi} and u_{xj}. The estimated covariance of the input quantities X_i and X_j is denoted by $u_{xi,xj} = u_{xi} \cdot u_{xj} \cdot r_{ij}$ where r_{ij} is the corresponding correlation coefficient. The partial derivatives $\frac{\partial f}{\partial x_i}$ are called sensitivity coefficients. This equation is also known as the law of Gaussian uncertainty propagation.

$$u_y = \sqrt{\sum_{i=1}^{N} \left(\frac{\partial f}{\partial x_i}\right)^2 u_{xi}^2 + 2 \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} \frac{\partial f}{\partial x_i} \frac{\partial f}{\partial x_j} u_{xi,xj}} \tag{2.5}$$

This approach allows us to directly calculate the uncertainty contribution of each input quantity and state the uncertainty budget, which can be very useful to identify the dominant sources of uncertainty in our measurement process.
By multiplying the combined uncertainty with a coverage factor we obtain the expanded uncertainty $U_y = k_p \cdot u_y$, which is often preferred in practice. The coverage factor corresponds to a particular level of confidence, where it is confidently believed that Y lies between the interval $y \pm U_y$. Actually the coverage factor can only be determined if we know the underlying PDF of Y. However, in many cases the PDF is assumed to be Gaussian and a coverage factor of $k_p = 2$ is used, which corresponds to a confidence level of approximately 95 %.

The GUM uncertainty framework is limited to models which are either linear or can be linearised with sufficient small errors at a specific operating point. Even if a given model fulfils this requirement, it can be difficult to determine sensitivity coefficients by taking the derivatives of the model equation. The GUM procedure also does not provide any information about the PDF of the output quantity Y [25]. This motivated the JCGM to develop the Guide to the Expression of Uncertainty in Measurement - Supplement 1 (GUM-S1) [22], which provides a numerical approach to evaluate measurement uncertainties. This approach is based on Monte-Carlo simulation where the input quantities $X_1, ..., X_N$, with their assigned state-of-knowledge PDFs, are sampled repeatedly and propagated through the model to obtain the PDF of the output quantity Y. This concept

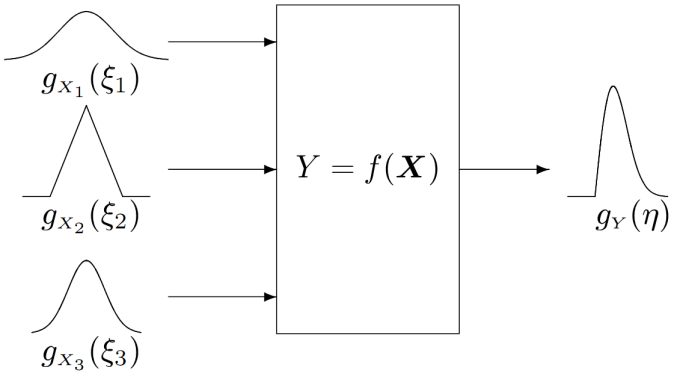


Figure 2.6: Illustrates the propagation of distributions for $N = 3$ independent input quantities (figure from [22]). $g_{X_i}(\xi_i)$ denotes the PDF with variable ξ_i for the input quantity X_i. $g_Y(\eta)$ denotes the PDF with variable η for the output quantity Y and is indicated to be asymmetric as generally arises for non-linear models $Y = f(\mathbf{X})$ or asymmetric $g_{X_i}(\xi_i)$.

is depicted in figure 2.6. The resulting PDF of the output quantity Y is used to calculate the combined measurement uncertainty u_y and the expanded measurement uncertainty U_y. Monte-Carlo simulation is a powerful tool to evaluate measurement uncertainties but it does not directly provide information about the uncertainty contribution of each input quantity X_i.

3 Implementation on the TMS320C6000 DSP Platform

This chapter describes the realisation of the SPU. First of all the measurement system and signal processing steps required to obtain the displacement of the HMI's measurement mirror with reasonable accuracy are explained. This is done without consideration of platform specific implementation aspects. Afterwards the realised and assembled hardware is described and important specifications, which are necessary for the measurement system, are included. Finally some implementation aspects with a focus on continuous sampling and filtering on the TMS320C6000 Digital Signal Processor (DSP) platform are discussed.

3.1 Measurement System Design

Figure 3.1 gives an overview of the system design with all related interfaces. The HMI provides two current signals. The signal characteristics are dependent on the velocity of the measurement mirror. In case of constant velocity, these current signals are sinusoidal with a constant frequency. Before these signals can be used for digital signal processing we have to pre-condition them and to convert them into the digital domain. The pre-conditioning includes the conversion to voltage signals, amplification of the magnitudes, level shifting, and finally filtering to avoid aliasing in the digital domain. The front end is responsible for these tasks before the signals are converted with an Analog-Digital-Converter (ADC). Additionally to the two interferometer channels the front end also offers a trigger input to synchronise the measurements with external devices.
To realise a measurement system, as outlined in section 1.2, several steps concerning the calibration of the system need to be done, before we can measure displacement in sub-nanometer range. On the one hand side we have to overcome the problem that the input currents of the front end may differ by large concerning their amplitudes and offsets. On the other hand side we have to compensate the lack of quadrature with an appropriate

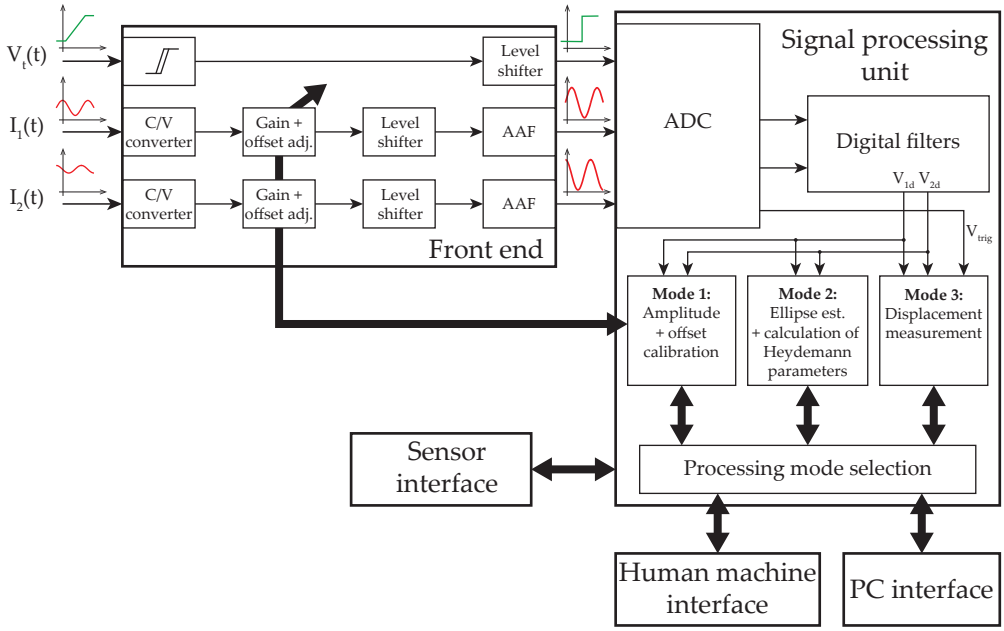


Figure 3.1: Overview of the system design with all related interfaces. The SPU uses the data from the ADC and process them according to the selected processing mode. The first processing mode adapts the front end such that the input voltages cover the full scale range of the ADC. The second processing mode uses the interferometer channel signals to estimate the ellipse parameters and calculate the parameters for the Heydemann correction. The third processing mode performs the measurement of the displacement in real-time.

correction method. Therefore we introduce three different signal processing modes as illustrated in figure 3.1:

Mode 1: Amplitude and offset calibration
The adaptation for different amplitudes and offsets is solved before the signals from the HMI are converted into the digital domain to utilise the full scale and resolution of the ADC. Hence the front end is equipped with a circuit to adjust the amplitude and the offset of the signals coming from the HMI. Because we do not want to calibrate the amplitude and offset manually we use an adequate algorithm to calibrate the channels automatically.

Mode 2: Ellipse estimation and calculation of Heydemann parameters
After adaptation of the front end we use the Heydemann correction [19] to compensate the lack of quadrature of the HMI signals. For the Heydemann correction the knowledge of the ellipse parameters is required to calculate the correction parameters. Once the Hey-

demann parameters have been determined, the calibration of the measurement system is finished. The measurement system automatically switches to the third processing mode.

Mode 3: Displacement measurement
This processing mode is used to perform displacement measurement in real-time. In this processing mode previously determined Heydemann parameters are necessary to correct the lack of quadrature and estimate the phase angle Θ between the interferometer channels. Further, we have to estimate the prevalent refractive index n and calculate the resulting wavelength λ to determine the displacement d according to equation 2.3. As long as we do not have the parameters for the adjustment at the front end and the correction parameters for compensating the lack of quadrature, the processing mode to measure displacement can not be entered.

3.1.1 Amplitude and Offset Calibration

To automatically calibrate the amplitude and offset of the interferometer channels we use a control loop to adapt the signal pre-conditioning at the front end. This approach is illustrated in figure 3.2. The SPU measures the amplitude and the offset of both interferometer channels and calculates the deviation between given reference values and measured values. The gain and the offset of the interferometer channels are adjusted at the front end if the deviations exceed accepted tolerance values. This procedure is repeated until the deviations are smaller than the given tolerance values. In this case the calibration of the amplitude and offset is finished, the control loop is deactivated, and the setting at the front end are kept constant.

3.1.2 Heydemann Correction and Ellipse Fitting

An idealised quadrature detector system is characterised by equal detector amplitudes and offsets and a phase shift between the signals of 90°. The signals can be described by equations 3.1. The orthogonal projection of V_1 and V_2 describes a vector from the origin with a constant length R rotating clockwise or counter clockwise depending on the phase angle variation $\Delta\Theta$. The end point of the vector (V_1, V_2) lies on a circle as illustrated in figure 3.3.

$$\begin{aligned} V_1 &= R \cdot \cos\Theta \\ V_2 &= R \cdot \sin\Theta \end{aligned} \tag{3.1}$$

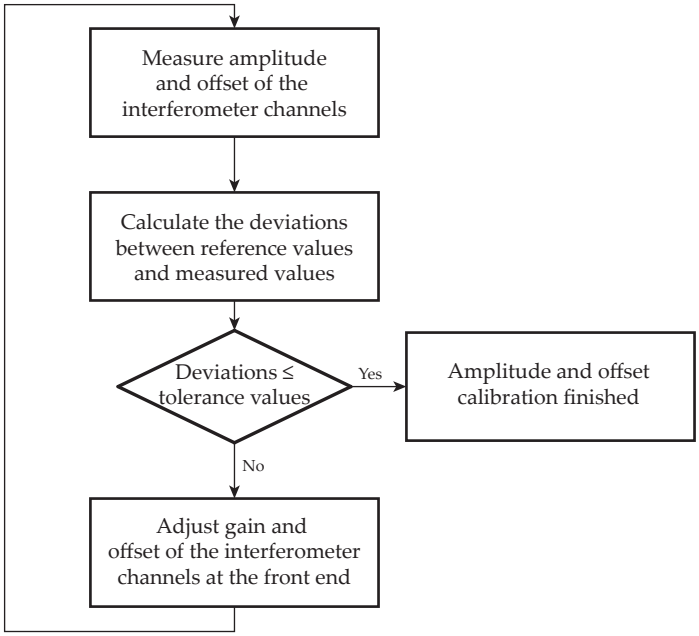


Figure 3.2: Control loop to adapt the signal pre-conditioning at the front end. The control loop is deactivated if the deviations between reference values and measured values are smaller than the given tolerance values.

As already explained in section 2.1, with real quadrature detectors the end point of the vector is rarely on a circle but on a distorted ellipse illustrated by the dashed line in figure 3.3. Heydemann [19] expresses the distortion from a circle (V_1, V_2) into an ellipse (V_{1d}, V_{2d}) by equations 3.2, where G is the channel gain ratio, p and q are the offsets in the cosine and sine channels, and a is the quadrature error.

$$\begin{aligned} V_{1d} &= V_1 + p \\ V_{2d} &= \frac{1}{G}\left(V_2 \cdot \cos(a) - V_1 \cdot \sin(a)\right) + q \end{aligned} \tag{3.2}$$

The distorted circle can now be described by equation:

$$(V_1 + p)^2 + \left(\frac{V_2 \cdot \cos(a) - V_1 \cdot \sin(a)}{G} + q\right)^2 = R^2 \tag{3.3}$$

Equation 3.3 is reformulated in form of a general ellipse equation:

$$A \cdot V_{1d}^2 + B \cdot V_{2d}^2 + C \cdot V_{1d} V_{2d} + D \cdot V_{1d} + E \cdot V_{2d} + F = 0 \tag{3.4}$$

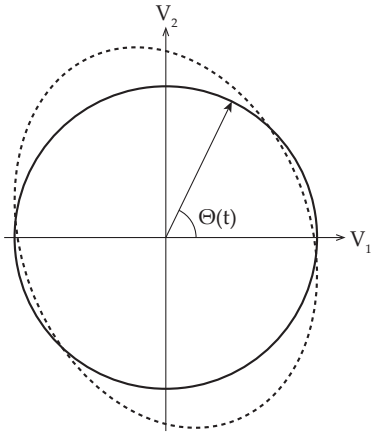


Figure 3.3: The solid line shows an idealised/corrected quadrature system and the dashed line shows a real quadrature detector system.

with the coefficients:

$$\begin{aligned} A &= \left(R^2\cos^2(a) - p^2 - G^2q^2 - 2Gpq\cdot\sin(a)\right)^{-1} \\ B &= AG^2 \\ C &= 2AG\cdot\sin(a) \\ D &= -2A\left(p + Gq\cdot\sin(a)\right) \\ E &= -2AG\left(Gq + p\cdot\sin(a)\right) \\ F &= -1 \end{aligned} \tag{3.5}$$

The coefficients A through E are obtained from a least square fit of the ellipse resulting from time domain signals V_{1d} and V_{2d}, acquired during linear motion of the measurement mirror. Fitting of an ellipse can be achieved by many different approaches. We decided to use a Recursive Least Square (RLS) algorithm, which is characterised by a fast rate of convergence that is relatively insensitive to the eigenvalue spread of the correlation matrix of the input data. Haykin [32] presents, beside the derivation of the RLS algorithm, a simple implementation of the algorithm and discusses some related finite precision aspects. As depicted in figure 3.4(a), the RLS algorithm uses the input data vector $\mathbf{u}(n)$ to adapt transversal filter taps $\hat{\mathbf{w}}(n)$ such that the resulting error $\xi(n)$ is minimized in the least squares sense. Figure 3.4(b) depicts the signal-flow graph of the RLS algorithm and table 3.1 summarises the algorithm. In the initialisation of the RLS algorithm we set the filter coefficients $\hat{\mathbf{w}}(0)$ of the transversal filter to $\mathbf{0}$ and the inverse correlation matrix $\mathbf{P}(0)$ to $\delta^{-1}\mathbf{I}$, where δ denotes the regularization parameter and $\mathbf{I}$ the identity matrix. The regularisation parameter should be assigned a small positive value for a high Signal-to-Noise Ratio (SNR) and a large positive value for a low SNR. The gain vector

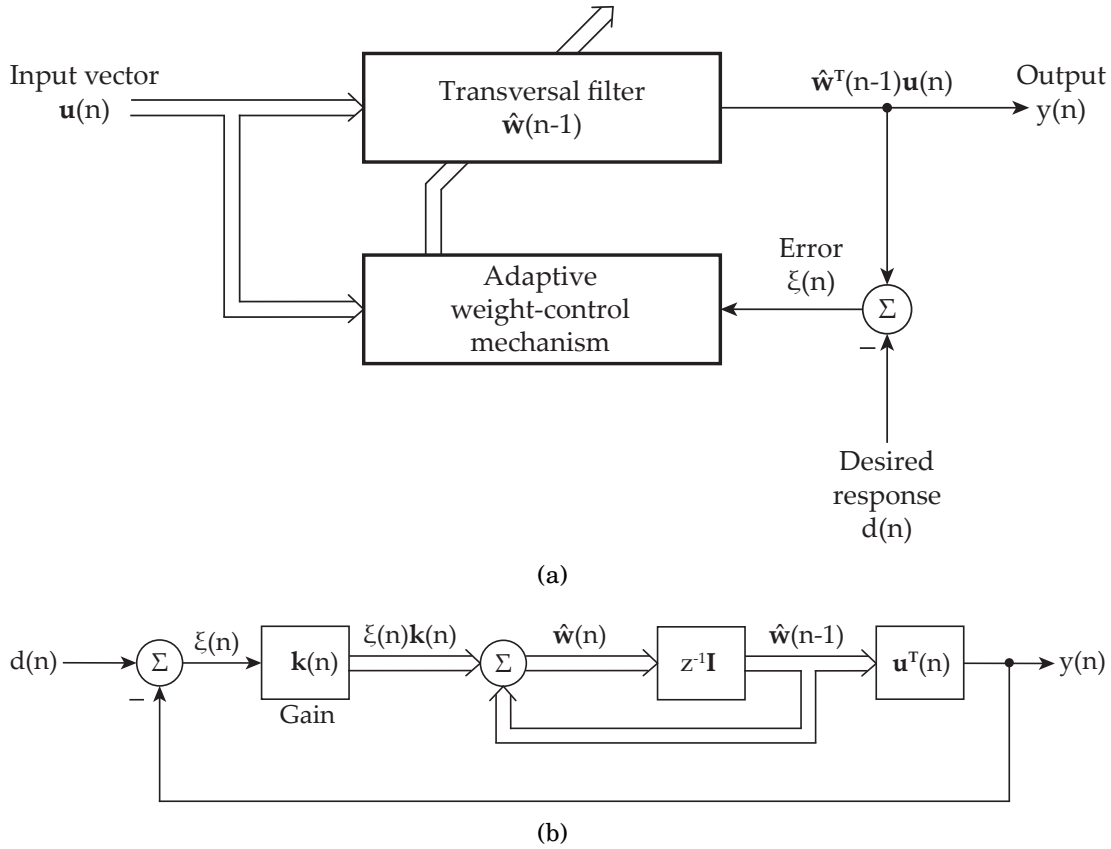


Figure 3.4: Representation of the RLS algorithm (a) block diagram, (b) signal-flow graph (figures from [32]).

$\mathbf{k}(n)$ is calculated in two stages, which is preferred from a finite-precision arithmetic point of view. Therefore we first calculate the intermediate value $\boldsymbol{\eta}(n)$ and afterwards the gain vector $\mathbf{k}(n)$, where Λ is the forgetting factor. In the next step the *a priori* error $\xi(n)$ is determined by subtracting the filter output $y(n) = \hat{\mathbf{w}}(n-1)\mathbf{u}(n)$ from the desired response $d(n)$. Finally we update the transversal filter coefficients $\hat{\mathbf{w}}(n)$ and calculate the inverse correlation matrix $\mathbf{P}(n)$ for the next iteration. To improve computational efficiency and to overcome numerical instabilities only the upper or lower triangular part of the inverse correlation matrix is computed. The rest of the matrix is filled to preserve symmetry. This is indicated by the $\mathrm{Tri}\{\cdot\}$ operator in table 3.1. We stop the RLS algorithm if $\xi(n)$ is below a certain tolerance level.

Persistent excitation of the system is necessary to find the ellipse parameters with the RLS algorithm. This means the measurement mirror must move during the RLS algorithm tries to find the ellipse parameters.

Initialisation of the algorithm:

$\hat{\mathbf{w}}(0) = \mathbf{0}$

$\mathbf{P}(0) = \delta^{-1}\mathbf{I}$

For each instant of time n = 1,2,3... compute

until $\xi(n)$ falls below a certain tolerance value:

$\boldsymbol{\eta}(n) = \mathbf{P}(n-1)\mathbf{u}(n)$

$\mathbf{k}(n) = \frac{\boldsymbol{\eta}(n)}{\Lambda + \mathbf{u}^{\mathbf{T}}(n)\boldsymbol{\eta}(n)}$

$\xi(n) = d(n) - \hat{\mathbf{w}}^{\mathbf{T}}(n-1)\mathbf{u}(n)$

$\hat{\mathbf{w}}(n) = \hat{\mathbf{w}}(n-1) + \mathbf{k}(n)\xi(n)$

$\mathbf{P}(n) = \mathrm{Tri}\{\Lambda^{-1}\mathbf{P}(n-1) - \Lambda^{-1}\mathbf{k}(n)\mathbf{u}^{\mathbf{T}}(n)\mathbf{P}(n-1)\}$

Table 3.1: Summary of the RLS algorithm [32].

In order to estimate the ellipse parameters with the RLS algorithm presented in table 3.1, we calculate the input vector $\mathbf{u}(n)$ and the desired filter response $d(n)$ for each instant of time ($n = 1,2,3...$) according to equations 3.6.

$$\begin{aligned} \mathbf{u}(n) &= \begin{bmatrix} V_{1d}^2(n) & V_{1d}(n) \cdot V_{2d}(n) & V_{1d}(n) & V_{2d}(n) & 1 \end{bmatrix} \\ d(n) &= V_{2d}^2(n) \end{aligned} \tag{3.6}$$

After convergence of the RLS algorithm the transversal filter coefficients $\hat{\mathbf{w}}(n)$ correspond to the estimated ellipse parameters as given in equation 3.7.

$$\hat{\mathbf{w}}^{\mathbf{T}} = [A \;\; C \;\; D \;\; E \;\; F] \tag{3.7}$$

Now the general ellipse equation can be formulated according to:

$$[\hat{\mathbf{w}}^{\mathbf{T}} \;\; -1] \begin{bmatrix} \mathbf{u}(n) \\ d(n) \end{bmatrix} = \hat{\mathbf{w}}_*^{\mathbf{T}} \mathbf{u}_* = 0 \tag{3.8}$$

with:

$$\hat{\mathbf{w}}_*^{\mathbf{T}} = [A \;\; C \;\; D \;\; E \;\; F \;\; B] \;, \quad \mathbf{u}_* = \begin{bmatrix} V_{1d}^2 & V_{1d}V_{2d} & V_{1d} & V_{2d} & 1 & V_{2d}^2 \end{bmatrix}^{\mathbf{T}}$$

The general ellipse equation 3.8 needs to be reformulated such that the parameter F equals -1 according to the coefficient of equation 3.4, before using it for determination of the Heydemann parameters. After reformulation of the results, obtained by the RLS algorithm, we use the estimated ellipse parameters to calculate the four quadrature error

terms a, G, p and q of the Heydemann correction using the equations

$$a = \arcsin\left(\frac{C}{\sqrt{4AB}}\right), \qquad G = \sqrt{\frac{B}{A}}, \qquad p = \frac{2BD - EC}{C^2 - 4AB}, \text{ and } q = \frac{2AE - DC}{C^2 - 4AB}. \tag{3.9}$$

Finally the errors arising from different amplitudes and offsets of the interferometer channels and the lack of quadrature can be corrected with equations 3.10.

$$\begin{aligned} V_1 &= V_{1d} - p \\ V_2 &= \frac{(V_{1d} - p)\cdot \sin(a) + G(V_{2d} - q)}{\cos(a)} \end{aligned} \tag{3.10}$$

3.1.3 Measurement of the Displacement

After the amplitude and the offset of the interferometer channels are adjusted and the Heydemann parameters are determined, the calibration of the measurement system is finished. Now the SPU enters the measurement mode. The necessary processing tasks to measure the displacement d of the measurement mirror are illustrated in figure 3.5. The digitalised and filtered detector signals V_{1d}, V_{2d} are corrected according to equations 3.10. The SPU uses the Heydemann corrected signals V_1, V_2 to calculate the current phase angle $\Theta'(n)$ according to equation 3.11. This function returns the angle Θ' in radians between the positive V_1 axis and the point determined by (V_2, V_1). The angle is positive in the upper half plane ($V_2 > 0$) and negative in the lower half plane ($V_2 < 0$).

$$\Theta'(n) = \operatorname{atan2}(V_2, V_1) = \begin{cases} \arctan\left(\frac{V_2}{V_1}\right) & V_1 > 0 \\ \arctan\left(\frac{V_2}{V_1}\right) + \pi & V_1 < 0, V_2 \geq 0 \\ \arctan\left(\frac{V_2}{V_1}\right) - \pi & V_1 < 0, V_2 < 0 \\ +\frac{\pi}{2} & V_1 = 0, V_2 > 0 \\ -\frac{\pi}{2} & V_1 = 0, V_2 < 0 \\ 0 & V_1 = 0, V_2 = 0 \end{cases} \tag{3.11}$$

After calculation of the current phase angle $\Theta'(n)$, the resulting phase angle variation $\Delta\Theta'(n)$ with respect to the previously phase angle $\Theta'(n-1)$ is calculated. This procedure is illustrated in the left diagram of figure 3.6, where the dashed lines indicate the calculated phase angles $\Theta' = \operatorname{atan2}(V_2, V_1)$. The results are depicted in the upper right diagram, where we recognize that the function exhibits discontinuities and causes phase jumps of 2π in Θ'. This will lead to wrong measurement results. To get rid of discontinuities arising from the phase estimation process the phase angle variation $\Delta\Theta'$ has to corrected

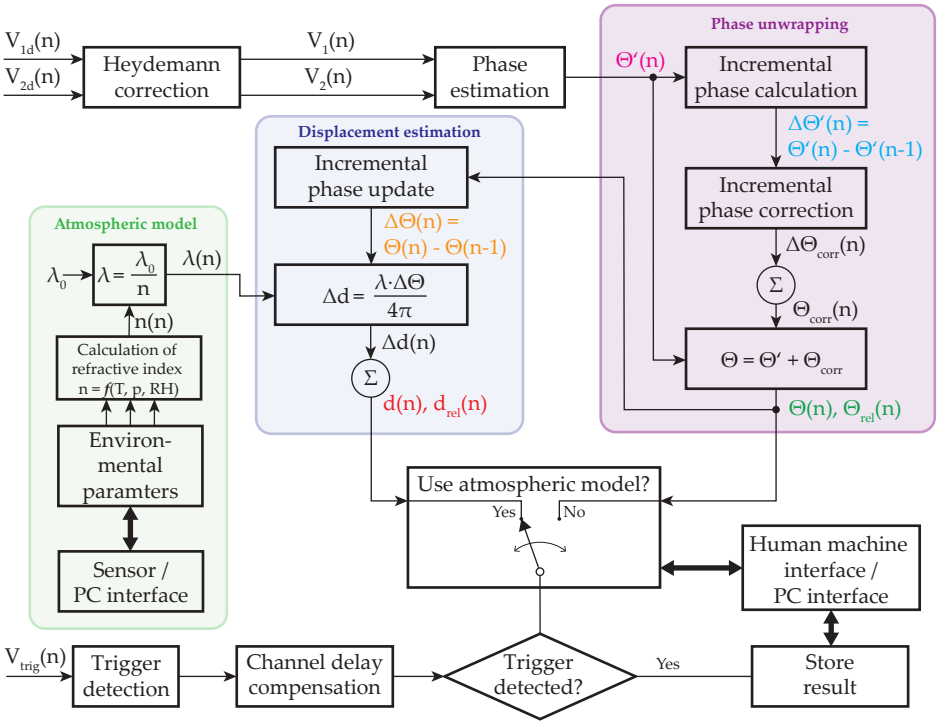


Figure 3.5: Signal flow graph to measure the phase angle Θ and the displacement d in real-time.

by Θ_{corr}. The code segment 3.1, which is an adapted form of the *unwrap* function in MATLAB, presents the method how we achieve this unwrapping of the phase angle Θ'. The diagram on the bottom right side in figure 3.6 shows the outcome of this unwrapping algorithm. As long as the phase angle variation $\Delta\Theta'$ is smaller than the defined *cutoff* value, we ignore the correction term Θ_{corr}. If the phase angle variation $\Delta\Theta'$ exceeds the defined threshold, as in case of $\Delta\Theta'(3)$, the correction term is updated accordingly in order to obtain the unwrapped phase angle variation $\Delta\Theta(3)$.
From the unwrapped phase angle Θ we obtain the unwrapped phase angle variation ΔΘ. The relationship between the unwrapped phase angle variation ΔΘ and the displacement variation Δd, given by equation 2.3, depends on the currently prevalent refractive index n and the resulting wavelength λ. The wavelength itself is a function of the vacuum wavelength λ_0 and the refractive index of air n. The latter is determined by the modified Edlén approach presented by the NIST [38]. Therefore we need to know the ambient temperature T, the barometric pressure p and the relative humidity RH. These quantities are either specified via the PC interface or measured with sensors attached to the SPU.
To obtain the overall displacement we need to accumulate the incremental displacement variations Δd at each instant of time according to equation 3.12. It is essential to apply

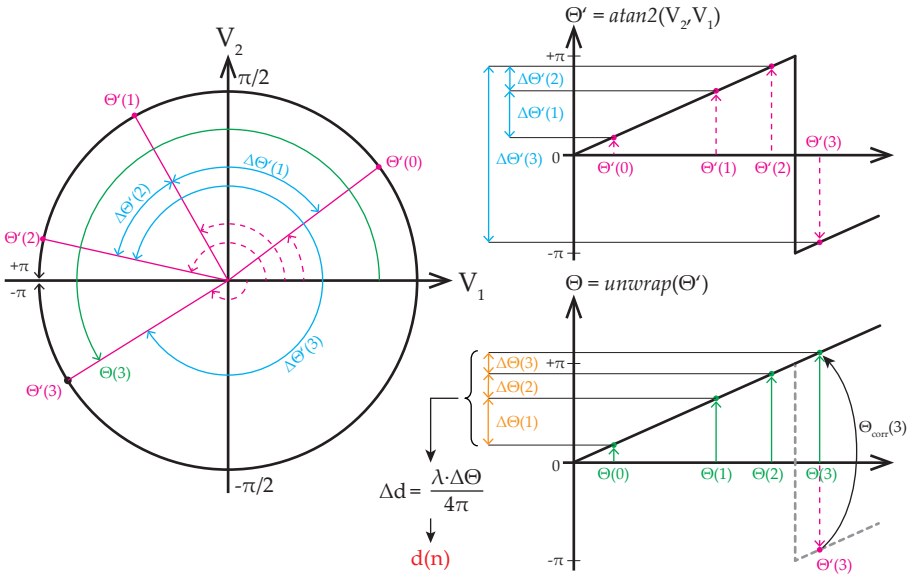


Figure 3.6: Relationship between the currently calculated phase angle Θ' by use of the atan2 function and the displacement d. Because the atan2 function exhibits discontinuities as shown in the top right corner plot we have to unwrap the results to obtain correct displacement results. The plot below illustrates the unwrapped phase angle Θ and the associated phase angle variations $\Delta\Theta$ which are used to determine the displacement d.

the atmospheric model result, the wavelength λ, to the currently determined phase angle variation $\Delta\Theta$ and not to the accumulated phase angle Θ. Otherwise a change of the refractive index of air affects the overall displacement d and leads to arbitrarily wrong results.

$$d(n) = \sum_n \Delta d(n) = \sum_n \frac{\lambda(n) \cdot \Delta\Theta(n)}{4\pi} \tag{3.12}$$

The user has the option to set a new reference point at an arbitrary position, to which the SPU refers all measurement results to. Thus we differentiate between relative Θ_{rel} and absolute accumulated phase angle Θ. The same is valid for the displacement. Equation 3.13 shows the corresponding relationship, where $\Theta(n)$ is copied to Θ_{ref} and $d(n)$ is copied to d_{ref} respectively, in case of setting a new reference point.

$$\begin{aligned} \Theta_{rel}(n) &= \Theta(n) - \Theta_{ref} \\ d_{rel}(n) &= d(n) - d_{ref} \end{aligned} \tag{3.13}$$

```
for(n=0; n<N; n++)
   // calculate incremental phase variations
   if(n==0)
      dp=theta_dash[n]-theta_dash_previous_run;
   else
      dp=theta_dash[n]-theta_dash[n-1];
   end
   // equivalent phase variations in [-pi,pi)
   dps=fmod((dp + 5*pi),2*pi)-pi;
   // preserve variation sign for pi versus -pi
   if(dps==-pi && dp>0)
      dps=pi;
   end
   // incremental phase correction
   dp_corr=dps-dp;
   // ignore correction when incr. variation is < cutoff
   if(fabs(dp)<cutoff)
      dp_corr=0;
   end
   // integrate correction and keep cumSum in memory
   cumSum+=dp_corr;
   // add the correction to the phase angle
   theta[n]=theta_dash[n]+cumSum;
   // update the displacement accordingly
   if(n==0)
      delta_d=(theta[n]-theta_previous_run)*lambda/(4*pi);
      d[n]=d_previous_run+delta_d;
   else
      delta_d=(theta[n]-theta[n-1])*lambda/(4*pi);
      d[n]=d[n-1]+delta_d;
   end
end
```

Code 3.1: Algorithm which unwraps the phase angle $\Theta'(n)$ by constraining the phase variations $\Delta\Theta'$ to the range of $[+\pi, -\pi]$ and accumulating $\Theta_{corr}(n)$. The displacement $d(n)$ is obtained by accumulation of the incremental displacement variations Δd.

In case of a trigger event the instantaneous result $d_{rel}(n)$ or $\Theta_{rel}(n)$ is stored in a result buffer. This buffer holds the past results for further processing and can be accessed via the human machine interface or the PC interface. For convenience, the setting of the trigger detection can be programmed to act on the falling or on the rising edge of

V_{trig}. Another issue concerning the trigger detection is the channel delay between the two interferometer channels and the trigger channel. To compensate this delay we have introduced an artificial trigger channel delay which is programmed once.

3.2 Realisation and Description of the Hardware

The following section gives an overview of the realised hardware. Each hardware module is described concerning its purpose, important specifications, interfaces, and connections to other modules.
Figure 3.7 an overview of the realised hardware modules and their communication interfaces. Basically we distinguish between the SPU, which consists of five stacked Printed Circuit Boards (PCBs), and three peripheral modules. The principal item of the SPU is the TMS320C6713 Evaluation Module (EVM) where the DSP is located. The 5-6k interface PCB make the DSP pins easily accessible. A split board is used to route all signals provided by the 5-6k interface PCB such that the signals keep available after stacking the PCBs. The ADC module THS1206 EVM uses the parallel port interface which is connected through the 5-6k interface and the split board. The peripheral modules and the McBsp2UART PCB communicates via the two Multichannel Buffered Serial Port (McBsp) with the DSP. The McBsp A is multiplexed on the McBsp2UART board and configured as master. The McBsp2UART board is responsible for the communication to the front end's 4-channel Digital-Analog-Converter (DAC) via Serial Peripheral Interface (SPI) protocol, to the touch display via Universal Asynchronous Serial Receive Transmit (UART) protocol, and to the sensor interface via Inter-Integrated Circuit (I^2C) protocol. Contrary to the McBsp A, the McBsp B is configured as slave and used to communicate with the FT2232H mini module, which is responsible for the data exchange to a PC. The power supply for the SPU with all stacked PCBs is provided by a common power bus which supplies the digital part with VD_{cc} and the analogue part with VA_{cc}. The realised SPU is shown in figure 3.8.
Additionally to the SPU, we realised three peripheral modules. The front end is responsible for the signal pre-conditioning of two interferometer channels and the trigger signal from an external device. The touch display provides the human machine interface to control the SPU, set-up some parameters and visualise measurement results. To facilitate communication between the DSP and a PC we use the FT2232H mini module. Except the front end, the peripheral devices are connected to the common power bus of the SPU.

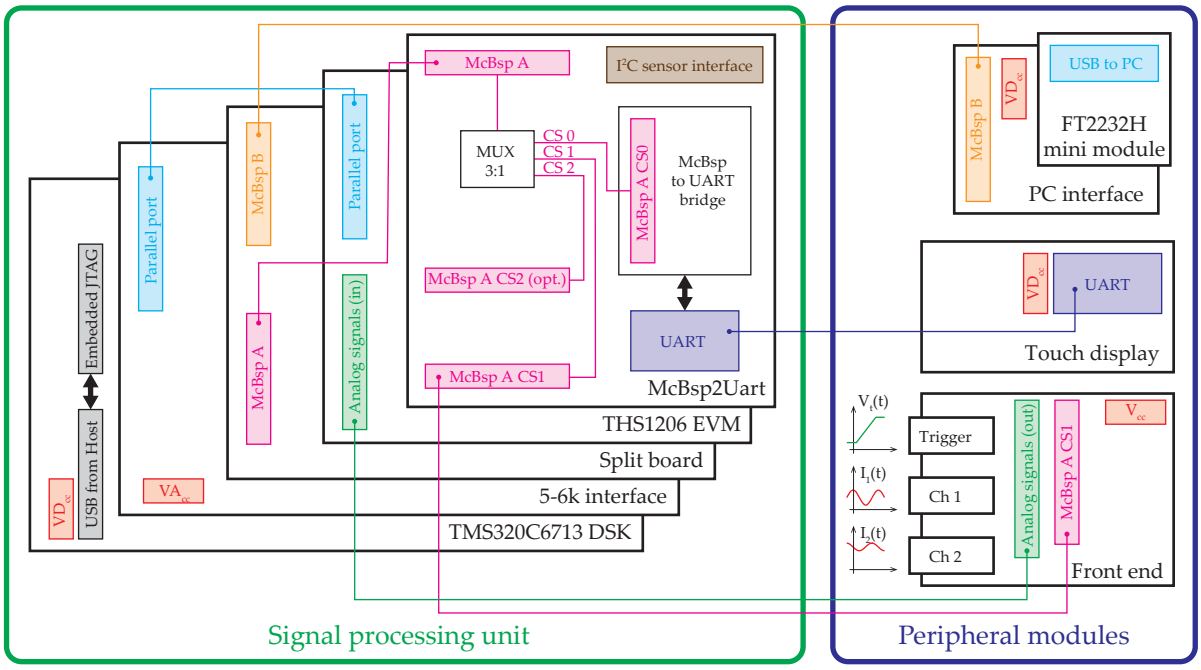


Figure 3.7: Overview of the realised hardware. It consists of the SPU and three peripheral modules.

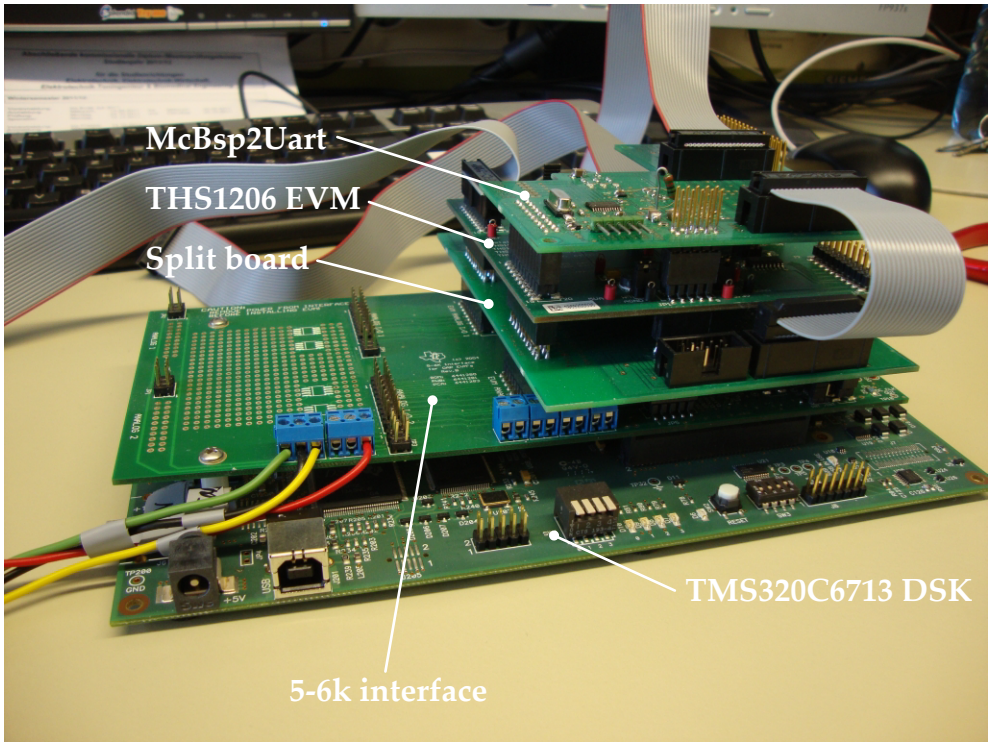


Figure 3.8: Picture of the realised SPU with the 5 stacked PCBs.

3.2.1 Front End

The HMI provides two sinusoidal current signals, obtained from intensity changes on optical detectors. The frequency of the current signals change according to the velocity of the measurement mirror. These current signals are converted into voltage signals by using the traditional operation amplifier circuit depicted in figure 3.9. Equation 3.14 characterises the transfer function of the Current to Voltage (C/V) converter stage considering an ideal Operation Amplifier (OPA).

$$\frac{V'_{1,2}}{I_{1,2}} = -Z_f(j\omega) = \frac{-R_f}{1 + j\omega R_f C_f} \tag{3.14}$$

A capacitance C_f across the feedback resistor R_f is sometimes required to suppress oscillation or gain peaking. Note that the capacitance C_f also reduces the bandwidth of the C/V converter stage. Z_f shows a low pass characteristic with a corner frequency of $f_{-3dB} = \frac{1}{2\pi R_f C_f}$. The maximum input frequency is specified by the maximum velocity of the measurement mirror which is limited by $v_{max} = 100\,\frac{\text{mm}}{\text{s}}$. This leads to a maximum signal frequency of $f_{max} \approx 316\,\text{kHz}$ according to equation 3.15.

$$f_{max} = \frac{2 \cdot v_{max}}{\lambda} \tag{3.15}$$

Generally, the signal to noise ratio of such a C/V converter circuit improves with $\sqrt{R_f}$ which means a high feedback resistor value is recommended [10]. If using a very high R_f we have to take care that the used OPA has a very small input bias current otherwise the output stage of the OPA will stay in saturation. Thus it is recommended to use an OPA with Field Effect Transistor (FET) input structures, which show very low input bias currents and very low input current noise. In order to reach highest sensitivities the photo diodes in the HMI are used in zero-bias operation. The output of the C/V converter stage provide a positive voltage with a certain amplitude and offset. A Variable Gain Amplifier (VGA) is used to adjust the amplitudes and offsets of the signals. Amplitude and offset of one interferometer channel is controlled with a DAC. Therefore we use the negative input of the VGA to shift the signal's offset towards negative voltages and the designated gain input of the VGA to change the amplitude. Additionally, we introduce a level shifting circuit to meet the given requirements of the single-ended ADC's input voltage range. The transfer function of the level shifting circuitry in case of equal resistor values R_3 is given by equations 3.16.

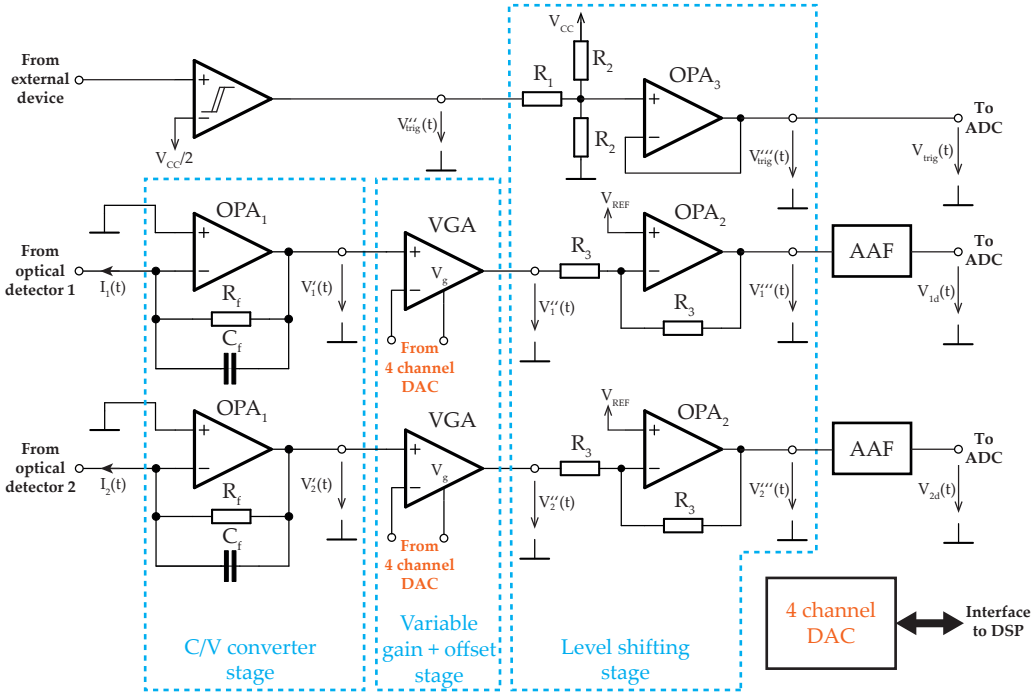


Figure 3.9: Outline of the front end unit of the SPU for two interferometer channels and one trigger input. Pre-conditioning the signals is required before they can be connected to an ADC. Signals provided from the interferometer are amplified and converted to voltage signals in the C/V converter stage. The variable gain and offset stage, controlled with a 4-channel DAC, adjust the voltage signals such that the full scale range of the ADC is utilised. Further the interferometer signals are level shifted and filtered to avoid aliasing. The trigger input comprises of a comparator, a passive voltage divider circuit and a voltage follower.

$$\frac{V'''_1}{V''_1} = -2 \cdot V_{REF} \qquad \frac{V'''_2}{V''_2} = -2 \cdot V_{REF} \tag{3.16}$$

Finally the Anti Aliasing Filter (AAF) suppresses image frequencies in the digital domain. The trigger channel consists of a comparator to guarantee fixed levels at the corresponding ADC channel. To adapt the output voltages of the comparator to the ADC specifications, we use a passive voltage divider and level shifting circuit, with a subsequent voltage follower circuit. In case of using a rail-to-rail device the comparator switches its output to either ground potential or V_{CC}. Equations 3.17 describe the characteristic of the passive

voltage divider and level shifting circuit.

$$\begin{aligned} V'''_{trig}\Big|_{V''_{trig}=0} &= V_{CC} \cdot \frac{R_1}{2R_1+R_2} \\ V'''_{trig}\Big|_{V''_{trig}=V_{CC}} &= V_{CC} \cdot \frac{R_1+R_2}{2R_1+R_2} \end{aligned} \tag{3.17}$$

Appropriate selection of R_1 and R_2 leads to an offset shifted and amplitude scaled version of the comparator's output voltage to cover the input voltage range of the ADC.

Control of the DAC channels

The output voltages of the 4-channel DAC required to adjust the offsets and amplitudes of V_1 and V_2, are determined with a simple digital control algorithm [18] illustrated in figure 3.10. The PI-control algorithm is described by equations 3.18 in form of a difference equation, where $s(n)$ and $s(n-1)$ denotes the current and the past actuating variable respectively. k_R is the control gain, T_d is the period time of the control algorithm and T_I is the time constant for the integral part of the controller. The control deviation $e(n)$ is obtained by subtracting the desired reference output r by the currently extracted parameter value $y(n)$. The extracted parameter $y(n)$ is either, a measurement of the amplitude or, a measurement of the offset. The measurement of the amplitude and the resulting control deviation is used to update the gain of the VGA, whereas the measurement of the offset and the resulting control deviation is used to update the voltage level at the negative input pin of the VGA. With the desired reference output value we are able to specify the amplitude and the offset of the interferometer channels at the ADC input. The plant consists of the VGA circuit outlined in figure 3.9, the level shifter and the subsequent AAF. This PI-control algorithm is implemented separately for each of the four DAC channels.

$$\begin{aligned} s(n) &= s(n-1) + \Delta s(n) = s(n-1) + k_R \left[e(n) - e(n-1) + \frac{T_d}{T_I} \cdot e(n) \right] \\ e(n) &= r - y(n) \end{aligned} \tag{3.18}$$

In practice the actuating variable $s(n)$ is limited by the maximum output ratings of the DAC. Thus we introduce a simple anti wind-up mechanism, where we turn off the integral part by setting T_I to a very high value, if $s(n)$ reaches the limit of the DAC output.
Note that the control algorithm is only used until the error signals $e(n)$ of the offsets and amplitudes fall below certain threshold values. If this happens, the control algorithm is turned off, the setting of the DAC channel is memorised and the output voltage remains unchanged.

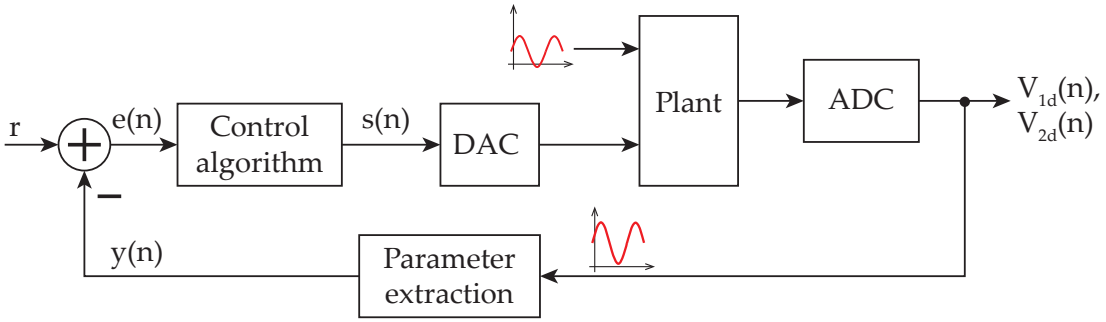


Figure 3.10: Structure of a digital controller which is used to determine the output voltages at the DAC in order to obtain the desired amplitudes and offsets of the interferometer channels at the ADC. In our case the desired reference output r is constant and the PI-controller is turned off if the error signal $e(n)$ falls below a certain threshold value. The parameter extraction depends on either, we are manipulating the amplitude or the offset value of an interferometer channel. This PI-control structure is implemented for each DAC channel separately.

The PI-controller can only find the adjustment automatically if the following conditions are met:

- The output voltages of the DAC can compensate the offsets and amplitudes such that we reach the defined threshold values where the controller algorithm is deactivated.
- While the PI-controller tries to find the settings for the DAC, a time-varying signal at the positive input pin of the VGA is necessary to manipulate the actuating variable $s(n)$ with respect to the measured amplitude and offset.

If one of the previous conditions are not fulfilled the automatic adjustment will fail. For this reason the user is able to modify the DAC output values manually with the human machine interface.

Oversampling and filtering

To keep the analog AAF simple we use an oversampling approach. In doing so the analog AAF is supported by an additional discrete AAF filter in the digital domain. This discrete AAF features much steeper slopes in the transition region and less phase distortion than the continuous counterpart. This approach requires that the band limited wanted signals, which are the pre-conditioned photo currents, are oversampled by a certain factor. The principle of such an oversampling approach is illustrated in figure 3.11. The figure shows the frequency spectra of the signal in different processing steps. $X_c(j\Omega)$ is the frequency spectrum of the continuous signal and the dashed line outlines the response of the analog AAF in the frequency domain. $X_a(j\Omega)$ sketches the result obtained by the analog AAF.

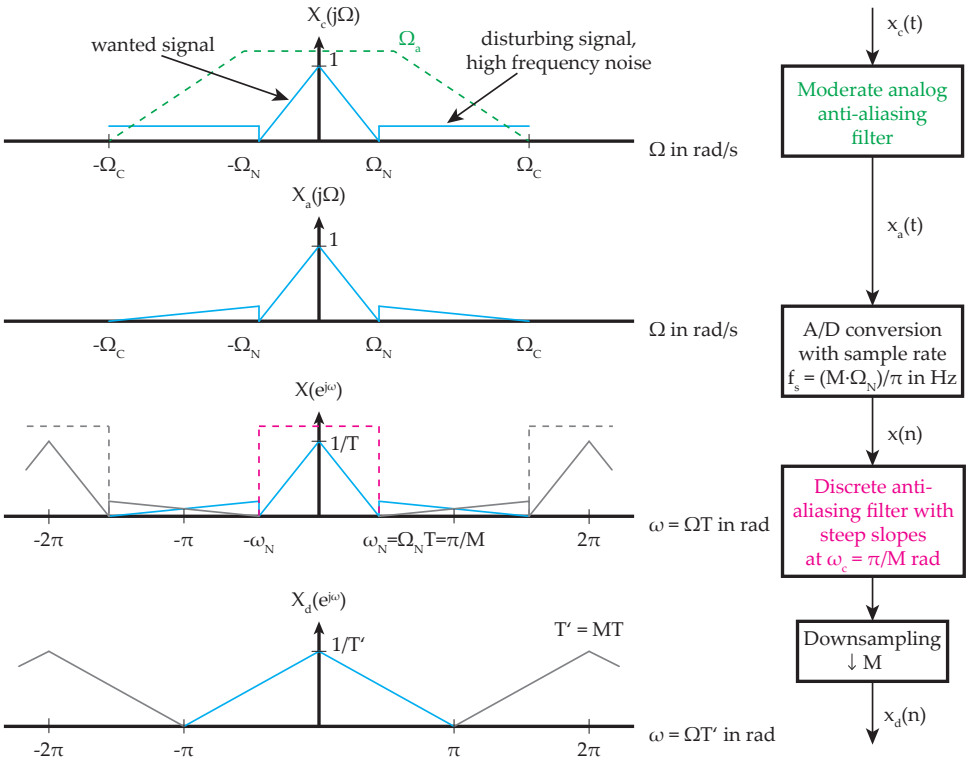


Figure 3.11: Oversampling approach with subsequent down sampling to reduce the data rate.

Analog to digital conversion of the signal leads to $X(e^{j\omega})$, where we observe aliasing in the frequency band $|\omega| > \omega_N$. However, the wanted signal is not affected as long as the cut off frequency and the slope of the analog filter leads to a sufficiently high attenuation of the mirrored spectrum of the disturbing signal at ω_N. A discrete AAF with steep slopes and a cut off frequency of $\omega_c = \omega_N = \frac{\pi}{M}$ is applied before the signal is down sampled with a maximum down sample factor of M. This leads to the frequency spectrum $X_d(e^{j\omega})$ [27]. The achievable down sampling factor M, depends on the sampling rate f_s in Hz and the wanted signal bandwidth f_N in Hz, according to equation 3.19.

$$M = \frac{f_s \cdot \pi}{\Omega_N} = \frac{f_s}{2 \cdot f_N} \tag{3.19}$$

Realisation of the PCB

The previously described front end is realised with the PCB depicted in figure 3.12. The C/V converter consists of an operational amplifier OPA140 [55] with a feedback resistor of 470 kΩ. This configuration leads to an amplification of approximately 113 dB with a phase shift of 180° according to equation 3.14. The OPA140 features a very low input

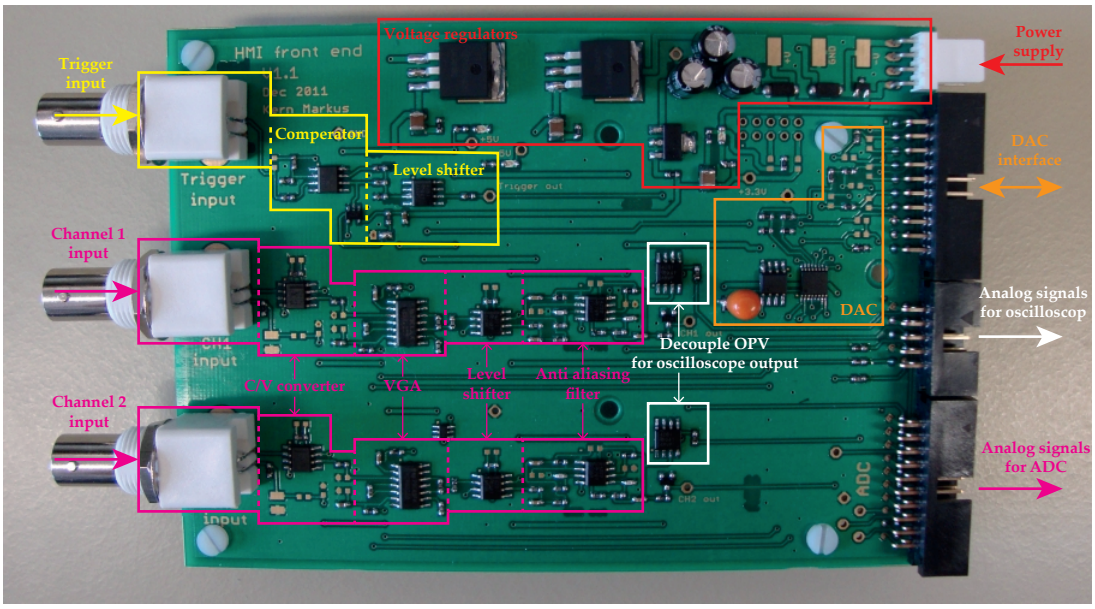


Figure 3.12: Picture of the front end PCB which pre-conditions the two interferometer channels and the trigger input.

bias current and a very low input current noise and is perfectly applicable for this task. The C/V converter stage is followed by a variable gain amplifier VCA820 [53] to adjust the amplitude and offset of the 4-channel DAC8534 [43], which is controlled by the DSP via SPI interface. The voltage range of the DAC is $0-2\,\mathrm{V}$. Note that, due to positive output voltages of the DAC, only positive offset voltages can be compensated in the interferometer channels. In practice this does not matter because the currents from the optical detectors always lead to positive offset voltages. The maximum gain of the VGA is determined by the value of two resistors. The resistors are selected such that we obtain a maximum gain of 20 dB at the VGA. This leads to an overall amplification of maximum 133 dB of the current signals from the optical detectors.

The active level shifter subsequent to the VGA shifts the offset of the signal by $+2.5\,\mathrm{V}$ to meet the input voltage requirements of the ADC. The level shifter introduces another $180°$ phase shift.

The analog AAF is realised by a Bessel type low pass filter. A Bessel type filter is characterised by a constant group delay, low ripple in the pass band and moderate attenuation in the transition region compared to other filter types. The Bessel low pass filter is designed with *Filter Solutions*. The transition region is specified such that the combination of the analog AAF and the discrete AAF applied to the oversampled digital signals, avoids aliasing in the frequency band of the wanted signal as illustrated in figure 3.11. To achieve sufficiently high attenuations we use a Bessel filter of fifth order.

Further voltage followers are used to decouple the analog signals for the oscilloscope output from the ADC input channels.
The trigger channel consists of a high speed rail-to-rail comparator TLV3501 [47] which switches the output from the negative (0 V) to the positive supply range (5 V) and vice versa if the input reaches a certain threshold. The threshold level is determined by a passive voltage divider and is set to 2.5 V. The internal hysteresis of the comparator is specified with 6 mV.
Further details to the front end PCB can be found in the schematic located in section A.3.1 and in the associated data sheets.

3.2.2 TMS320C6713 DSK

The TMS320C6713 DSK from Spectrum Digital [37] is a stand-alone platform that enables users to evaluate and develop applications for Texas Instruments' TMS320C6713 floating point DSP [49]. Figure 3.13 depicts a functional overview of the development platform which offers some on-board peripherals such as a Complex Programmable Logic Device (CPLD) to configure the development platform for different purposes, a non-volatile flash memory as booting option and 8 MByte Synchronous Dynamic Random Access Memory (SDRAM). These on-board peripherals and the memory expansion connector are connected to the External Memory Interface (EMIF) of the DSP. The memory expansion and peripheral expansion connectors form the so called daughter card interface for third party add-in boards like the 5-6k interface. The AIC23 codec, which uses the McBsp0 (= McBsp A) and the McBsp1 (= McBsp B) by default, are not used in our configuration. In order to make the McBsp interface accessible on daughter card boards the necessary bits in the MISC register of the CPLD need to be set by software. Further, the configuration switches SW3 allows the user to control the operational state of the DSP when released from reset. The necessary configurations of the switches are:

- Little endian mode (switch 1 - off), where the lowest address contains the least significant byte and the highest address contains the most significant byte of a register value.
- General Purpose Input/Output (GPIO) pins enable on the Host Port Interface (HPI) (switch 4 - on).

Figure 3.14 depicts the memory map of a generic TMS320C67xx floating point DSP on the left hand side. The right hand side illustrates the memory map of the development kit. The addressing of the program code and data is carried out by a 32-bit wide address word.

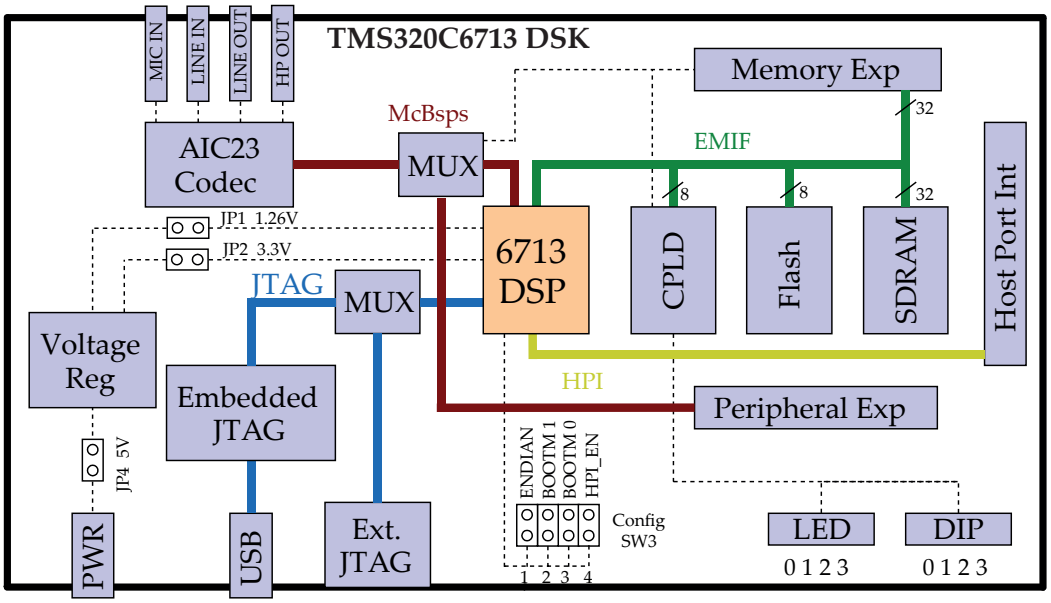


Figure 3.13: Block diagram of the development board TMS320C6713 DSK from Spectrum Digital.

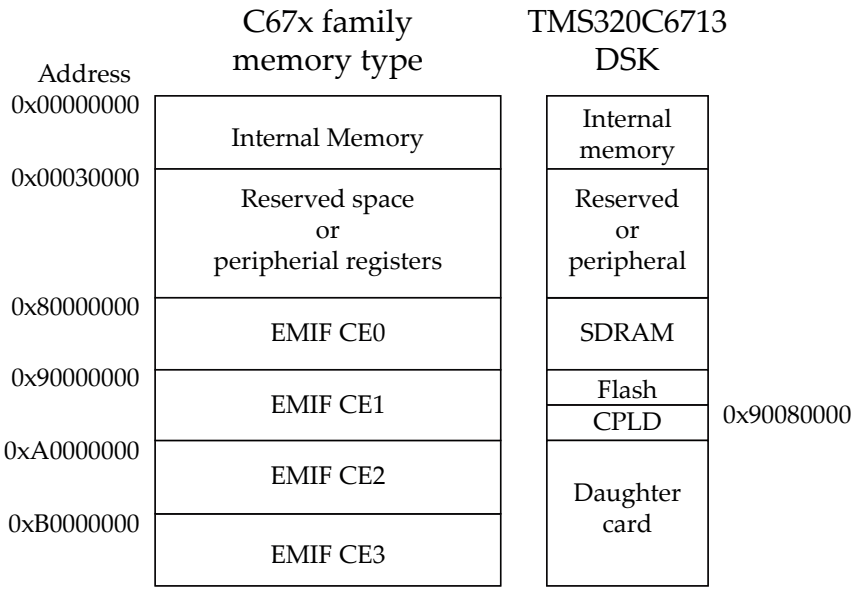


Figure 3.14: Memory maps of a generic TMS320C67xx floating point DSP (left) and of the DSK development platform (right).

The internal memory starts at the beginning of the address space. The EMIF has four separate addressable regions called chip enable spaces (CE0-CE3). The SDRAM occupies the CE0 while the flash and the CPLD share CE1. The spaces CE2 and CE3 are reserved for daughter cards.
The development board operates from a single +5V external power supply which is internally converted to different voltage levels using separate voltage regulators to supply all digital components of the board. The development board offers a Universal Serial Bus (USB) host interface to communicate with the Integrated Development Environment (IDE) of Texas Instruments.

3.2.3 5-6k Interface

This board passes signals from the TMS320C6713 DSK memory expansion and peripheral expansion connectors to use a variety of analog-to-digital or digital-to-analog converter EVM's from Texas Instruments. The 5-6k interface provides a range of connectors and has many different setting options [44]. However, the most important features and settings are:

- Power supply bus with power connections to the interface cards. Note that analog power must be supplied from an external source at the three-port screw terminals *J1* and *J2*.
- Two signal conditioning analog Input/Output (I/O) converter connectors for up two eight single-ended or four differential ADC/DAC channels.
- Two serial digital I/O connectors which gives access to the McBsp A and McBsp B respectively. These serial I/O connectors provide also access to the external interrupt pins of the DSP and some open leads for general purpose usage.
- A parallel data bus connector and the associated parallel control connector which feeds the chip-select, read, write, and address lines to parallel EVMs. These address lines are fed by a four-bit 2:1 bus switch. This provides access to DSP address lines *A2* to *A5* or *A14* to *A17* depending on the jumper setting *W1*. In our case the jumper *W1* must be removed to access DSP address lines *A14* to *A17* for proper operation of the THS1206 EVM. Further the parallel control connector feeds the clock source according to the jumper setting *J14* and the interrupt input according to the jumper setting *J13* to the THS1206 EVM. As clock source we use the *DC_TOUTb* line which corresponds to the Timer1 output of the DSP. The *DC_INTb* at the 5-6k

interface is the interrupt signal from the THS1206 EVM to the DSP. Note that the corresponding DSP signal name is *EXT_INT5*.

These settings and other jumper settings are illustrated in the schematic of the 5-6k interface in section A.3.2.

3.2.4 Split Board

The schematic of the split board can be found in section A.3.3. Beside routing all signals to keep them available after stacking the PCBs, it also duplicates the common power bus to supply the FT2232H module and the touch display. Light-Emitting Diodes (LEDs) mounted on the split board indicate the presence of every relevant voltage supply.

3.2.5 THS1206 EVM

The main item of the THS1206 EVM [41] is the 12-bit, high speed simultaneous sampling, 4-channel ADC THS1206 [42] with a maximum sampling rate of 6 MS/s. Additional to the ADC the EVM offers an address decoder logic and various settings concerning the power supply.
We use three single-ended channels of the ADC to convert the two interferometer channels and the trigger channel respectively. In this configuration the input voltage of the THS1206 ranges from 1.5 V to 3.5 V and a maximal sampling rate of 1.5 MS/s per channel is achieved.
The ADC is connected to the parallel port interface. The address of the internal buffer of the THS1206 is determined by the jumper *W4* on the THS1206 EVM. Details to the jumper settings are illustrated in the schematic in section A.3.4.
For proper operation of the THS1206 EVM we have to attach an external analog power supply (+5 V) to the 5-6k interface PCB.

3.2.6 McBsp2UART Interface Board

The McBsp2UART board is responsible for multiplexing the McBsp A from the DSP and for the conversion of the McBsp protocol to a UART protocol to exchange data with the touch display as illustrated in figure 3.15. The McBsp A is configured as SPI master. The chip select line *FSX* is multiplexed by a single-pole triple-throw switch to enable a specific slave device. We use two GPIO pins to control the switch.

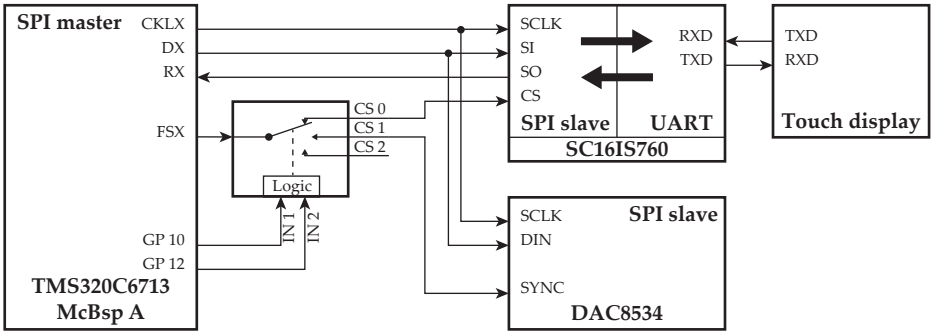
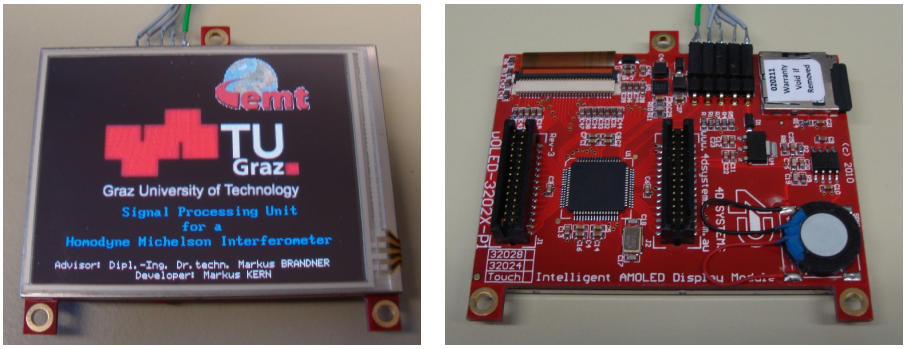


Figure 3.15: Block diagram of the McBsp A with all connected slave devices.

If the *CS0* line is active the DSP communicates with the SC16IS760 Integrated Circuit (IC) [26], which is responsible to translate the SPI protocol to a UART protocol. This IC can handle SPI data rates up to 15 MBit/s and data rates at the UART interface up to 5 MBit/s (programmable). The IC features two separate 64 Byte First In - First Out (FIFO) buffers for the transmitter and the receiver respectively.
In case of an active *CS1* line the DSP communicates with the DAC device located on front end PCB. Optionally an additional slave device can be connected to the McBsp A which is selected by setting the *CS2* line active.
The I^2C interface of the DSP, to connect digital sensors which measure the temperature, pressure and the relative humidity, is also offered at the McBsp2UART board.
In section A.3.5 the schematic of the McBsp2UART board is illustrated in detail.

3.2.7 Touch Display

The intelligent touch display module μOLED32028 P1T [1] represents the human machine interface to control the SPU and visualise measurement results. The module, depicted in figure 3.16, offers a 2.8" LED display with a resistive touch panel, which is controlled by a PICASO processor in SGC configuration. In SGC configuration the graphic processor offers a 5 pin serial interface for the host controller, which consists of a +5 V power supply, one reset, one transmit and one receive line. The graphic processor uses a UART protocol with programmable data rate (max. 256 kBit/s) to communicate with the host which is the TMS320C6713 in combination with the SC16IS760 IC.
The graphic processors offers a set of basic functions to visualise data on the display or to read coordinates from the touch panel [2]. The touch display module features also a micro Secure Digital (SD) memory card slot to read/write data from/to a micro SD memory card in File Allocation Table (FAT) format.



(a) (b)

Figure 3.16: Top and bottom side of μOLED32028 P1T display module with resistive touch panel.

3.2.8 PC Interface

For data exchange with a PC we utilise an FT2232H mini module [16] from FTDI. The FT2232H [14] on the module is a USB 2.0 high speed (480 Mb/s) to UART/Multi-Protocol Synchronous Serial Engine (MPSSE) IC with two independent configurable channels. Additionally, the module offers GPIO pins dependent on the configuration.
Figure 3.17 outlines the connection for the communication interface. Channel A of the FT2232H is connected to the McBsp B of the TMS320C6713 DSP. Both devices are configured to work in SPI 0 mode, whereas the FT2232H bridge IC is the master. Additional to the 4-wire SPI bus we use an *SPI_DATA_AV* and an *SPI_SLAVE_RESET* line. The *SPI_DATA_AV* is necessary to obtain smooth communication with the DSP, which uses the Enhanced Direct Memory Access (EDMA) controller to send/receive data to save processor resources.
Detailed information concerning the signal routing can be found in the schematic of the split board in section A.3.3. To supply the module it is connected to the common power bus of the SPU.

3.3 Implementation and Software Development

Implementation and software development on the TMS320C6000 platform with the associated hardware modules described in section 3.2 is necessary to achieve the required functionality of the SPU. All methods described in section 3.1 are implemented on the target processor using C programming language. For user friendliness and applicability of

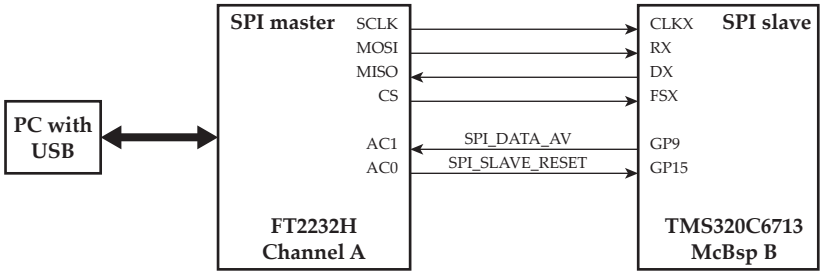


Figure 3.17: Interface between FT2232H bridge IC and the TMS320C6713 to exchange data to a PC.

the SPU a software menu, which can be accessed via the touch screen or the PC interface, is implemented. Rather than presenting detailed information about the source code, we summarise the essential implementation aspects with a special focus on sampling and digital filtering of the continuous data stream from the ADC.

3.3.1 Implementation on TMS320C6713 DSP

Development environment
Code Composer Studio (CCS) V 3.1 from Texas Instruments is used for the software development on the TMS320C6713 DSP. Note that the USB device driver for the development board's USB host interface is only available for Windows XP.

Operating system on target processor
DSP/BIOS - Cuda 4.90 [39] from Texas Instruments is used on the target processor. DSP/Bios is a real-time operating system which gives developers the ability to develop and to analyse embedded real-time software. The DSP/BIOS real-time library provides basic run-time services to embedded programs that run on the target hardware such as thread scheduling. DSP/BIOS supports four thread types with different priorities, execution and preemption characteristics. The four thread types listed from the highest to lowest priority are:

- Hardware Interrupt (HWI)
- Software Interrupt (SWI)
- Task (TSK)
- Background thread

With DSP/BIOS we create objects of a certain thread type and assign different priorities to them. Each object has an associate function which is executed if the object is scheduled

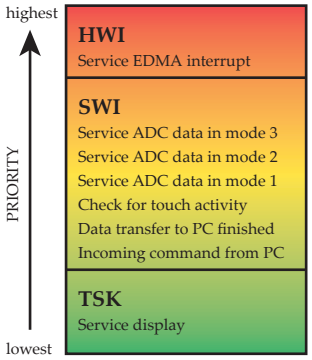


Figure 3.18: Overview of the essential functions classified by their priorities which are scheduled by the DSP/BIOS. The HWI routine decides on the TCC from the EDMA and the processing mode of the SPU, which SWI function is called. There are only two exceptions: One is the SWI service function *Check for touch activity*, which is called periodically and the other is the TSK function *Service display*, which is executed whenever necessary resources are available.

by the DSP/BIOS. Figure 3.18 illustrates the essential functions of the developed software. We use an HWI function to service an interrupt from the EDMA [50] controller, which signals a finished data transmission. Based on the provided Transfer Complete Code (TCC) from the EDMA and the current processing mode of the SPU, the HWI routine decides which SWI function has to be called for data processing. Thus software execution on the target processor depends mainly on the hardware interrupt coming from the EDMA controller and the associated TCC. In our case there are only two exceptions. The first one is the SWI function *Check for touch activities*, which periodically calls the routine to check whether there has been a touch activity on the touch display or not. Therefore we use the periodic function manager provided by DSP/BIOS. The second one is the TSK function which services the display. Different to SWI functions, TSK functions are not triggered by a call in the program code. TSK functions are executed whenever necessary resources are available on the DSP.

Interface configurations

In our case external events determine the schedule of the DPS/BIOS. Different incoming and outgoing data transfers initiate these events and direct the program execution on the DSP. The interfaces are configured as following:

- THS1206: The external memory interface EMIF [52] in combination with the EDMA controller is used to continuously receive data from the ADC. It consists of a parallel interface with data and address lines. If the EMIF puts the address *0xA0024000*

on the address bus the internal FIFO of the THS1206 is read. Two chained EDMA channels are used for continuous data exchange. The THS1206 signals the EDMA controller whenever the FIFO is full and needs to be read. To exclude data loss, this data transfer has the highest priority.

- Front end and touch display: These peripheral modules communicate with the DSP via the McBsp A (= McBsp0) [51]. This port is configured as master in SPI configuration without the usage of the EDMA controller. The timing of these transfers is not critical. Thus the transfer is handled by the Central Processing Unit (CPU) when resources are available. Additionally, two GPIO pins [46] are necessary to control the chip select line to choose between the front end, the touch display or an additional slave device.
- FT2232H module: The interface for data exchange with the FT2232H is realised by the McBsp B (= McBsp1) of the DSP. This port is configured as slave in SPI mode and uses two separate EDMA channels. One to receive data from the FT2232H and one to transmit data to the FT2232H. Although SPI is a full duplex interface, only one EDMA channel is active at a time. The *SPI_DATA_AV* line signals the FT2232H if the DSP is ready to receive or ready to transmit data. The received data packets have - different to the transmitted data packets - a fixed length of 12 Bytes. Meaning that the EDMA controller generates a CPU interrupt if 12 Bytes have been received. This interface is designed to use as little CPU resources as possible. CPU resources are only necessary to process the received data but not to handle the data transfer itself. In case of requesting measurement results the CPU moves the requested data to a buffer from where it is transmitted with the EDMA controller.
- Environmental sensors: The I^2C 0 port of the DSP can be used to communicate with sensors which measure relevant environmental parameters. However this interface is not implemented in software in the current setup. Currently the required environmental parameters to calculate the refractive index of air are programmed by default and can be modified with the PC interface.

Sampling and filtering of ADC data

The processing algorithms on the SPU must have the capability to process the continuous data stream from the THS1206 ADC fast enough. The frequency of the external event originating from the ADC is determined by its sampling rate. The ADC is configured for continuous sampling where an external clock is necessary and has to be applied to the *CONV_CLK* pin. At each falling edge of the external clock, which is derived from the

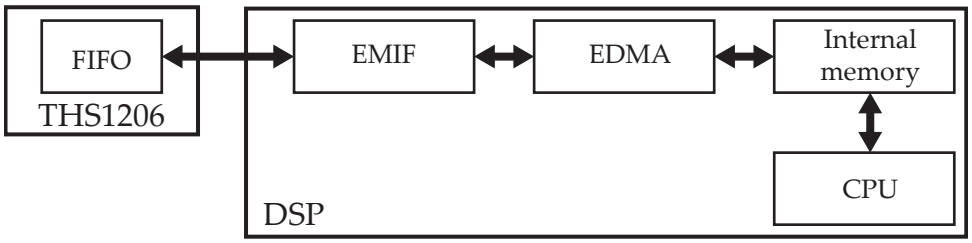


Figure 3.19: Block diagram of the data exchange between the DSP and the FIFO buffer of the THS1206 ADC.

Timer1 of the DSP, a new converted value is written into the ADC's FIFO buffer. The frequency of the Timer1 is calculated according to equation 3.20, if the internal clock source is selected by setting the register bit CLKSRC to 0 [48].

$$f_{CLK} = \frac{CPU\ clock/4}{2 \cdot Timer\ period\ register} \tag{3.20}$$

The resulting sample rate of one ADC channel in dependence of the ADC's clock input frequency f_{CLK} and the amount of used channels is calculated according to equation 3.21. After we reach a defined FIFO trigger level the *DATA_AV* pin which is connected to the *EXT_INT5* of the DSP, indicates that ADC values can be read via the parallel port interface. The ADC buffer content is accessed via address *0xA0024000*, which corresponds to the CE2 space of the EMIF (see figure 3.14).

$$f_s = \frac{f_{CLK}}{Number\ of\ ADC\ channels} \tag{3.21}$$

To process data efficiently on the DSP, data blocks are pushed through the processing algorithms instead of single ADC samples. This necessitates an intermediate data buffer. The EDMA controller is used in combination with the EMIF to fill this intermediate data buffer with data from the ADC as illustrated in figure 3.19. The usage of the EDMA controller brings the benefit that no CPU resources are required for these data transfers. Thus the CPU can process a data block while the EDMA controller buffers data from the ADC. Simultaneous memory access of the EDMA controller and the CPU to the same address has to be avoided which results in the implementation of a so called ping-pong buffer scheme. Therefore we define two EDMA channels which service two designated data buffers. If the first EDMA channel completes the transfer, it submits a new transfer for the second EDMA channel which has the identical source but a different destination address and a different TCC than the first channel. Simultaneously the EDMA controller generates a CPU interrupt (*EDMA_INT*) with a specific TCC. This informs the CPU that

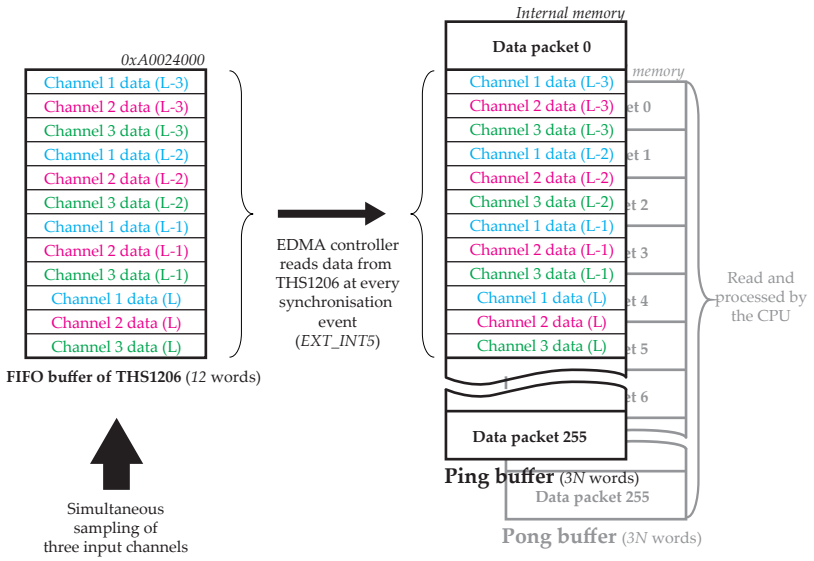


Figure 3.20: Ping-pong buffer scheme. The EDMA controller copies the content of the 12 word deep FIFO buffer of the THS1206 ADC at each synchronisation event to the ping buffer and increments the destination address automatically for the next copy until the ping buffer is full. In the mean time the CPU reads and processes the data which is stored in the pong buffer. If the ping buffer is full with new ADC samples, the EDMA controller generates a CPU interrupt and switches to the pong buffer. The content of the ping buffer is now ready to be read and processed by the CPU. This process is repeated again and again.

data, which needs to be processed, is now available in one of the two buffers. The CPU observes the delivered TCC to locate the buffer which contains the newly arrived data. Figure 3.20 illustrates such a ping-pong buffer scheme. While the EDMA transfer fills the ping buffer the data in the pong buffer is read and processed by the CPU.
There is a trade-off between buffer size and CPU interrupt frequency. Moreover it must be guaranteed that the CPU finishes the data processing for one buffer before the other buffer is filled, which causes the EDMA controller to generate a new CPU interrupt. The block size for the EDMA transfer has been specified by 3072 words and contains 256 data packets of the THS1206. Each data packet itself consists of 12 data words. As illustrated in figure 3.20 the adjacent data words of a data packet correspond to different ADC channels. Thus we have to sort the ping/pong buffer content to obtain three separated data blocks as illustrated in 3.21(a). The subsequent Finite Impulse Response (FIR) filters require the knowledge of the last processed data block. We use FIR filters with M = 128 filter coefficients. This requires the knowledge of the last M-1 = 127 input samples processed in the previous run. For this reason we need an intermediate FIR input buffer with a size of N + M - 1. Note that, FIR filtering is applied only for ADC channel 1 and

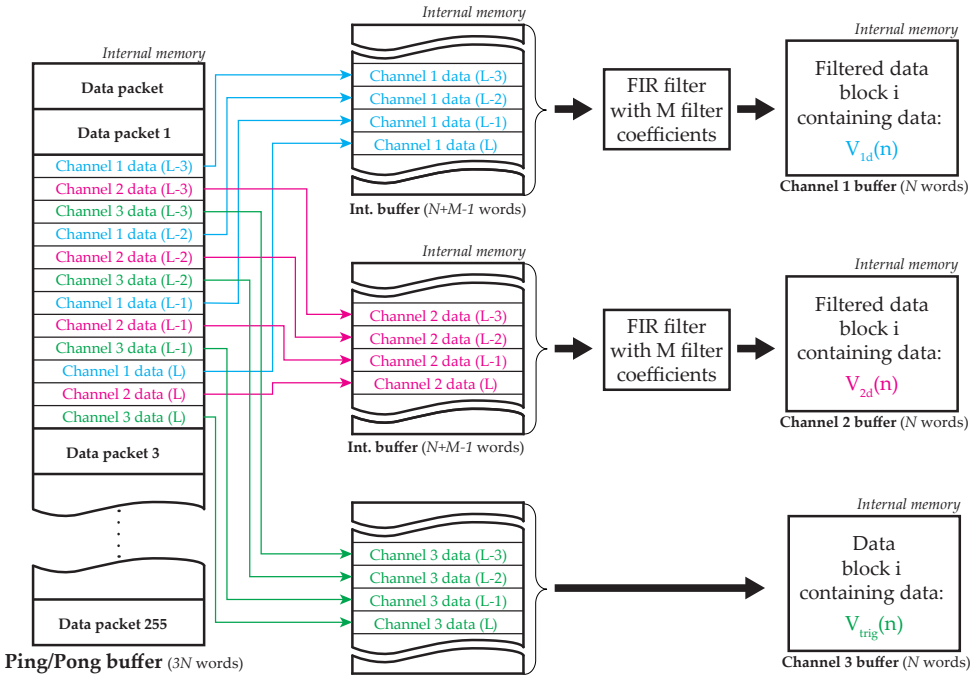


Figure 3.21: (a) The ping/pong buffer content is sorted out to obtain separate data blocks for each channel. In case of channel 1 or 2 the data is moved into an intermediate buffer which also contains data samples from the last data block i-1, to obtain correct filter results. Diagram (b) explains the implemented wrap around scheme to process consecutive data blocks of size N = 1024. Each data block is filtered by an FIR filter with M = 128 coefficients. This requires the knowledge of the last M-1 data samples of the previously processed data block i-1.

2 which correspond to the signals coming from the HMI. The implemented buffer wrap around scheme is illustrated in figure 3.21(b). The newly arrived data block i of size N is copied to the end of the intermediate buffer. The first M-1 samples in the intermediate buffer originate from the last samples of the previously processed data block i-1. The intermediate buffer content is now fed through the FIR filter with M filter coefficients. The filter outcome of an input data block is of size N. Equation 3.22 describes the filter operation for one output sample.

$$y(n) = \sum_{h=0}^{M-1} h(k)x(n-k) \tag{3.22}$$

For the FIR filter implementation itself, an optimised implementation from the digital signal processing library provided by Texas Instruments is used. The synthesis of the filter has been done with MATLAB.

Subsequent data processing

After filtering the ADC samples, they are ready for further processing. Based on the current processing mode of the SPU, illustrated in figure 3.1, the HWI function calls the corresponding C routine to process the filtered data. The methods to obtain the displacement of the measurement mirror, described in section 3.1, and the necessary routines to exchange data, to visualise measurement results, and to control the SPU are implemented in C. Therefore the following libraries are included to the CCS project:

- Chip support library *csl6713.lib* for easy access to the DSP registers [45].
- Board support library *dsk6713bsl.lib* to configure the development board.
- Digital signal processing library *dsp67x.lib* which contains assembler-optimised filters and signal transformations [54].
- Fast run-time-support (FastRTS) library *fastmath67x.lib* which contains a collection of optimised floating-point math functions [40]. Note that the linker command file needs to be updated, to use the FastRTS functions in place of the existing, standard C math functions.

The implementation of the necessary methods, described in section 3.1, is straightforward and not further discussed in this thesis.

Implementation remarks

1. The internal memory of the TMS320C6713 DSP is not sufficient for the data buffers and the source code. It is necessary to move parts of source code to the SDRAM memory section illustrated in figure 3.14. Thus we modify the linker script to put code, which is not directly involved in the data signal processing, to the slower SDRAM section.

2. For the data exchange between the SPU and the touch display a UART interface is used. The UART interface does not transmit a clock signal, hence the clock generators of the connected devices need to be synchronised. If the error between the two clock generator frequencies is too large the data transmission fails. With the current oscillating crystal attached to the SC16IS760 IC the maximum possible data rate to communicate with the touch display is 128 kBit/s.
   Further we have to wait for a data acknowledge byte from the touch display before the touch display is ready to receive the next command. The touch display sends this data acknowledge byte after it completes the requested display operation. The time between sending a command and receive the data acknowledge byte depends on the requested display operation and can be quite long. This leads to a decrease of the overall performance of the DSP, because CPU resources are required to interact with the touch display.

3. The required time to process the continuous data stream from the ADC in displacement measurement mode is the most important factor to determine the performance of the measurement system. Data blocks of size N = 3072 are processed at once and processing needs to be finished before the next data block arrives. Experiments showed that the DSP requires about 5.14 ms in measurement mode to complete this task. This leads to a theoretical limit of the ADC's sample rate of $f_s = 204.8\,\text{kHz}$ according to equation 3.23. In practice the CPU additionally has to handle incoming commands from the PC interface and the user interaction with the touch display. Thus we reduced the sample rate of the ADC to $f_s = 151.210\,\text{kHz}$.

$$f_s \leq \frac{Samples\ per\ block}{T_{processing} \cdot (Number\ of\ ADC\ channels)} \tag{3.23}$$

4. The sampling rate of $f_s = 151.210\,\text{kHz}$ is determined by performance analysis of the implemented software. However, to avoid aliasing in the frequency band of the interferometer signals, we configure the analog AAF on the front end PCB such that an attenuation of $-6\,\text{dB}$ is obtained at the frequency of 40 kHz and implement

the digital FIR filter with a cut off frequency of 10 kHz. Consequently input signals with a maximum frequency of 10 kHz can be processed. This is equivalent to a measurement mirror velocity of about 3.16 $\frac{\text{mm}}{\text{s}}$ according to equation 3.15.

3.3.2 PC Application

Microsoft Visual Studio 2008 is used to develop a Win32 console application which enables the user to communicate with the SPU. The required USB device driver (D2XX) for the FT2232H is available on the FTDI website and needs to be installed. The *libMPSSE* library [15] offered by FTDI is used to communicate with the McBsp B (SPI mode) of the SPU.
The developed PC application allows the user to read out various calibration results and position information, to transfer environmental parameters, to set the vacuum wavelength, and to control the SPU. A complete list of the implemented functions can be found in section A.2.

3.3.3 Summary

All necessary algorithms explained and described in section 3.1 are implemented and tested on the TMS320C6713 DSP. The performance of the DSP limits the maximum velocity of the measurement mirror in displacement measurement mode to 3.16 $\frac{\text{mm}}{\text{s}}$.

4 Measurement Uncertainty

This chapter is concerned with the evaluation and expression of the Measurement Uncertainty (MU) budget on the basis of GUM [21] and GUM-S1 [22]. With the knowledge of the measurement process we develop a measurement model and derive the cause-effect relationship (see also [36]). We use the measurement model for Monte-Carlo simulations to analyse how the input quantities affect the measurement result. We experimentally determine the sensitivity coefficients by evaluating the simulation outcome. As described in the GUM each input quantity is specified by its standard uncertainty u_{xi}, in order to calculate the uncertainty contribution to the measurement result and to present the MU budget. Finally the MU budget allows us to identify sources with large degree of contribution to the combined uncertainty u_y.
The evaluation of the MU is focused on the signal processing part of the measurement system. Thus we do not discuss any MU contributions arising from the mechanical assembly of the HMI and geometrical alignment of the measurement system. Those MU contributions are reviewed by Auer [6].
We use MATLAB/Simulink to implement the developed model and perform the Monte-Carlo simulation.

4.1 Measurement Process

The signal processing of the developed measurement system can be decomposed into four functional components:

- Detector unit with optical transducers
- Signal pre-conditioning unit
- Signal processing unit
- Atmospheric model

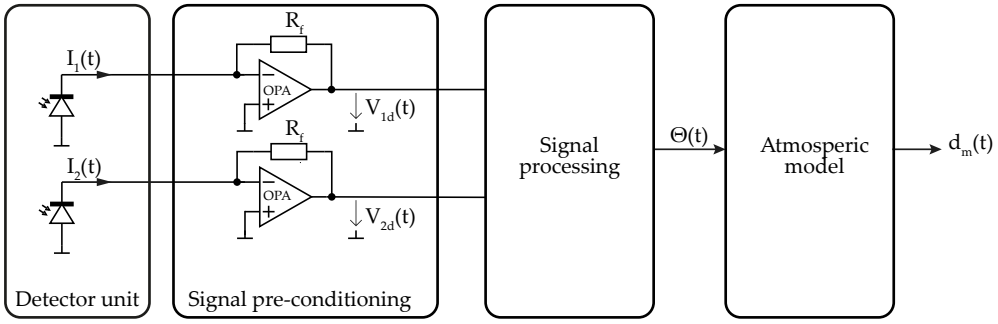


Figure 4.1: Measurement process separated into its functional components. The detector unit consists of two optical transducers. The signal pre-conditioning unit is simplified for the evaluation of the MU and consists of a C/V converter for each channel. The signal processing block is responsible for digitisation and processing of the pre-conditioned interferometer channels. Finally the atmospheric model uses the phase angle Θ to determine the displacement d_m.

Figure 4.1 illustrates the decomposition of the measurement process in its four functional components. The optical transducers of the detector unit convert the intensities of the incident laser beams into electrical currents. These current signals are converted to voltage signals, amplified, shifted, and filtered by the signal pre-conditioning unit. The signal processing unit is responsible for signal digitisation, calibration, and estimation of the phase angle Θ. To calculate a displacement we apply the atmospheric model to the estimated phase angle.
Each of these components has a certain contribution to the MU of the displacement measurement and will be evaluated in detail. However we introduce some simplifications in the measurement process concerning the signal pre-conditioning unit. We simplify the signal pre-conditioning unit to a C/V converter circuit under the following assumptions:

- The variance of the control voltages of the VGA circuit, which adapts the offset and the gain of the interferometer channels, is zero. The gain of the VGA is one.
- The variance of the voltage source used in the level shifting stage, which shifts the signal by +2.5V, is zero.
- No noise is introduced by the VGA stage, the level shifting stage and the AAF.

These neglected quantities would influence the offset, the amplitude and the SNR of the pre-conditioned detector signals V_{1d} and V_{2d}. However, the consequence of these influences to the measurement result d_m can also be analysed by manipulation of the photo currents of the optical transducers.

4.2 Cause-Effect-Diagram

The measurement result is affected by various influencing quantities, which lead to a deviation between the real displacement of the measurement mirror d_r and the measured displacement d_m. The relevant influencing quantities attributed to the functional components of the measurement process are summarised in the cause-effect diagram depicted in figure 4.2. The relevant influencing quantities are:

- Variations of the amplitude, offset and phase of the photo currents from the optical transducers.
- Thermal, shot, and quantum noise introduced by the optical transducers.
- Input current, input voltage, and thermal noise introduced by the C/V converters.
- Static errors and quantisation noise of the ADC.
- Fitting error introduced by the RLS algorithm.
- The vacuum wavelength of the laser source.
- Error introduced by the determination of the refractive index of air with the Edlén equation.
- Variations of the refractive index of air caused by changes of temperature, barometric pressure, and humidity.

4.3 Modelling and Model Implementation

This section describes the development and the functionality of the model, which is used for Monte-Carlo simulations to empirically determine the sensitivity coefficients of relevant influencing quantities. Based on the knowledge of the measurement process and the influencing quantities summarised in the cause-effect-diagram a model for Monte-Carlo simulations is developed. Figure 4.3 gives an overview of the developed model under consideration of the relevant influencing quantities. Θ denotes the digitised phase angle, n denotes the refraction index of ambient air, and λ_0 denotes the vacuum wavelength of the laser source. Equation 4.1 is the model equation derived from the model depicted in figure 4.3.

$$\begin{aligned} \Delta d_r = {} & \delta_{Amp} + \delta_{Off} + \delta_{Pha} + \delta_{TND} + \delta_{SND} + \delta_{QND} + \delta_{TNS} + \delta_{IVN} + \delta_{ICN} \\ & + \delta_{SE} + \delta_{QN} + \delta_{RLS} + \frac{\Delta\Theta \cdot \lambda_0}{4\pi \cdot n} + \delta_{Temp} + \delta_{Pres} + \delta_{Hum} + \delta_{EE} + \delta_{VWL} \end{aligned} \tag{4.1}$$

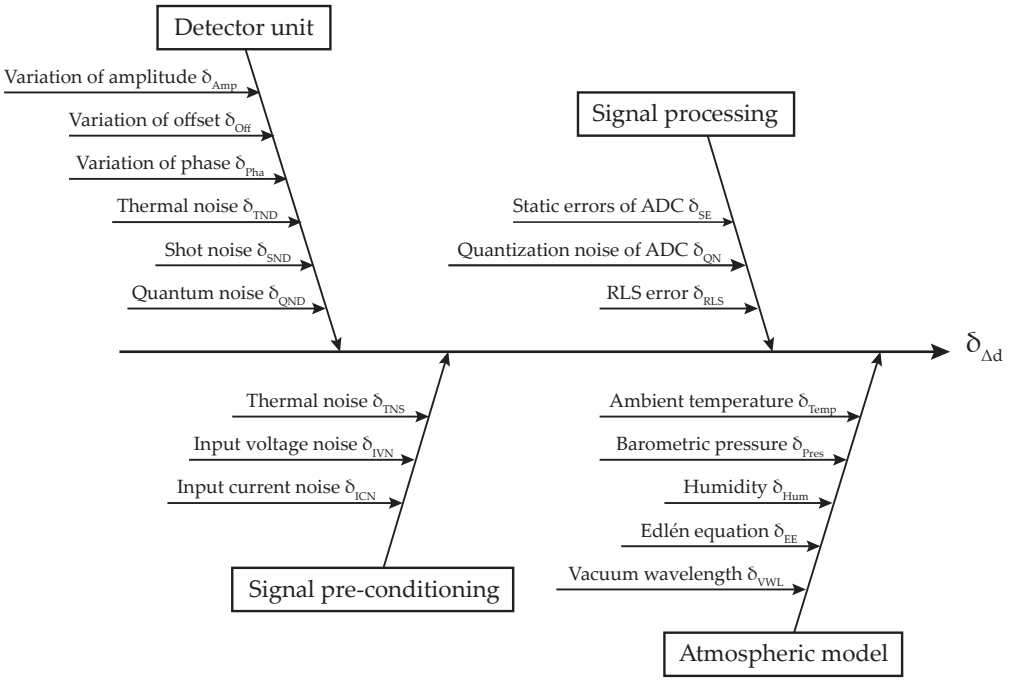


Figure 4.2: Cause-effect diagram summarising relevant influencing quantities.

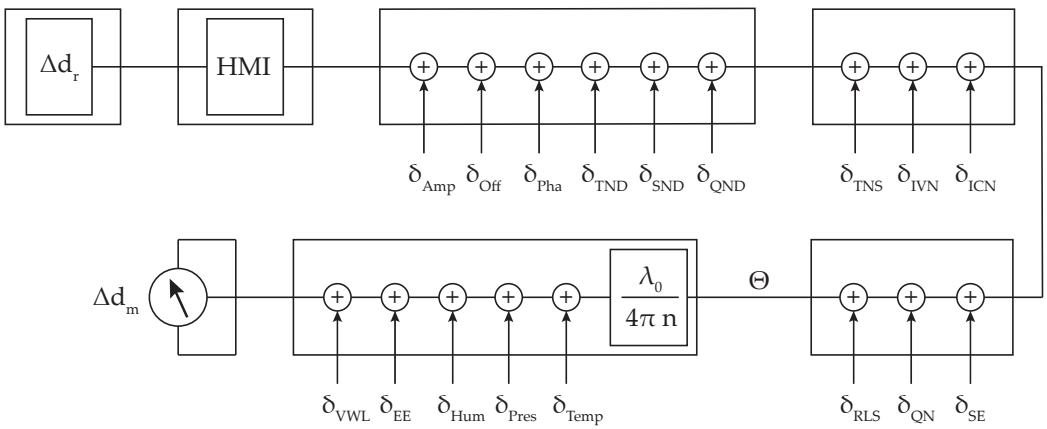


Figure 4.3: Overview of the model for Monte-Carlo simulations under consideration of the relevant influencing quantities. The influencing quantities affect the measurement result and leads to a deviation of the real displacement d_r and the measurement displacement d_m.

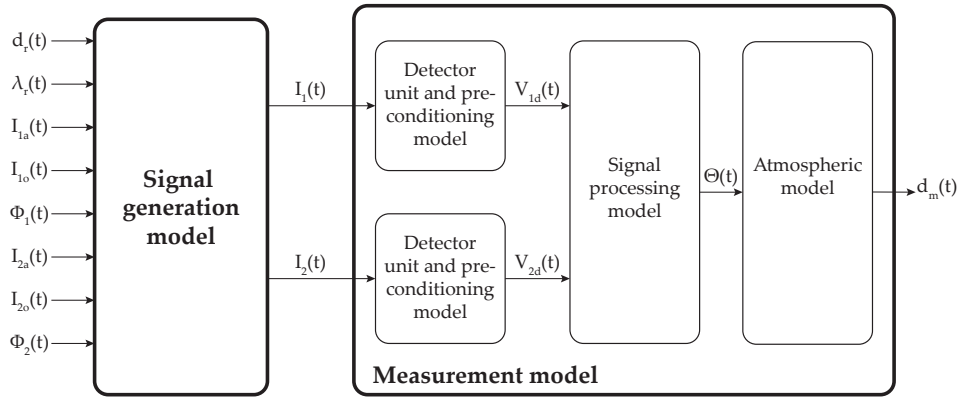


Figure 4.4: Signal generation and measurement model for the Monte-Carlo simulation. The input quantities of the signal generation model define the signal characteristics of the photo currents I_1 and I_2. In calibration mode the photo currents are used in the measurement model to estimate the ellipse parameters and to calculate the Heydemann parameters. In measurement mode the calculated Heydemann parameters are used to correct the photo currents, to finally estimate the displacement d_m.

For the implementation of the model we differentiate between a signal generation model and a measurement model as illustrated in figure 4.4. The interface between the signal generation and the measurement model are the photo currents I_1 and I_2 of the photo detectors generated by the incident coherent laser light. The measurement model uses these photo currents for two different operating modes. In calibration mode the measurement model estimates the ellipse parameters with a direct least squares approach to calculate the Heydemann parameters. In measurement mode the calculated Heydemann parameters are used to compensate the lack of quadrature of the input signals. Further we use the information about the phase angle Θ, the measured environmental variables and the known vacuum wavelength of the stabilised laser λ_0, to estimate the displacement of the measurement mirror d_m.

4.3.1 Signal Generation Model

The signal generation model simulates the behaviour of the HMI and provides two current signals for the subsequent measurement model. As depicted in figure 4.4 the signal generation model uses the input quantities displacement d_r, wavelength of the stabilised laser in ambient air λ_r, amplitude, offset and phase delay of the signals to determine the output of the HMI in form of the photo currents I_1 and I_2.

$$
\begin{aligned}
I_1(t) &= I_{1a}(t) \cdot \left(1 + \cos\left(\frac{4\pi \cdot d_r(t)}{\lambda_r(t)} + \Phi_1(t)\right)\right) + I_{1o}(t) \\
I_2(t) &= I_{2a}(t) \cdot \left(1 + \sin\left(\frac{4\pi \cdot d_r(t)}{\lambda_r(t)} + \Phi_2(t)\right)\right) + I_{2o}(t)
\end{aligned}
\tag{4.2}
$$

Equations 4.2 describe the transfer function of the signal generation model and are related to equations 2.4. The term $4\pi \cdot d_r/\lambda_r$ determines the phase angle according to the position of the measurement mirror and the wavelength of the laser source in ambient air. We can specify the measurement mirror movement with a path-time diagram with an arbitrary characteristic. If the velocity of the measurement mirror and the wavelength of the laser sources are constant over time, the frequency of the generated photo currents does not vary. However, the wavelength λ_r depends on the vacuum laser frequency λ_{0r} and the environmental parameters temperature T_r, barometric pressure p_r, relative humidity RH_r, and CO_2 concentration. Based on the knowledge of the previously stated parameters, we use the NIST approach [38] to estimate the wavelength in ambient air. With Φ_1 and Φ_2 we are able to determine the phase of the two interferometer channels. The phase difference $\Delta\Phi$, between the two interferometer channels, is defined by equation 4.3. In practice the parameter $\Delta\Phi$ is an interferometer constant and does not vary over time or displacement.

$$
\Delta\Phi = \Phi_2 - \Phi_1 \tag{4.3}
$$

4.3.2 Measurement Model

The behaviour of the photo detectors, the signal pre-conditioning unit, the signal processing and the application of the atmospheric model is implemented in the measurement model in order to reach the final goal, the estimation of the measurement mirror displacement d_m. Details about the modelling and the implementation of model components are discussed in the following section.

Detector and signal pre-conditioning unit

The detector and the signal pre-conditioning unit are analysed in detail concerning their noise contributions. The output of the signal generation model, the photo currents I_1 and I_2, are idealised and therefore noiseless signals. The photo detectors and the C/V circuit exhibit different noise sources with different transfer characteristics to the output. To analyse the noise sources, we introduce a noise equivalent circuit to derive the transfer function of each source. With this information we are able to implement each

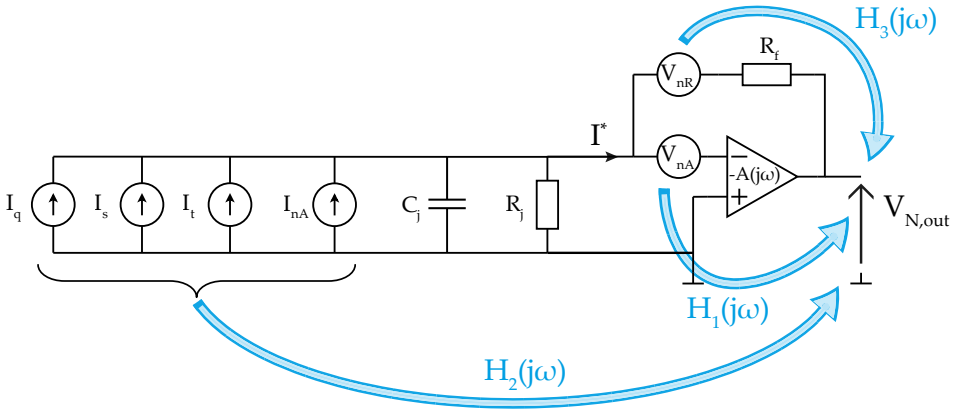


Figure 4.5: Noise equivalent circuit of the photo detector and the signal pre-conditioning unit and the associated transfer functions of the noise sources.

noise source into our measurement model and determine the corresponding sensitivity coefficients. Figure 4.5 shows the equivalent noise circuit of the photo detector and the signal pre-conditioning unit. Noise sources related to the photo detector are:

- Thermal noise I_t with a standard uncertainty of $u_{I_t} = \sqrt{\frac{4kT_{abs}B}{R_j}}$
- Shot noise I_s with a standard uncertainty of $u_{I_s} = \sqrt{2eI_dB}$
- Quantum noise I_q with a standard uncertainty of $u_{I_q} = \sqrt{2eIB}$

Noise sources related to the signal pre-conditioning unit are:

- Input current noise I_{nA} of the OPA with a standard uncertainty of $u_{I_{nA}}$
- Input voltage noise V_{nA} of the OPA with a standard uncertainty of $u_{V_{nA}}$
- Thermal noise of the feedback resistor R_f with a standard uncertainty of $u_{V_{nR}} = \sqrt{4kT_{abs}R_fB}$

Where k denotes the Boltzmann constant in J/K, B the bandwidth in Hz, e the elementary charge in C and T_{abs} the absolute temperature in K. R_j is the shunt resistance of the diode junction, I_d is the dark current, and I the photo current of the photo detector. In order to determine the bandwidth B to calculate the standard uncertainties of the noise sources we use the cut off frequency of the analog AAF, defined in section 3.3.1, instead of calculating the equivalent noise bandwidth of the filter. Especially if dealing with low pass filters of higher order, this approach is often applied in practice and the resulting errors are neglected (cf. [56, p. 489]).
While the spectral distribution of the thermal, shot, and quantum noise sources are constant over frequency (white noise), the amplitude distributions are normal with zero-mean.

The input current and input voltage noise densities of the OPA show different behaviour concerning their spectral distributions. At high frequencies the spectral distribution is constant over frequency (white noise) but at low frequencies the spectral distribution rises at 10 dB/decade. This noise is also known as $1/f$ noise. The transition between the white and the $1/f$ region is specified by the corner frequency f_c, which can be found in the data sheet of the OPA. To calculate the noise contribution of the input current and input voltage noise of the OPA we use the equation 4.4 and 4.5 respectively (cf. [4]).

$$u_{V_{nA}}(f_l, f_h) = v_{nw}\sqrt{f_c \cdot \ln\left(\frac{f_c}{f_l}\right) + (f_h - f_c)} \tag{4.4}$$

$$u_{I_{nA}}(f_l, f_h) = i_{nw}\sqrt{f_c \cdot \ln\left(\frac{f_c}{f_l}\right) + (f_h - f_c)} \tag{4.5}$$

f_l and f_h specifies the bandwidth of interest, v_{nw} the voltage noise density in $\mathrm{V}/\sqrt{\mathrm{Hz}}$ and i_{nw} the current noise density in $\mathrm{A}/\sqrt{\mathrm{Hz}}$ in the white region.
Under the assumption that all noise sources are uncorrelated we can replace the parallel input noise sources by a current noise source with a standard uncertainty of $u_{I^*} = \sqrt{u_{I_t}^2 + u_{I_s}^2 + u_{I_q}^2 + u_{I_{nA}}^2}$. Further we use the law of superposition to derive the transfer functions $H_i(j\omega)$.
In doing so, we need to replace all voltage noise sources by a short-circuit and all current noise sources by an open-circuit. Only the *i-th* noise source stays active during the determination of the transfer function $H_i(j\omega)$. Figure 4.6 shows the equivalent circuit to determine the transfer function $H_1(j\omega)$ of V_{nA}. The voltage noise source V_{nR} is short-circuited and all current noise sources are replaced by an open-circuit. To derive the transfer function $H_1(j\omega)$ we assume an ideal OPA where the input current I_{in} goes to zero. Keeping this fact in mind we obtain equation 4.6 by observing the meshes *i* and *ii* in figure 4.6.

$$I^* = I_{Rf}$$

$$V_{nD}\left(\frac{1}{R_j} + j\omega C_j\right) = \frac{1}{R_f}\left(V_{1,out} - V_{nD}\right) \tag{4.6}$$

Mesh *iii* in figure 4.6 leads to:

$$V_{nD} = V_{nA} - \frac{V_{1,out}}{A(j\omega)} \tag{4.7}$$

Now we substitute V_{nD} of equation 4.6 by the expression in equation 4.7 and transform the equation to obtain $V_{1,out}$. This leads us to the following expression 4.8 which can be

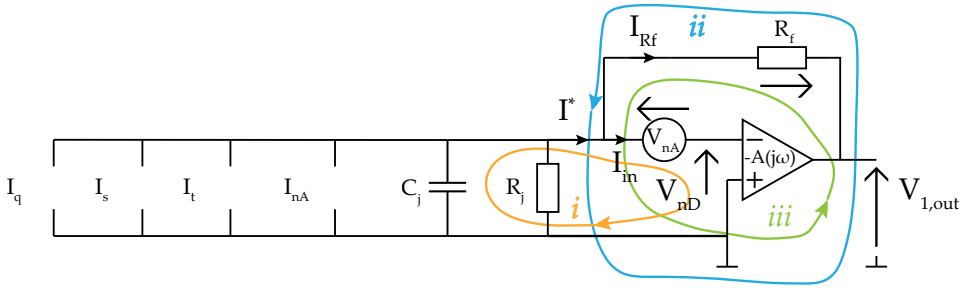


Figure 4.6: Noise equivalent circuit of the photo detector and the signal pre-conditioning unit to determine the transfer function $H_1(j\omega)$ of V_{nA}.

simplified under the assumption that $A(j\omega) >> R_f/R_j$ to equation 4.9

$$V_{1,out} = \frac{V_{nA}\left(\frac{1}{R_f} + \frac{1}{R_j} + j\omega C_j\right)}{1 + \frac{1}{A(j\omega)} + \frac{R_f}{R_j A(j\omega)} + \frac{j\omega C_j R_f}{A(j\omega)}} \tag{4.8}$$

$$V_{1,out} \approx V_{nA}\left(1 + \frac{R_f}{R_j} + j\omega C_j R_j\right) \tag{4.9}$$

Finally the transfer function of $V_{1,out}$ can be expressed by equation 4.10.

$$H_1(j\omega) = \frac{V_{1,out}}{V_{nA}} \approx \left(1 + \frac{R_f}{R_j} + j\omega C_j R_j\right) \tag{4.10}$$

The derivation procedure of the transfer functions $H_2(j\omega)$ and $H_3(j\omega)$ is identically to the one of $H_1(j\omega)$ and not demonstrated in detail. However, equations 4.11 and 4.12 present the result of the derivation under the same assumption as before, namely that $A(j\omega) >> R_f/R_j$.

$$H_2(j\omega) \approx -R_f \tag{4.11}$$

$$H_3(j\omega) \approx 1 \tag{4.12}$$

Notice that the current noise sources at the input undergo the same transfer characteristic than our wanted signal, the photo current I, which already indicates that these noise contributors are the most corruptive ones. Anyway, $H_1(j\omega)$, $H_2(j\omega)$ and $H_3(j\omega)$ represent the transfer functions of different sources from the photo detector to the output of the signal pre-conditioning unit and are used to implement the noise sources into the measurement model as illustrated in figure 4.7.

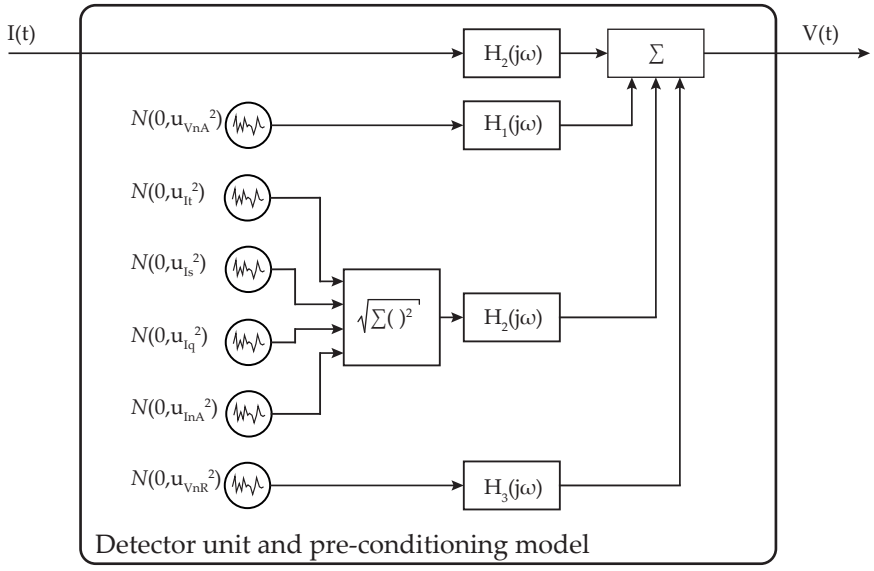


Figure 4.7: Detector and simplified pre-conditioning model of one interferometer channel with related noise contributors. Note that some noise sources show the same transfer characteristic than the wanted signal I.

Analog to digital converter

The ADC model consists of a continuous non-linear function $g(x)$ cascaded with an ideal quantizer $quant(g(x))$ which is illustrated in figure 4.8. The error model reflects the static errors introduced by the gain, offset and integral non-linearity, as well as the quantisation of the ADC. This error model is very simple and does not take into account any non-linear dynamic effects such as non-linear dynamic transformations of the input signal and timebase errors (aperture uncertainty or the jitter in the sample clock signal). Thus, the model is applicable for input signals with slow temporal variations with respect to the aperture time and the bandwidth of the ADC (cf. [5]).

Uncertainties of gain u_g, offset u_o and integral non-linearity u_{inl} are covered by the non-linear function $g(x)$. Under the assumption that these uncertainty sources are normally distributed with zero-mean and independent to each other we are able to model these uncertainties by an additive noise source with a standard uncertainty given by equation 4.13.

$$u_{se} = \sqrt{u_g^2 \cdot FSR^2 + u_o^2 + u_{inl}^2} \tag{4.13}$$

$$u_{qn} = \frac{1}{\sqrt{12}} \cdot \frac{FSR}{2^{NOB} - 1} \tag{4.14}$$

The cascaded ideal quantizer $quant(g(x))$ introduces an uncertainty due to quantization of the input signal and is implemented by a standard Simulink block, which can be bypassed to deactivate this uncertainty contributor. However, we use equation 4.14 to

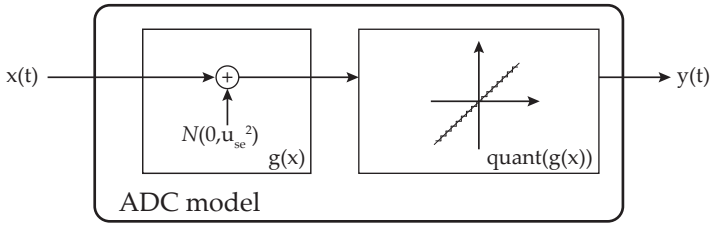


Figure 4.8: ADC model. Gain, offset and integral non-linearities of the ADC are simulated with an additive noise source $N(0, u_{se}^2)$. To introduce a quantisation effect we use a standard quantiser block of Simulink.

calculate the standard uncertainty of the quantisation noise. *FSR* denotes the full scale range and *NOB* the number of converter bits of the ADC.

Signal processing on the DSP

The task of the DSP is to estimate the phase angle Θ and to translate it into a displacement d_m. Therefore we use the digital signal processing model of the DSP in two different operation modes.

In calibration mode we use the output of the ADC model to estimate the ellipse parameters and to subsequently calculate the Heydemann parameters. The Heydemann parameters are necessary to compensate the lack of quadrature of our interferometer channels. Instead of using the RLS algorithm for ellipse fitting as implemented on the DSP and described in section 3.1.2, we use a different method based on a direct least square approach published by Fitzgibbon *et al.* [13]. Details to this approach can be found in the Appendix. We consider the result of this direct least square ellipse fitting algorithm as a reference standard to test the implemented RLS algorithm of the DSP.

In measurement mode we apply the Heydemann correction, with the previously calculated parameters, to the interferometer channels to estimate the phase angle Θ of the resulting quadrature signal. By phase unwrapping and by use of the atmospheric model we obtain the displacement of the measurement mirror d_m.

Figure 4.9 illustrates the signal processing and subsequent atmospheric model. In order to take the error between the direct least square approach and the RLS approach into account, we add an additive multivariate normally distributed noise source to the Heydemann parameter vector in our model. The parameters of this multivariate normally distributed noise source are specified in section 4.4.5. Another contributor to the MU budget is the atmospheric model. The estimated wavelength λ_e of the laser source in ambient air depends on the vacuum wavelength of the laser λ_0 and the measured environmental parameters temperature T_m, barometric pressure p_m, humidity RH_m,

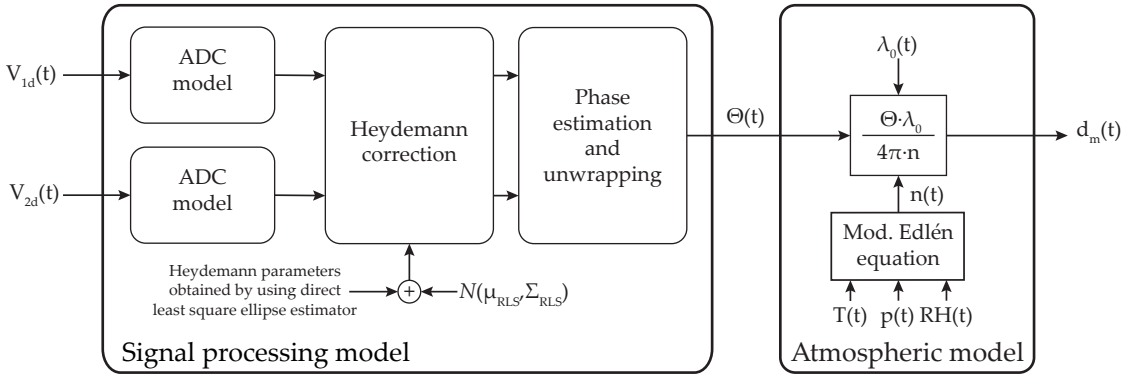


Figure 4.9: Signal processing and subsequent atmospheric model considering the modelling error introduced by the RLS algorithm.

and the CO_2 concentration. The measurements of T_m, p_m, and RH_m have a certain measurement uncertainty. Moreover, the calculation of the refractive index of air n with the modified Edlén equation [38] has an uncertainty due to modelling errors. This leads to a deviation between the real wavelength of the laser λ_r, specified as input quantity of the signal generation model (see figure 4.4), and the estimated one.

4.3.3 Summary

In the previously described model we introduced the relevant influencing quantities which affects the measurement result as demonstrated in the cause-effect diagram in figure 4.2. The developed model is implemented in Simulink and controlled by MATLAB scripts to manipulate the properties of the Simulink blocks and to generate the characteristics of some input variables. In the next section we use the model to accomplish the Monte-Carlo simulation to determine the sensitivity coefficients of the influencing variables.

4.4 Simulation Studies

With the model described in section 4.3 we study how the influencing variables considered in the cause-effect diagram (see figure 4.2), affect the measurement result. We run different scenarios to determine the influence of each input quantity. Basically we distinguish between four different simulation scenarios:

- Sensitivity analysis of noise sources
- Sensitivity analysis of atmospheric influences and vacuum wavelength

- Sensitivity analysis of photo current changes
- Influence of the RLS ellipse fitting algorithm

This simulation scenarios are analysed in detail in order to find the sensitivity coefficients of each influencing quantity, which are necessary for the MU budget. Therefore we measure the change of the output quantity by varying a particular input quantity, while holding the remaining input quantities constant. The ratio between the standard deviation of the resulting model values and the standard uncertainty associated with the best estimate of the relevant input quantity can be taken as a sensitivity coefficient [22].

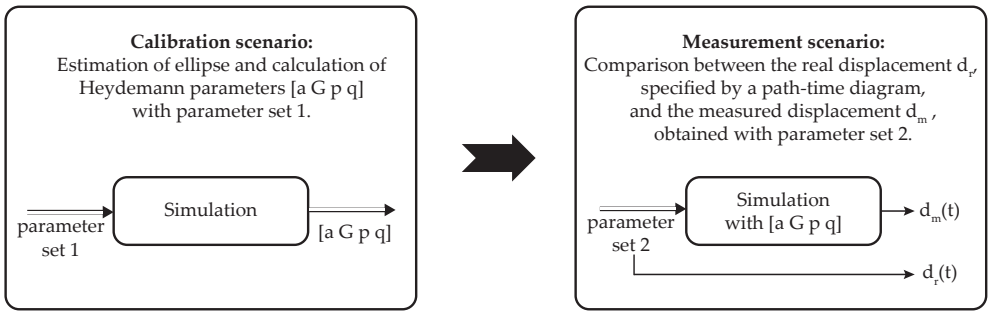


Figure 4.10: Principle of a simulation. A calibration is necessary to obtain the Heydemann parameters before simulating a measurement. The parameter sets control properties of the noise sources, the input quantities and the Simulink blocks.

4.4.1 Calibration of the Model

In order to simulate a displacement measurement a calibration needs to be performed, where the ellipse is estimated and the Heydemann parameters are calculated. Therefore we specify a parameter set 1 to determine the properties of all Simulink blocks and input quantities. Basically this calibration is performed without any source of interference, to find the best estimate of the Heydemann parameters with respect to a given parameter set 1. After calibration we can simulate a measurement with a separate parameter set 2. The measurement parameter set allows us to activate noise sources, change the input quantities of the signal generation model, or change the properties of the Simulink blocks. This is the way how we can analyse the behaviour of the measurement outcome by varying input quantities and keeping the calibration unmodified. Figure 4.10 illustrates this procedure. Finally, we compare the deviation between the real and the measured displacement $e_d = d_r - d_m$ with respect to an activated influencing variable to find the

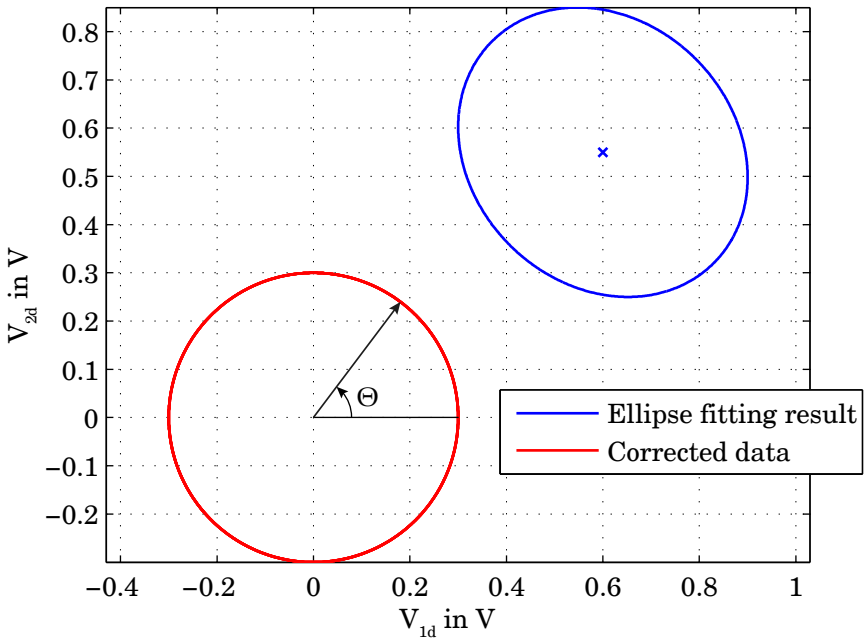


Figure 4.11: Result of the calibration scenario. The xy-plot shows the ellipse fitting result and data after Heydemann correction. Θ denotes the instantaneous phase angle.

associated sensitivity coefficient.
In our case, finding sensitivity coefficients is not a trivial task because the measurement uncertainties depend on the input data. Thus we are using following parameters for the simulation studies unless otherwise noted:

- Amplitudes of the photo currents $I_{1a} = I_{2a} = -300\,\text{nA}$
- Offset of the photo currents $I_{1o} = -300\,\text{nA}$ and $I_{2o} = -250\,\text{nA}$
- Phase delay between the interferometer channels $\Delta\Phi(t) = -10^\circ$ with $\Phi_1 = 10^\circ$
- Feedback resistor of the signal pre-conditioning model $R_f = 1\,\text{M}\Omega$

The amplitude, offset and phase delay of the interferometer channels are empirically determined from the realised interferometer. We use a feedback resistor of $1\,\text{M}\Omega$ to obtain reasonable amplitudes for further data processing. Figure 4.11 shows the result of a calibration scenario. The xy-plot displays the result of the ellipse fitting and the data after Heydemann correction.

4.4.2 Sensitivity Analysis of Noise Sources

To find the sensitivity coefficients of a noise source the following steps are carried out:

1. Without any source of interference (all noise sources turned off) we perform a calibration of the model to estimate the Heydemann parameters.
2. Keep the same model parameters as used for the calibration scenario but activate the noise source we want to analyse in both interferometer channels and perform a measurement.
3. Calculate the deviation between the real displacement and the measured displacement $e_d = d_r - d_m$.
4. Find the worst ratio between the standard uncertainty of e_d and the standard uncertainty of the activated noise source to obtain the sensitivity coefficient.

This procedure has to be repeated for each noise source. However, as an example we present the analysis of the input current noise I_{nA} of the OPA in more detail.
After running a calibration scenario we turn on the influencing quantity I_{nA} in both interferometer channels and simulate a measurement with constant velocity of the measurement mirror. Figure 4.12 depicts the deviation between the real and the measured displacement with the corresponding normalised histogram. We recognize that e_d shows a periodical behaviour. By activating the noise sources the phasor does not follow the estimated ellipse any more. Because the consequence of the Heydemann correction depends on the position (x, y) of the sample in the xy-plot, depicted in figure 4.11, and the estimated ellipse during calibration, we obtain this periodical behaviour. The period time relates to the velocity of the measurement mirror. Thus the influence of I_{nA} to the measurement result depends on the ellipse parameters and on the instantaneous phasor position determined by the phase angle Θ. Figure 4.13 illustrates the deviation e_d between the real and the measured displacement with respect to the phase angle Θ. Nevertheless in case of constant velocity of the measurement mirror, the amplitudes of e_d are normally distributed like the influencing quantity I_{nA} itself.
To obtain the sensitivity of the noise source we look out for the section with the worst ratio between the standard deviations of e_d and the standard deviation of I_{nA} during one revolution of the phasor.
This has been done for all sensitivity coefficients related to the noise sources of the detectors, the signal pre-conditioning unit and the ADC. The results are summarised in table 4.1. Note that, although the noise sources of the detectors and the input current noise of the OPA show the same transfer function $H_2(j\omega)$ (see equation 4.11), the sensitivity coefficients are slightly different because the estimated variance of e_d depends on the underlying distribution. Especially the quantum noise I_q and its dependency on the photo current I leads do a different distribution of e_d.

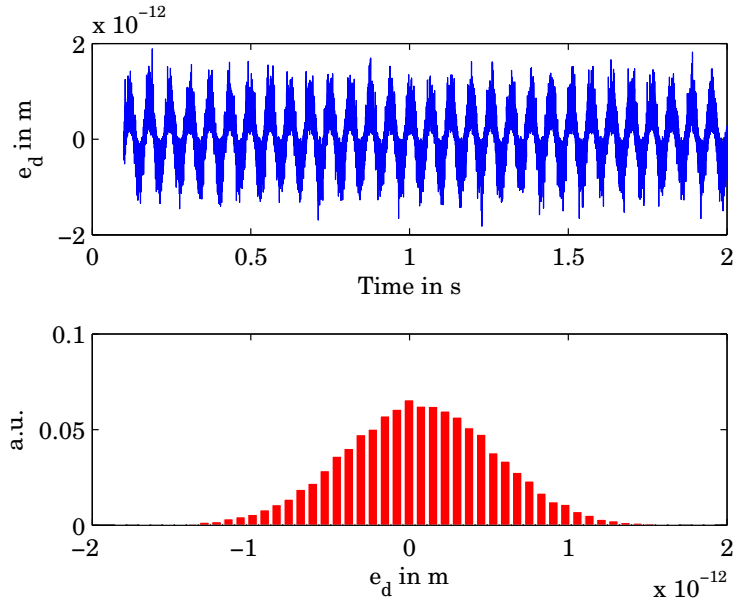


Figure 4.12: Deviation between real and measured displacement if influencing quantity I_{nA} is active. We recognise a periodic behaviour which arises from the Heydemann correction and relates to the velocity of the measurement mirror. Nevertheless, the amplitude of e_d is normally distributed.

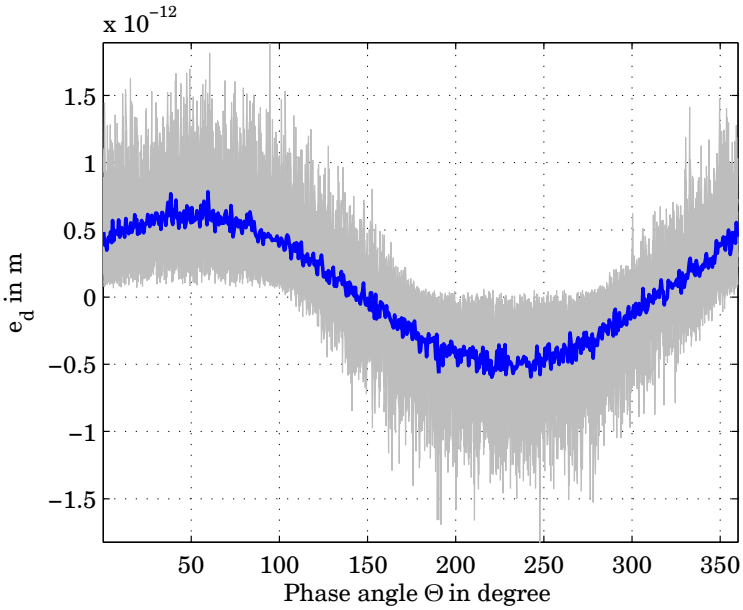


Figure 4.13: Deviation between real and measured displacement with respect to the current phase angle Θ with active influencing quantity I_{nA}. The thick line shows the mean value over 29 phasor revolutions.

CAUSE	SENSITIVITY COEFF.	UNIT
Thermal noise of detectors	$\frac{\partial e_d}{\partial I_t} = 3.67 \cdot 10^{-2}$	$\frac{\mathrm{m}}{\mathrm{A}}$
Shot noise of detectors	$\frac{\partial e_d}{\partial I_s} = 3.70 \cdot 10^{-2}$	$\frac{\mathrm{m}}{\mathrm{A}}$
Quantum noise of detectors	$\frac{\partial e_d}{\partial I_q} = 4.03 \cdot 10^{-2}$	$\frac{\mathrm{m}}{\mathrm{A}}$
Input current noise of OPA	$\frac{\partial e_d}{\partial I_{nA}} = 3.83 \cdot 10^{-2}$	$\frac{\mathrm{m}}{\mathrm{A}}$
Input voltage noise of OPA	$\frac{\partial e_d}{\partial V_{nA}} = 3.47 \cdot 10^{-14}$	$\frac{\mathrm{m}}{\mathrm{V}}$
Thermal noise of feedback resistors	$\frac{\partial e_d}{\partial V_{nR}} = 3.39 \cdot 10^{-14}$	$\frac{\mathrm{m}}{\mathrm{V}}$
Static errors of ADC	$\frac{\partial e_d}{\partial V_{SE}} = 3.33 \cdot 10^{-14}$	$\frac{\mathrm{m}}{\mathrm{V}}$
Quantisation noise of ADC	$\frac{\partial e_d}{\partial V_{QN}} = 6.31 \cdot 10^{-14}$	$\frac{\mathrm{m}}{\mathrm{V}}$

Table 4.1: Summary of the sensitivity coefficients related to the noise sources of the photo detectors, signal pre-conditioning unit and the ADC.

4.4.3 Sensitivity Analysis of Atmospheric Influences and Laser Wavelength

The estimated wavelength λ_e of the laser in ambient air depends on the vacuum wavelength of the laser λ_0 and the refractive index of air n. The latter is a function of the measured environmental parameters ambient temperature T_m, barometric pressure p_m, relative humidity RH_m, and CO_2 concentration. The refractive index of air is calculated with the modified Edlén equation presented by NIST[38], where the CO_2 concentration is assumed to be 450 μmol/mol by default.
To analyse the influence of the vacuum wavelength and the environmental parameters to the measurement result, the following steps are carried out, after performing a calibration of the model:

1. In measurement mode we introduce a constant deviation over time between the real wavelength λ_r and the estimated wavelength λ_e by changing one of the above mentioned parameters exclusively. E.g. introducing a temperature deviation of $\Delta T_1 = T_r - T_m$, between the real ambient temperature T_r and the measured ambient temperature T_m. Because the wavelength in ambient air is the scale for interferometric displacement measurement, the measured displacement d_m starts to deviate from the real displacement d_r. The resulting deviation $e_d = d_r - d_m$ is cumulative and increases with the displacement of the measurement mirror as depicted in figure 4.14. The parameter we are temporarily interested in, is the slope $\left.\frac{\partial e_d}{\partial d_r}\right|_{\Delta T_1}$ of the deviation e_d over the real displacement d_r.
2. We repeat the experiment described in step 1 with several different temperature

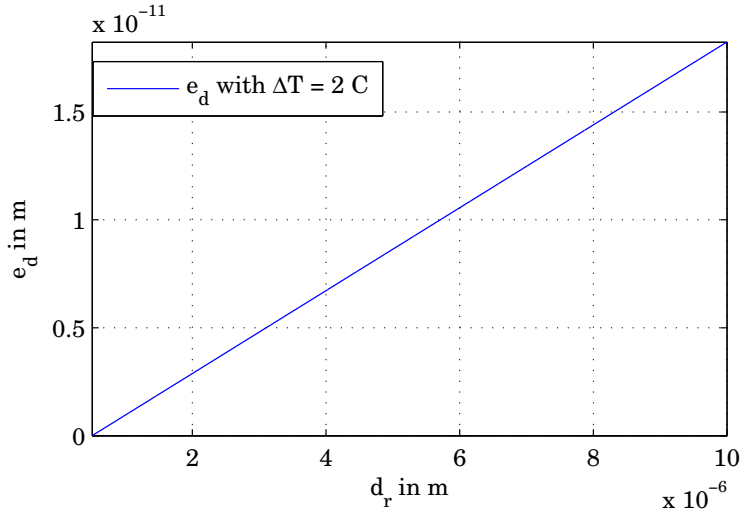


Figure 4.14: Deviation between real and specified measurement in case of $\Delta T = 2\,°\mathrm{C}$ and constant velocity of measurement mirror.

deviations $\Delta T_1, \Delta T_2, ..., \Delta T_N$ and keep the associated slopes $\left.\frac{\partial e_d}{\partial d_r}\right|_{\Delta T_{1,2,...,N}}$ in memory.

3. The final goal is to obtain a sensitivity coefficient depending on the real displacement of the mirror d_r and the parameter ΔT. Therefore we use the information from step 2 and develop a diagram as depicted in figure 4.15(a). It shows the dependency of the deviation between real and measured displacement per meter $\frac{\partial e_d}{\partial d_r}$ over the parameter ΔT.

This procedure has to be repeated for the barometric pressure, the relative humidity and the vacuum wavelength of the laser source. Figure 4.15 illustrates the sensitivity plots obtained with following simulation parameter:

- $T_r = 20\,°\mathrm{C}$
- $p_r = 101325\,\mathrm{Pa}$
- $RH_r = 50\,\%$
- $\lambda_{0r} = 632.9911599\,\mathrm{nm}$

The sensitivity coefficients are obtained by calculating the slopes of the sensitivity plots depicted in figure 4.15. The results are summarised in table 4.2 (cf. [57]).
The refractive index of air, calculated by the modified Edlén equation is only an approximation and has a certain uncertainty. This equation uncertainty can not be evaluated by simulation. In order to obtain a sensitivity coefficient we have to analyse the relationship

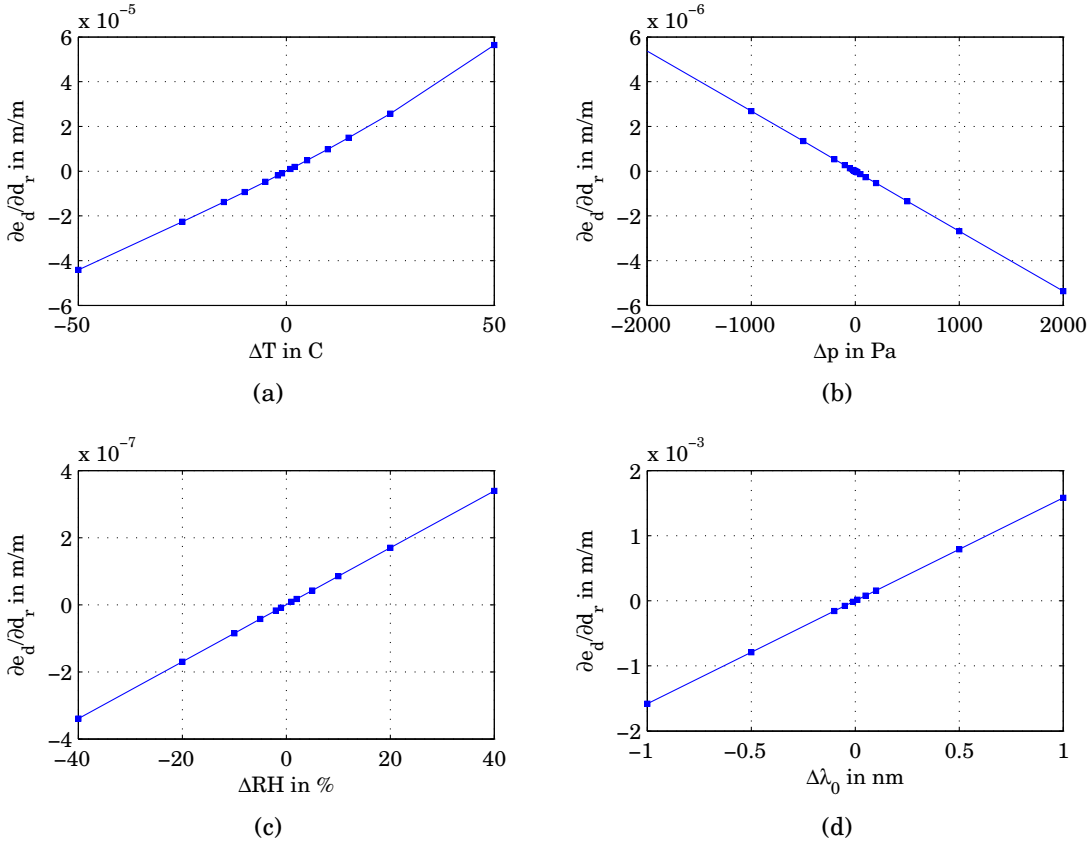


Figure 4.15: Sensitivity plots concerning the (a) ambient temperature ΔT, (b) barometric pressure Δp, (c) relative humidity ΔRH and (d) vacuum wavelength $\Delta\lambda_0$.

CAUSE	SENSITIVITY COEFF.	UNIT
Ambient temperature	$\frac{\partial^2 e_d}{\partial T \partial d_r} = 9.55 \cdot 10^{-7}$	$\frac{\text{m}}{\text{m}\cdot°\text{C}}$
Barometric pressure	$\frac{\partial^2 e_d}{\partial p \partial d_r} = -2.68 \cdot 10^{-9}$	$\frac{\text{m}}{\text{m}\cdot\text{Pa}}$
Humidity	$\frac{\partial^2 e_d}{\partial RH \partial d_r} = 8.49 \cdot 10^{-9}$	$\frac{\text{m}}{\text{m}\cdot\%}$
Vacuum wavelength of laser	$\frac{\partial^2 e_d}{\partial \lambda_0 \partial d_r} = 1.60 \cdot 10^{-3}$	$\frac{\text{m}}{\text{m}\cdot\text{nm}}$
Edlén equation	$\frac{\partial^2 e_d}{\partial n \partial d_r} = -1$	$\frac{\text{m}}{\text{m}}$

Table 4.2: Summary of the sensitivity coefficients related to the atmospheric model.

between the deviation e_d of real and measured displacement caused by Δn:

$$e_d = d_r - d_m = \frac{\lambda \cdot \Delta\Theta}{4\pi} = \frac{\lambda_0}{4\pi \cdot n} \cdot \left(\frac{4\pi \Delta n}{\lambda_0} \cdot d_r \right) = \frac{\Delta n}{n} \cdot d_r \tag{4.15}$$

Under the assumption that $\Delta n \approx 0$ and by taking the partial derivatives with respect to n and d_r from equation 4.15 we obtain:

$$\frac{\partial^2 e_d}{\partial n \partial d_r} = -\frac{1}{n^2} \approx -1 \tag{4.16}$$

Equation 4.16 determines the sensitivity coefficient of Edlén's equation with respect to the real displacement d_r.
Note that the contribution of the atmospheric model to the combined uncertainty depends on the displacement range of the measurement mirror. The influencing quantities affect the wavelength in ambient air and lead to a cumulative increase of the deviation between real and measured displacement.

4.4.4 Sensitivity Analysis of Input Current Changes

Once the measurement system is calibrated, the amplitude, offset and phase shift of the interferometer channels may not change. However in real applications the amplitude and the offset of the interferometer channels do not stay constant with respect to the displacement, due to non-perfect collimation of the laser beam, reflections and tilt of the measurement mirror [6]. To obtain the associated MU contribution we perform the following steps after calibration of the simulation model:

1. In measurement mode we introduce a specific deviation of one parameter in equations 4.2 exclusively, after calibration of the model. E.g. $\Delta I_{1a} = I_{1a,c} - I_{1a,m}$ where $I_{1a,c}$ is the amplitude of the photo current of interferometer channel 1 in calibration mode and $I_{1a,m}$ in measurement mode respectively. The resulting deviation e_d between real and measurement displacement depends on the instantaneous phase angle Θ as illustrated in figure 4.16.

2. Repeating the simulation in step 1 we obtain a certain characteristic of the deviation e_d for different introduced deviation ΔI_{1a} as illustrated in figure 4.17. We memorise the minimum, maximum and mean value of e_d to the corresponding deviation ΔI_{1a}.

3. The minimum, maximum and mean value of e_d of the simulations are used to develop the sensitivity plot as depicted in figure 4.18. From the sensitivity plot

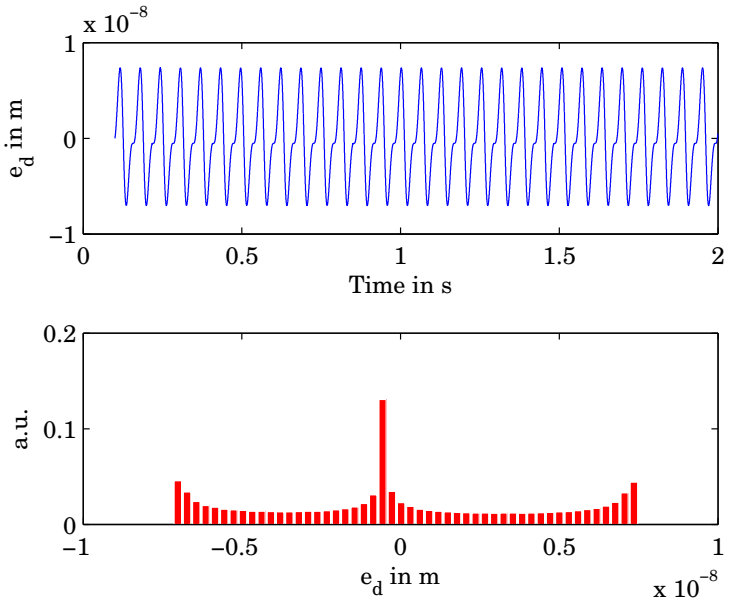


Figure 4.16: The first plot shows the deviation e_d between real and measured displacement in case of $\Delta I_{1a} = -30\,\text{nA}$ and constant velocity of the measurement mirror. We recognize a periodical and deterministic behaviour of e_d. The second plot is the corresponding normalised histogram of e_d.

> we derive the sensitivity coefficient by connecting the minima/maxima inside a specified region and find the line with the steepest slope in order to consider the worst e_d.

The following simulation results are obtained by using the interferometer channel parameters from section 4.4.1. In figure 4.16 and figure 4.17 we recognize a periodical and deterministic behaviour of the deviation e_d with respect to the instantaneous phase angle Θ. The period time relates to the velocity of the measurement mirror. The characteristic of the deviation e_d depends on the properties of the interferometer channels (amplitude, offset and phase delay) and the estimated ellipse parameters, which were used to calculate the Heydemann parameters. In case of introducing changes in the amplitude of a channel, e_d shows a different characteristic than in case of introducing changes in the offset of a channel. In general, if the set-up of the model changes, different results of the sensitivity coefficients can be expected.
However, with the model it can be shown that the deviations, based on different parameter modifications, are independent to each other. Hence, the linear combination of the resulting deviations e_d caused by exclusively introduced changes of amplitudes, offsets, and phase delays, lead to the same over all deviation as if we repeat the simulation by simultaneously change the amplitudes, offsets and phase delays of the interferometer

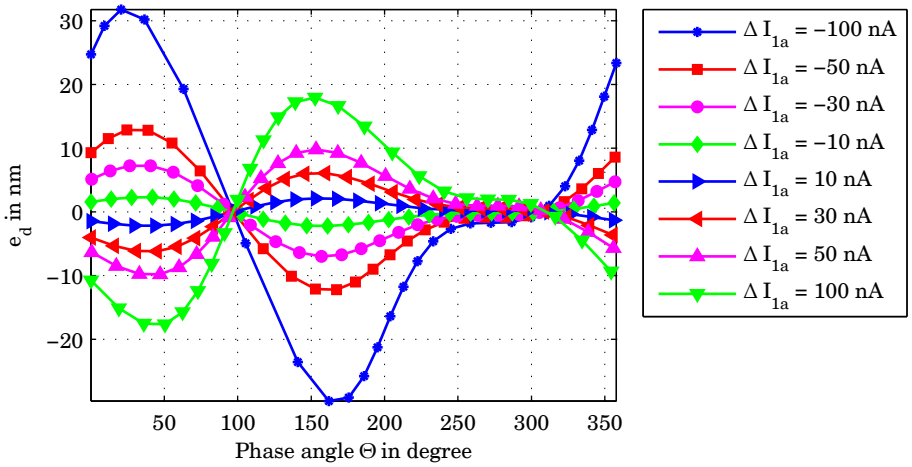


Figure 4.17: Deviation e_d between real and measured displacement with respect to the instantaneous phase angle Θ and different ΔI_{1a}.

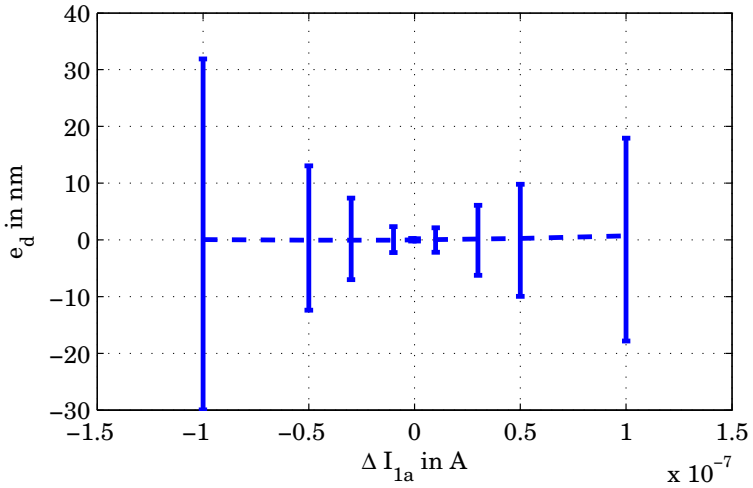


Figure 4.18: Error bar plot of deviation e_d between real and measured displacement with respect to different modification of ΔI_{1a} after calibration of the model. Thereby the dashed line shows the mean deviation of one phasor revolution and the error bars indicate the minimum and maximum deviations e_d during one phasor revolution in case of a specific ΔI_{1a}. The sensitivity coefficient is obtained by observing the range $\Delta I_{1a} = \pm 30\,\mathrm{nA}$ and finding the worst sensitivity coefficient.

CAUSE	SENSITIVITY COEFF.	UNIT
Channel 1 amplitude I_{1a}	$\frac{\partial e_d}{\partial I_{1a}} = -0.23$	$\frac{\mathrm{m}}{\mathrm{A}}$
Channel 1 offset I_{1o}	$\frac{\partial e_d}{\partial I_{1o}} = +0.23$	$\frac{\mathrm{m}}{\mathrm{A}}$
Channel 2 amplitude I_{2a}	$\frac{\partial e_d}{\partial I_{2a}} = -0.32$	$\frac{\mathrm{m}}{\mathrm{A}}$
Channel 2 offset I_{2o}	$\frac{\partial e_d}{\partial I_{2o}} = -0.32$	$\frac{\mathrm{m}}{\mathrm{A}}$
Phase shift $\Delta\Phi$	$\frac{\partial e_d}{\partial \Delta\Phi} = -7.61 \cdot 10^{-10}$	$\frac{\mathrm{m}}{^\circ}$

Table 4.3: Summary of the sensitivity coefficients related to the input currents of the detectors. The listed sensitivity coefficients are obtained with a feedback resistor R_f of $1\,\mathrm{M}\Omega$ and need to be recalculated in case of different resistor values.

channels.
The procedure above has to be repeated for the amplitude, offset and phase shift of each interferometer channel exclusively, to obtain the associated sensitivity coefficients. For brevity only the amplitude of the interferometer channel 1 is shown in detail. Other sensitivity coefficients are summarised in table 4.3. The biases of the resulting deviations e_d are neglected in the analysis of the MU. The observation range, to calculate the sensitivity coefficients of the photo current amplitudes and the photo current offsets, is specified with $\pm 30\,\mathrm{nA}$. To calculate the sensitivity coefficients associated to the phase difference $\Delta\Phi$ the range of deviation is specified with $\pm 5^\circ$.

4.4.5 Influence of RLS Fitting Algorithms

In our developed model a direct least square approach [13] to estimate the ellipse parameters as mentioned in section 4.3.2 is used. On the DSP an RLS approach is used instead. The result of the direct least square approach is considered as a reference standard to evaluate the RLS introduced fitting error. For this reason we develop a simple model depicted in figure 4.19 to calculate the difference between the Heydemann parameters obtained by the two different approaches. Therefore we run the simulation with 1000 realisations of the interferometer channel signals, to estimate the mean vector μ_{RLS} and the covariance matrix Σ_{RLS} of the error between the calculated Heydemann parameters. We repeat this procedure for different ratios of the input amplitudes I_{2a}/I_{1a} and different phase relationships $\Delta\Phi$ between both interferometer channels.
Figure 4.20(a) and 4.20(b) illustrated the results obtained by simulation. We recognize that there are significantly higher errors between the two approaches in the region, where $I_{2a}/I_{1a} > 1$ and the phase relationship $\Delta\Phi = \Phi_2 - \Phi_1$ is less than or equal to zero. The reason for higher errors in this region arises from a worse initial position of the estimators.

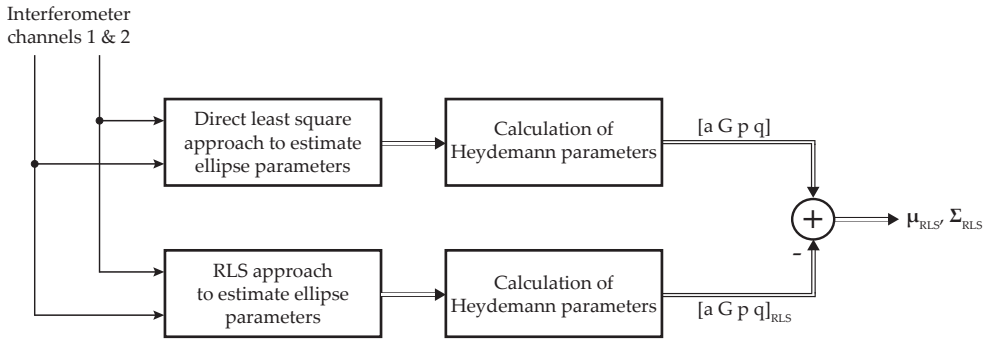


Figure 4.19: Simulation model used to estimate the mean vector and the covariance matrix of the error between the calculated Heydemann parameters obtained by using two different ellipse estimation algorithms. The obtained results are used to specify the properties of the multivariate normally distributed noise source depicted in figure 4.9.

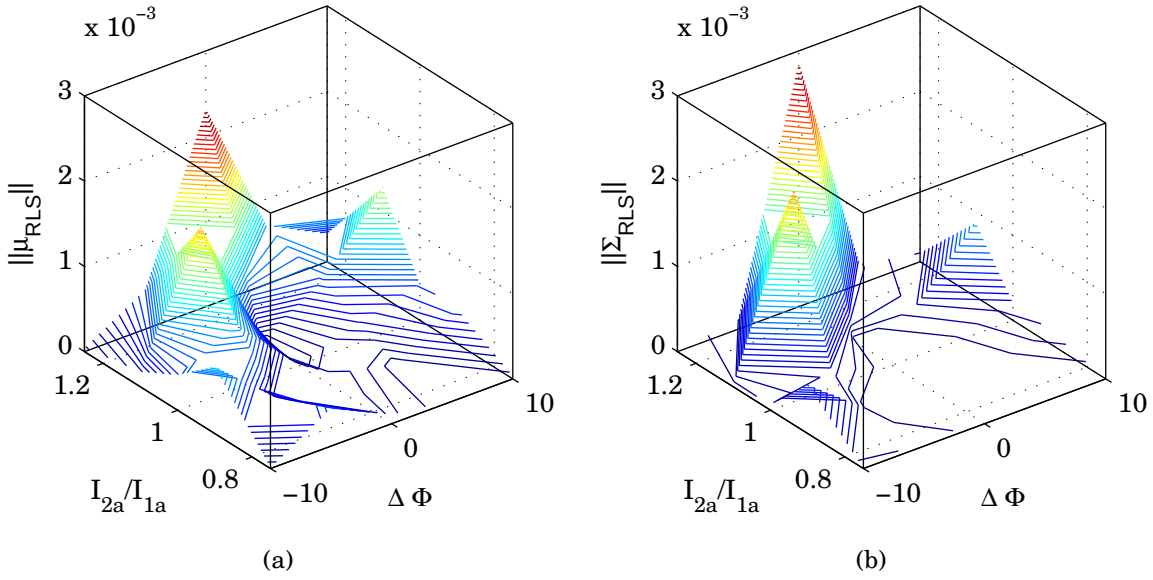


Figure 4.20: Error between the Heydemann parameters obtained with the direct least square approach and the RLS algorithm. Different amplitude ratios I_{2a}/I_{1a} and different phase relationships $\Delta\Phi$ between both interferometer channels are used to see how this will influence the results of the RLS algorithm. (a) shows the norm of the estimated mean values and (b) the norm of the estimated covariance matrix.

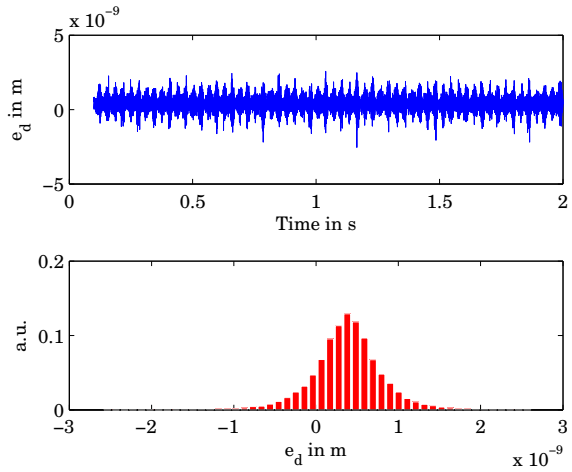


Figure 4.21: Deviation e_d between real and measured displacement and the corresponding normalised histogram with simulated RLS algorithm error. We obtain this result under the assumptions that $I_{2a}/I_{1a} = 1$ and $\Delta\Phi = -10°$. The mean value of the e_d is given with 0.28 nm and the standard uncertainty $u_{RLS} = 0.31$ nm.

Note further that the Fitzgibbon approach shows an eccentricity bias.
Under the assumption that $I_{2a}/I_{1a} = 1$ and $\Delta\Phi = -10°$ we use the estimated mean vector μ_{RLS} and covariance matrix Σ_{RLS} of the error to specify the properties of the multivariate normally distributed noise source depicted in figure 4.9.
Now that we know the properties of the multivariate normally distributed noise source, which simulates the error between the Heydemann parameters introduced by the RLS algorithm, we will again use the simulation model described in section 4.3, perform a calibration of the model without any source of interference (all noise sources turned off) and simulate a measurement scenario. During simulation of a measurement we turn on the multivariate normally distributed noise source depicted in 4.9 and observe the deviation e_d between the real and the measured displacement. Figure 4.21 depicts the deviation e_d and the corresponding normalised histogram.
The simulation leads to the result that, under the assumption $I_{2a}/I_{1a} = 1$ and $\Delta\Phi = -10°$, the mean value of the deviation e_d is given with 0.28 nm and the standard uncertainty $u_{RLS} = 0.31$ nm. In order to incorporate the RLS algorithm introduced uncertainty to the MU budget we use the obtained standard uncertainty u_{RLS}. The bias is neglected in the analysis of the MU.

4.4.6 Summary

In general the sensitivity coefficients obtained in table 4.1 and 4.3 depend on the set-up of the model. Especially the feedback resistor R_f has an important effect because the calculated sensitivity coefficients are related to the photo currents I_1 and I_2 of the detectors and not to the input voltages of the ADC. Hence, if the amplification of the realised measurement system (determined by R_f) differs significantly to the one used in the simulation, the sensitivity coefficients concerning the photo currents have to be recalculated. However, this task can be easily accomplished, due to the linear dependency of the input currents to the output voltages of the signal pre-conditioning model.
Further, different results can be expected if the properties of the photo currents (amplitude, offset and phase delay), generated by equations 4.2, differs significantly from the parameters used in the simulation. Consequently this evaluation reflects the sensitivity coefficients to a certain set-up. Nevertheless, the developed model can be used to do further simulations with other interferometer specific parameters to evaluate sensitivity coefficients.

4.5 Evaluation of the Input Quantities

In this section we evaluate all input quantities by their standard uncertainties u_{xi} as described in the GUM. These standard uncertainties lead in combination with the previously determined sensitivity coefficients, to the contribution of each input quantity to the combined uncertainty u_c.

4.5.1 Input Quantities related to the Detector and Signal Pre-Conditioning Unit

Noise sources:
We specify following parameters and use additional information given in the data sheet of the OPA [55] and the data sheet of the photo diode [20] to calculate the standard uncertainties of the noise sources related to the detectors and the signal pre-conditioning unit. The specifications are:

- Bandwidth $B = f_h = 40\,\text{kHz}$ and $f_l = 0.1\,\text{Hz}$
- Absolute temperature $T_{abs} = 293.15\,\text{K}$

CAUSE	TYPE	DISTR.	STD. UNCERTAINTY	UNIT
Thermal noise of detectors	B	Normal	$u_{I_t} = 2.55 \cdot 10^{-12}$	A
Shot noise of detectors	B	Normal	$u_{I_s} = 0$	A
Quantum noise of detectors	B	Normal	$u_{I_q} = 8.77 \cdot 10^{-11}$	A
Input current noise of OPA	B	Normal	$u_{I_{nA}} = 1.76 \cdot 10^{-13}$	A
Input voltage noise of OPA	B	Normal	$u_{V_{nA}} = 1.12 \cdot 10^{-6}$	V
Thermal noise of R_f	B	Normal	$u_{V_{nR}} = 2.54 \cdot 10^{-5}$	V

Table 4.4: Summary of the calculated standard uncertainties related to the noise sources of the detectors and the signal pre-conditioning unit.

- Maximum photo current of the detector $I_{max} = 600 \cdot 10^{-9}\,\mathrm{A}$
- Feedback resistor of C/V circuitry $R_f = 1\,\mathrm{M\Omega}$
- Dark current $i_d = 0\,\mathrm{A}$ (no bias voltage)
- Junction resistance of photo diode $R_j = 100\,\mathrm{M\Omega}$ (assumed)

We use the equations determined in section 4.3.2 to calculate the standard uncertainties of the input quantities listed in table 4.4. To calculate the standard uncertainty of the input current and input voltage noise we use equations 4.4 and 4.5 respectively. The required information can be found in the data sheet of the OPA.

Photo detector currents:
The standard uncertainties of the photo detector amplitude and offset variations of both interferometer channels are obtained by repeated observation of the signals on a measurement distance of 20 mm. The results summarised in table 4.5 are obtained with a feedback resistor value of 1 MΩ and published by Auer [6]. The phase delay $\Delta\Phi$ between the interferometer channels is constant. Thus the standard uncertainty of this parameter is zero.
The uncertainty contribution of the detector unit and the signal pre-conditioning unit is calculated with equations 4.17 and 4.18 respectively. The associated sensitivity coefficients are listed in table 4.1 and 4.3. In equation 4.17, u_{phc} denotes the standard uncertainty caused by variations of the photo currents of both interferometer channels and u_{dtn} denotes the standard uncertainties caused by the noise of the photo detectors. We recognise that u_{phc} constitutes the main part of the standard uncertainty of the detector unit u_{DET}.

CAUSE	TYPE	DISTR.	STD. UNCERTAINTY	UNIT
Amplitude of channel 1	A	Normal	$u_{I_{1a}} = 10 \cdot 10^{-9}$	A
Amplitude of channel 2	A	Normal	$u_{I_{2a}} = 10 \cdot 10^{-9}$	A
Offset of channel 1	A	Normal	$u_{I_{1o}} = 5 \cdot 10^{-9}$	A
Offset of channel 2	A	Normal	$u_{I_{2o}} = 5 \cdot 10^{-9}$	A
Phase between channels	-	-	$u_{\Delta\Phi} = 0$	°

Table 4.5: Summary of the estimated standard uncertainties related to the photo currents of the detectors by observing the signal variation on a measurement distance of 20 mm (from Auer [6]).

$$u_{phc} = \sqrt{\left(\frac{\partial e_d}{\partial I_{1a}} \cdot u_{I_{1a}}\right)^2 + \left(\frac{\partial e_d}{\partial I_{1o}} \cdot u_{I_{1o}}\right)^2 + \left(\frac{\partial e_d}{\partial I_{2a}} \cdot u_{I_{2a}}\right)^2 + \left(\frac{\partial e_d}{\partial I_{2o}} \cdot u_{I_{2o}}\right)^2 + \left(\frac{\partial e_d}{\partial \Delta\Phi} \cdot u_{\Delta\Phi}\right)^2}$$

$$u_{dtn} = \sqrt{\left(\frac{\partial e_d}{\partial I_t} \cdot u_{I_t}\right)^2 + \left(\frac{\partial e_d}{\partial I_s} \cdot u_{I_s}\right)^2 + \left(\frac{\partial e_d}{\partial I_q} \cdot u_{I_q}\right)^2}$$

$$u_{DET} = \sqrt{u_{phc}^2 + u_{dtn}^2} = \sqrt{(3.96\,\text{nm})^2 + \left(3{,}56 \cdot 10^{-3}\,\text{nm}\right)^2} = \underline{\underline{3.96\,\text{nm}}} \tag{4.17}$$

The standard uncertainty of the signal pre-conditioning unit u_{SC} is calculated using equation 4.18.

$$u_{SC} = \sqrt{\left(\frac{\partial e_d}{\partial I_{nA}} \cdot u_{I_{nA}}\right)^2 + \left(\frac{\partial e_d}{\partial V_{nA}} \cdot u_{V_{nA}}\right)^2 + \left(\frac{\partial e_d}{\partial V_{nR}} \cdot u_{V_{nR}}\right)^2} = \underline{\underline{6.74 \cdot 10^{-6}\,\text{nm}}} \tag{4.18}$$

4.5.2 Input Quantities related to the Signal Processing

To calculate the standard uncertainties related to the ADC we use the information given in the ADC data sheet [42] and the equations 4.13 and 4.14. The standard uncertainty of the RLS algorithm is obtained by simulation in section 4.4.5. Table 4.6 summarises the results. The standard uncertainty related to the signal processing is calculated with equation 4.19. The associated sensitivity coefficients are listed in table 4.1.

$$u_{SP} = \sqrt{\left(\frac{\partial e_d}{\partial V_{SE}} \cdot u_{se}\right)^2 + \left(\frac{\partial e_d}{\partial V_{QN}} \cdot u_{qn}\right)^2 + u_{RLS}^2} = \underline{\underline{0.31\,\text{nm}}} \tag{4.19}$$

CAUSE	TYPE	DISTR.	STD. UNCERTAINTY	UNIT
Gain error	B	Uniform	$u_g = 5.6 \cdot 10^{-3}$	V
Offset error ($k_p = 2$)	B	Normal	$u_o = 4.9 \cdot 10^{-3}$	V
Integral non-linearity	B	Uniform	$u_{inl} = 5.07 \cdot 10^{-4}$	V
Static errors	B	Uniform	$u_{se} = 12.3 \cdot 10^{-3}$	V
Quantization error	B	Uniform	$u_{qn} = 1.41 \cdot 10^{-4}$	V
RLS error	A	Normal	$u_{RLS} = 0.31$	nm

Table 4.6: Summary of the calculated standard uncertainties related to signal processing.

4.5.3 Input Quantities related to the Atmospheric Model

To evaluate the measurement uncertainties associated to the ambient temperature, barometric pressure, and relative humidity we need to know which sensors are used. Currently, no sensors are attached directly to the SPU. Hence, we have to assume the uncertainties of the measured values. As reference values we use the data sheet information of the suggested pressure sensor [9] and temperature/humidity sensor [31].
The used stabilised He-Ne-laser SL 03 [33] emits, after a starting time of 30 minutes, coherent light with a vacuum wavelength of $\lambda_0 = 632.9911599\,\text{nm}$. The relative long term stability of the wavelength is given in the test certificate [34] with $\pm 2 \cdot 10^{-8}$. Under the assumption that the underlying probability is uniform, we calculate the standard uncertainty of the vacuum wavelength λ_0 with $u_{\lambda_0} = \lambda_0 \cdot \frac{2 \cdot 10^{-8}}{\sqrt{3}} = 7.31 \cdot 10^{-6}\,\text{nm}$.
The calculation of the refractive index of air is based on the modified Edlén equation presented by NIST [38]. The NIST states that the expanded standard uncertainty of the refractive index of air due to model errors is $2 \cdot 10^{-8}$ ($k_p = 2$). This leads to a standard uncertainty of $u_{Edlén} = 1 \cdot 10^{-8}$.
Table 4.7 summarises the described standard uncertainties related to the atmospheric model. The standard uncertainty of the atmospheric model depends on the measurement distance d_r and is calculated according to equation 4.20. The measurement distance is specified with $d_r = 20\,\text{mm}$. The corresponding sensitivity coefficients are listed in table 4.2.

$$u_{AM} = d_r \cdot \sqrt{\left(\frac{\partial e_d}{\partial T \partial d_r} \cdot u_T\right)^2 + \left(\frac{\partial e_d}{\partial p \partial d_r} \cdot u_p\right)^2 + \left(\frac{\partial e_d}{\partial RH \partial d_r} \cdot u_{RH}\right)^2 + \left(\frac{\partial e_d}{\partial n \partial d_r} \cdot u_{Edlén}\right)^2} \quad (4.20)$$
$$= \underline{\underline{8.38\,\text{nm}}}$$

Cause	Type	Distr.	Std. Uncertainty	Unit
Ambient temperature	B	Uniform	$u_T = 0.17$	°C
Barometric pressure	B	Uniform	$u_p = 144$	Pa
Humidity	B	Uniform	$u_{RH} = 1.04$	%
Vacuum wavelength	B	Uniform	$u_{\lambda_0} = 7.31 \cdot 10^{-6}$	nm
Edlén equation ($k_p = 2$)	B	Normal	$u_{Edlén} = 1 \cdot 10^{-8}$	a.u.

Table 4.7: Summary of the standard uncertainties related to the atmospheric model.

4.6 Measurement Uncertainty Budget

The previous MU analysis of the functional components of the measurement system, depicted in figure 4.1, is summarized in the table 4.8. Figure 4.22 illustrates the contribution of each functional components graphically.

u_x	Component	MU Contribution
u_{DET}	Detector unit	3.96 nm
u_{SC}	Signal pre-conditioning	$6.74 \cdot 10^{-6}$ nm
u_{SP}	Signal processing	0.31 nm
u_{AM}	Atmospheric model	8.38 nm

Table 4.8: MU contribution of the functional components of the measurement system.

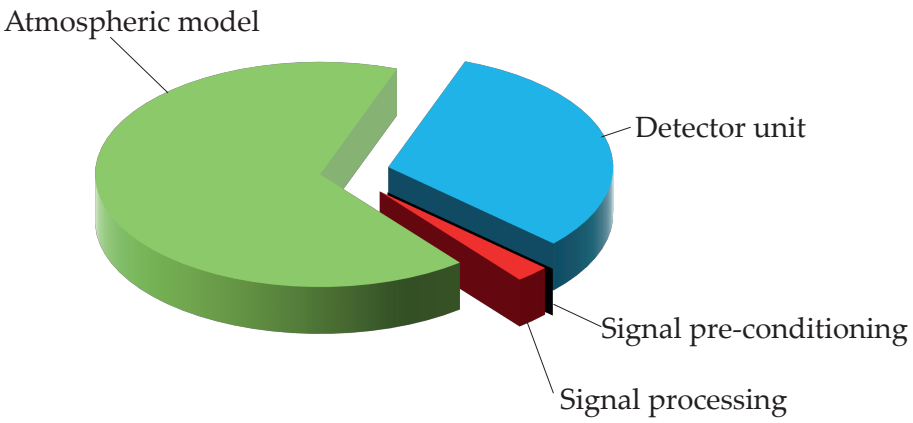


Figure 4.22: Pie chart of MU-budget for a measurement distance of 20 mm.

Conclusion

The combined uncertainty of the measurement system, without taking the mechanical assembly and the geometrical alignment of the measurement system components into account, is calculated with equation 4.21.

$$u_c = \sqrt{u_{DET}^2 + u_{SC}^2 + u_{SP}^2 + u_{AM}^2} = \underline{\underline{9.27\,\text{nm}}} \tag{4.21}$$

The influence of the signal pre-conditioning has a negligible small influence to the MU of the measurement system. Furthermore the contribution of the signal processing component is very small compared to those of the detector unit and the atmospheric model. The contribution of the detector unit mainly depends on the variation of the amplitudes and offsets of the interferometer channels. The contribution of the atmospheric model depends on the measurement distance and grows with increasing measurement distances. Note that correlations of the influencing quantities are not taken into account in this MU analysis.

5 Experiments and Findings

This chapter involves the observation of the required time to process the interferometer signals on the TMS320C6713 and the experimental validation of the measurement results. Therefore the measurement results of the SPU are compared with an off-line data processing method developed by Auer [6]. Finally some suggestions to improve the performance of the SPU are presented.

5.1 Data Processing Analysis on the DSP

As mentioned in section 3.3, the TMS320C6713 DSP has to process the successive data blocks fast enough to avoid loss of ADC samples. The required processing time of one data block depends on the following quantities:

- Size of ping/pong buffer (3072 data samples)
- Processing mode of the SPU
- Down sample factor M used in the processing mode (reduces the effective data rate)
- Source code compilation in CCS (accomplished without code optimisation)

The ping/pong buffer size of 3072 samples and the sample rate of $f_s = 151.210\,\text{kHz}$ leads, according to equation 3.23, to a time interval of 6.77 ms between successive data blocks. Thus the CPU is interrupted by the EDMA controller each 6.77 ms, to read and process the ping or pong buffer entry. To avoid data loss, the data processing of the ADC data needs to be finished before the EDMA controller generates the next CPU interrupt. The different processing modes of the SPU require totally different processing times for one data block, if equal down sample factors are used. Thus the down sample factor of each processing mode is adapted, such that we do not exceed the theoretical processing time limit of 6.77 ms.
A tool provided by the IDE is used to determine the required processing times. This tool

Processing Mode	Max. Req. Time	M	Max. Velocity
1: Gain and offset calibration	2.88 ms	1	$3.16\,\frac{\text{mm}}{\text{s}}$
2: Ellipse estimation	4.83 ms	10	$2.39\,\frac{\text{mm}}{\text{s}}$
3: Displacement measurement	5.14 ms	2	$3.16\,\frac{\text{mm}}{\text{s}}$

Table 5.1: Maximum required processing time for one data block (3072 samples) in different processing modes of the SPU under consideration of the down sample factor M.

Algorithm	Max. Required Time
Input data sorting and data type conversion	0.68 ms
FIR filtering	0.62 ms
Heydemann correction and phase angle estimation	0.57 ms
Calculation of refractive index of air	0.02 ms
Unwrap phase and apply atmospheric model	2.74 ms
Trigger detection	0.39 ms

Table 5.2: Essential algorithms in the displacement measurement mode and the maximum required processing times for one data block (3072 samples, M = 2). The division of the algorithms results from the software implementation.

measures the minimum, maximum and average processing time of the implemented algorithms. Table 5.1 summarises the maximum required processing time of each processing mode under consideration of the down sample factor. Note, that the maximum allowed measurement mirror velocity is determined by the cut off frequency of the discrete AAF. Because the maximum down sample factor of 7.56 according to equation 3.19 is exceeded in processing mode 2, the maximum allowed input frequency is reduced from 10 kHz to 7.56 kHz in order to avoid aliasing. Consequently, the measurement mirror velocity in processing mode 2 must not exceed $2.39\,\frac{\text{mm}}{\text{s}}$ according to equation 3.15.
Table 5.2 breaks down the SPU's displacement measurement mode into its essential subroutines. We recognise, that the function which unwraps the phase and applies the atmospheric model to translate the phase angle Θ to a displacement d, demands the most time.

5.2 Validation of the Measurement Results

To validate the measurement results of the SPU, several comparison measurements are performed with the set-up depicted in figure 5.1. It consists of the HMI developed by Auer [6], a translation stage with the associated control device to move the measurement

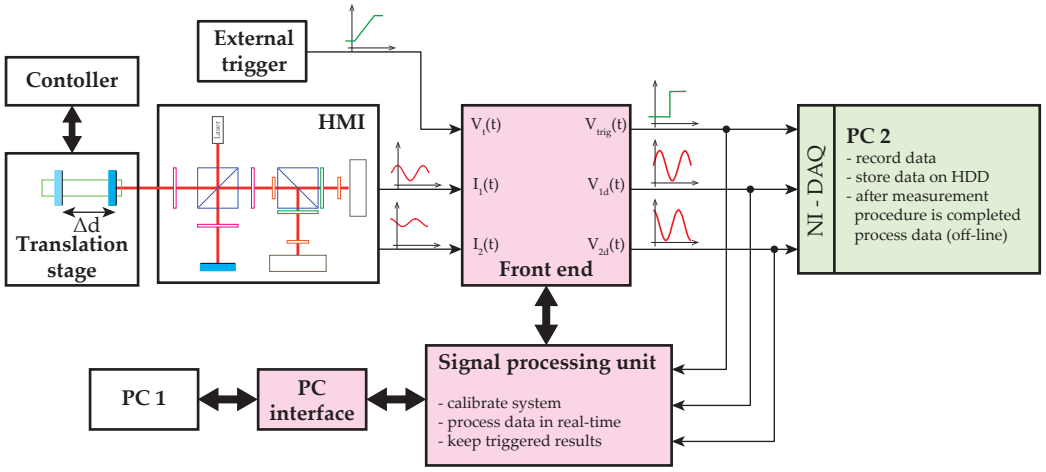


Figure 5.1: Measurement set-up to validate the developed measurement algorithms implemented on the TMS320C6713 DSP.

mirror, an external trigger device, the developed SPU with all associated modules to measure displacement in real-time, and two PCs. The external trigger device is used to synchronise the measurements between the SPU and the PC 2. PC 2 uses a data acquisition card from National Instruments to record the interferometer channels and to store the data on the Hard Disk Drive (HDD).
The pre-conditioned interferometer signals V_{1d}, V_{2d} and V_{trig} from the front end's outputs are fed into the SPU and into PC 2. Additionally, we connect the SPU to PC 1 to access the triggered measurement results with the developed console application.
The measurement procedure itself consists of a calibration step and a measurement step. In the calibration step the SPU adjusts the amplitude and the offset of the front end's output signals and estimates the ellipse to calculate the Heydemann parameters afterwards. When the system is calibrated, we start the measurement data acquisition on PC 2. Now the measurement mirror is moved with the translation stage several times. The synchronisation of the measurement happens at the time, where the measurement mirror stands still to minimise timing errors. After the measurement procedure is completed, the recorded data is processed on PC 2 with software developed in Lab-View and MATLAB by Auer. Thereby Auer uses the stored data set to estimate the ellipse parameters, to calculate the Heydemann parameters, and to estimate the displacement at the trigger events. Finally the measurement results from the SPU and from the PC 2 are compared. The previously described measurement procedure is repeated several times with different measurement mirror velocities to test the SPU. However, table 5.3 summarises the obtained measurement results of such a measurement procedure. It shows that the maximum deviation between the measurement results is about 1 nm.

TRIGGER	$\Delta d_{real-time}$	$\Delta d_{off-line}$
1	$0.000\,\mu m$	$0.0000\,\mu m$
2	$-19.640\,\mu m$	$-19.6395\,\mu m$
3	$-40.586\,\mu m$	$-40.5854\,\mu m$
4	$-25.231\,\mu m$	$-25.2316\,\mu m$
5	$-21.101\,\mu m$	$-21.1018\,\mu m$

Table 5.3: Comparison between the measurement results obtained by the developed SPU with real-time capabilities and by the off-line data processing software from Auer.

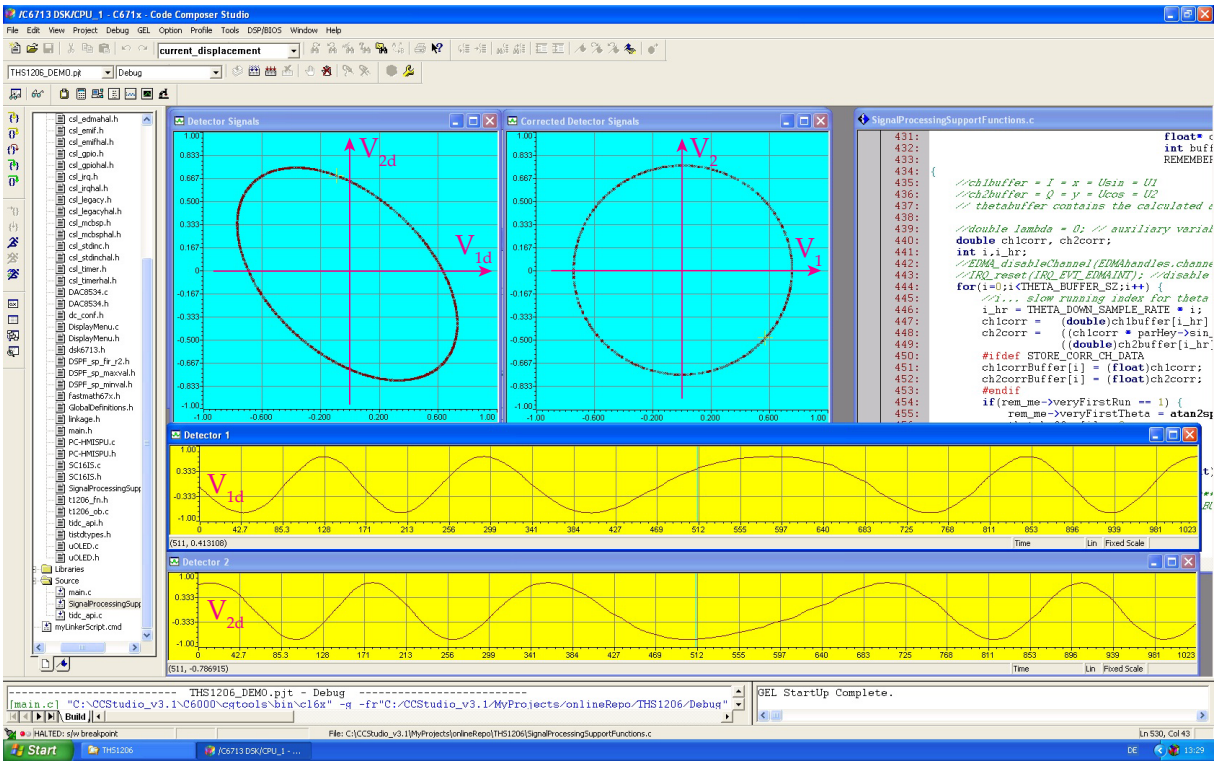


Figure 5.2: Screen shot of the IDE shows the detector signals V_{1d} and V_{2d} in the lower part of the image. The upper part illustrates the effect of the Heydemann correction. The left diagram shows the signal constellation of the detector signals V_{1d} and V_{2d} and the right diagram shows the signal constellation of the Heydemann corrected detector signals V_1 and V_2.

Figure 5.2 shows a screen shot of the IDE with different visualisations of a data block from the ADC. The lower part of the figure shows diagrams of the detector signals V_{1d} and V_{2d} over time. The projection of this detector signals leads to a signal constellation in shape of an ellipse, as depicted on the left hand side of the figure. The result of the subsequent Heydemann correction is depicted in the middle. The projection of the corrected detector signals V_1 and V_2 results in a circle.

5.3 Suggestions to Increase the Data Processing Rate

Although system architecture optimised functions provided by Texas Instruments have been used for algorithm implementation, the required data processing rate to handle measurement mirror velocities of $100\frac{\text{mm}}{\text{s}}$ can not be achieved.
To improve the performance of the SPU we could increase the down sample factor by simultaneously increasing the sample rate of the ADC. Note that this would require also an adequate adaptation of the analog AAF and of the discrete FIR filter. This iterative procedure finds the setting, which leads to the maximum performance of the existing hardware.
Another possibility to improve the performance is to use a delta-sigma ADC with a high input sampling rate and an internal digital filter. This leads to a reduced output sample rate of the ADC and makes the digital FIR filtering on the DSP unnecessary.
However, the required processing rate to handle measurement mirror velocities of $100\frac{\text{mm}}{\text{s}}$ will not be reached with the existing TMS320C6713 DSP. Beside the data processing algorithms itself, especially the user interactions via the slow touch display interface cost valuable CPU resources. To improve the performance of the measurement system it would be an advantage to decouple the user applications and visualisation on the display from the DSP, which processes the measurement data. Therefore, we could use a multi-core DSP, where one dedicated CPU is used for user applications while the other CPUs process the measurement data. Alternatively, we could use a heterogeneous system-on-chip architecture, such as the OMAP-L138 from Texas Instruments. The OMAP-L138 consists of a DSP and an ARM processor. While the DSP is responsible for the signal processing, the ARM is used as general purpose processor and handles all user application related tasks.

6 Conclusion

This thesis covers the design, development and realisation of a stand-alone SPU, which processes signals from an HMI in real-time, to measure displacements in sub-nanometer resolution. The basics of interferometric displacement measurements are explained to understand the relationship between the displacement of the measurement mirror and the output signals of the HMI. Problems arising from the quadrature detector system of the HMI are discussed and necessary correction methods are presented. Based on this discussion an SPU is developed and realised. The required processing algorithms are implemented on a TMS320C6713 DSP from Texas Instruments. The measurement results obtained from the developed SPU are validated by comparison measurements. The performance of the implemented processing algorithms on the TMS320C6713 DSP is evaluated and suggestions to increase the data processing rate are discussed. The performance of the DSP limits the maximum velocity of the measurement mirror to $3.16\,\frac{\mathrm{mm}}{\mathrm{s}}$.
A measurement model is developed to analyse the measurement uncertainty by means of simulations. Relevant influencing quantities are explained and specified. The contribution of each influencing quantity to the measurement uncertainty are determined by simulation studies and summarised in an MU budget. The resulting combined uncertainty of the measurement system, under consideration of a measurement distance of 20 mm, is 9.27 nm. With a coverage factor of 2, which corresponds to a confidence level of approximately 95 % in case of a Gaussian PDF, this result leads to an expanded measurement uncertainty of 18.54 nm.

Outlook

To improve the user friendliness of the developed SPU it needs to be installed in an appropriate housing with included power supply for all hardware modules. Furthermore, sensors to measure the ambient temperature, the barometric pressure, and the humidity have to be connected to the SPU's I^2C interface and the dedicated communication protocols have to be implemented in the software. Currently the IDE is necessary to run the developed software on the SPU. To use the SPU without the IDE, the developed software

has to be programmed into the flash memory located on the TMS320C6713 DSK. Concerning the analysis of the MU, correlations of the influencing quantities has to be analysed and taken into account in the MU budget.

Appendix

A.1 Direct Least Square Ellipse Fitting

Fitzgibbon *et al.* [13] proposed a direct least square ellipse fitting approach with high computational efficiency, high robustness to noise and high ellipse specificity. The approach minimizes the sum of squared algebraic distances from the scattered data to the ellipse under the constraint $4ac - b^2 = 1$ by solving a generalized eigensystem.
A general conic is represented by an implicit second order polynomial:

$$F(\mathbf{a};\mathbf{x}) = \mathbf{a}\cdot\mathbf{x} = ax^2 + bxy + cy^2 + dx + ey + f = 0 \tag{A.1}$$

where $\mathbf{a} = [a\ b\ c\ d\ e\ f]^T$ and $\mathbf{x} = \left[x^2\ xy\ y^2\ x\ y\ 1\right]^T$. $F(\mathbf{x_i})$ is the algebraic distance of a point (x,y) to the conic $F(\mathbf{a};\mathbf{x}) = 0$. The fitting of a general conic may be approached by the following cost function which minimizes the quadratic sum of squared algebraic distances:

$$J(\mathbf{a}) = \sum_{i=1}^{N} F(\mathbf{x_i})^2 \tag{A.2}$$

The constraint $4ac - b^2 = 1$ to the parameter vector $\mathbf{a}$ makes the fitting ellipse specific. Fitzgibbon introduces a design matrix $\mathbf{D} = [\mathbf{x_1}\ \mathbf{x_2}\ ...\ \mathbf{x_n}]^T$ and expresses the constraint in matrix form $\mathbf{C}$. Now the problem can be stated as minimizing $J(\mathbf{a}) = \|\mathbf{Da}\|^2$ subject to the constraint $\mathbf{a}^T\mathbf{Da} = 1$. By introduction of the Lagrange multiplier λ and differentiation of the cost function with respect to the parameter vector $\mathbf{a}$ we obtain the system:

$$\begin{aligned} &2\mathbf{D}^T\mathbf{D} - 2\lambda\mathbf{Ca} = 0 \\ &\mathbf{a}^T\mathbf{Ca} = 1 \end{aligned} \tag{A.3}$$

These equations are expressed by:

$$\mathbf{Sa} = \lambda\mathbf{Ca} \tag{A.4}$$

$$\mathbf{a}^T\mathbf{Ca} = 1 \tag{A.5}$$

where $\mathbf{S}$ is the scatter matrix $\mathbf{D}^T\mathbf{D}$. This can now be solved considering the generalized eigenvectors of equation A.5. It has been shown that this leads to exactly one solution by finding the only negative eigenvalue and its associated eigenvector (see Matlab code A.1).

```
% Build design matrix
D = [x.*x x.*y y.*y x y ones(size(x))];
% Build scatter matrix
S = D'*D;
% Build 6x6 constraint matrix
C(6,6) = 0; C(1,3)=-2; C(2,2)=1; C(3,1)=-2;
% Solve generalized eigensystem
[gevec, geval] = eig(S,C);
% Find the only negative eigenvalue
[~,negC] = find(geval<0 & ~isinf(geval));
% Get fitted parameters
a = gevec(:,negC);
```

Matlab code A.1: Direct least square ellipse fitting approach from [13].

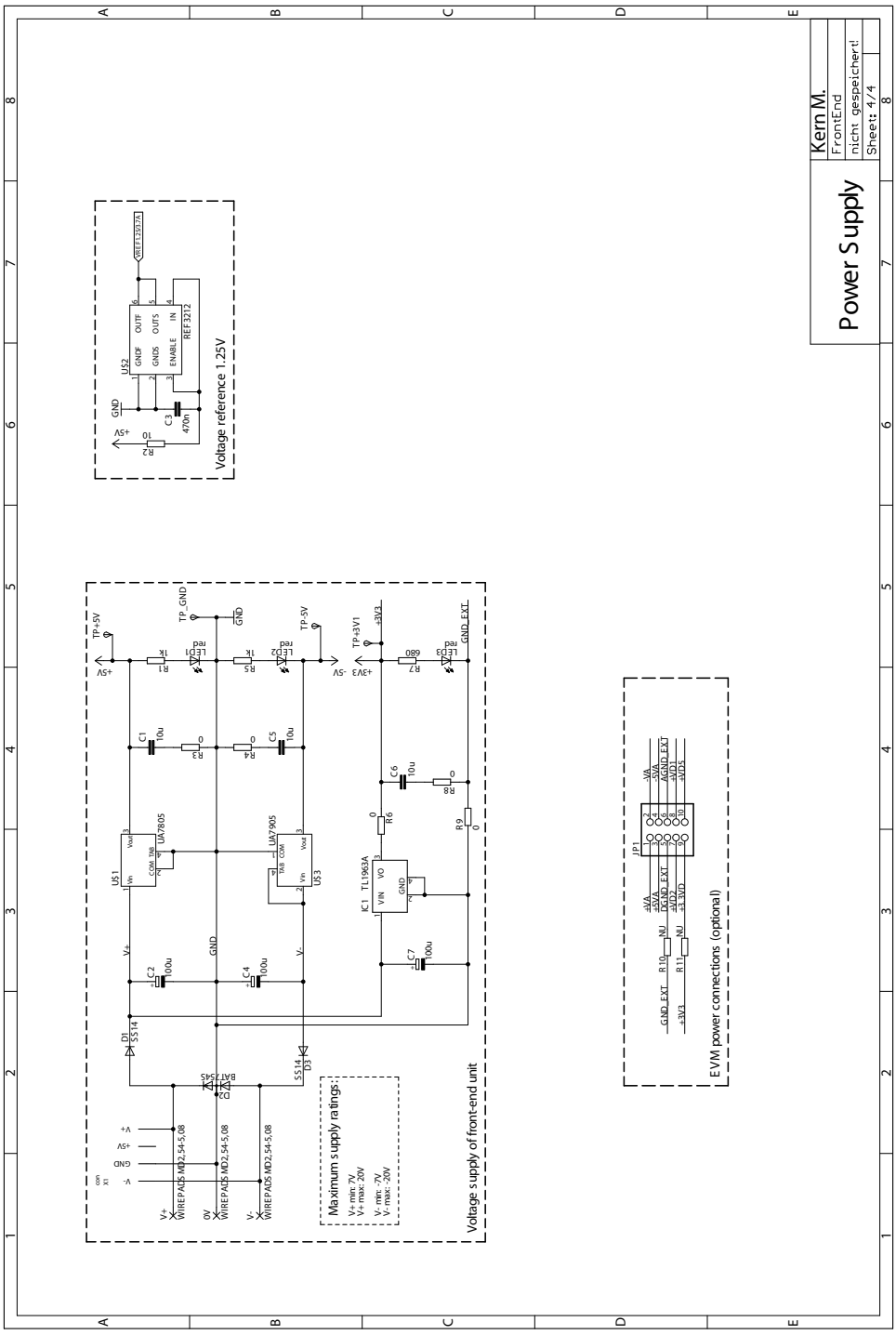
A.2 Functions of PC Console Application

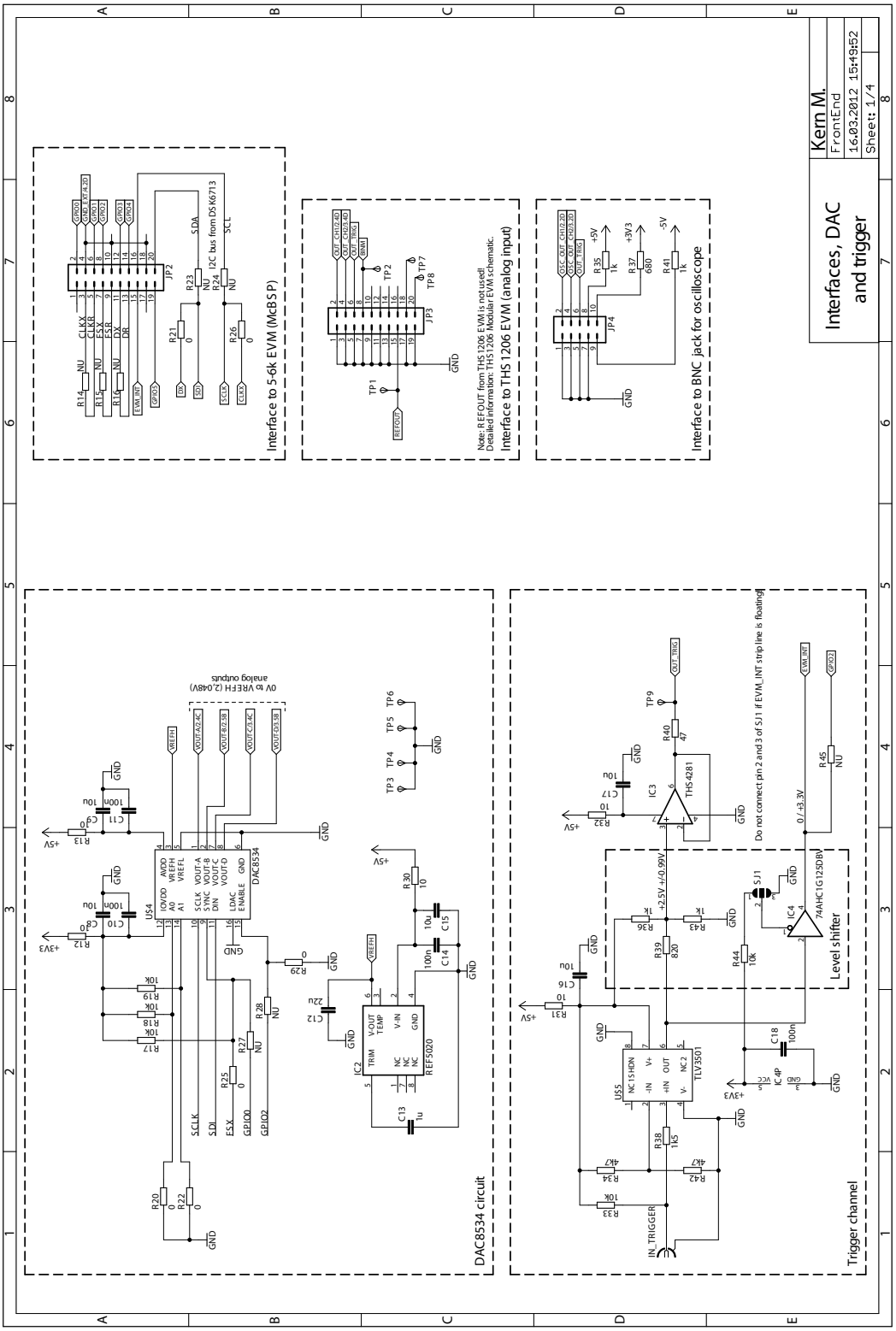
Basic information	Device status	Device ID	Version number	Device name			
Atmospheric model	All environmental parameters	Temperature	Barometric pressure	Relative humidity	Vacuum wavelength	Refraction index of air	Wavenlength in ambient air
Control	Set new reference point	Start/Abort gain and offset Calibration	Start/Abort ellipse estimation	Reset gain and offset calibration	Reset ellipse estimation		
Option	Trigger (on/off)	Trigger edge (rising/falling)	Atmospheric model (on/off)				
Calibration results	Calibration status	Gain and offset calibration result	Ellipse and Heydemann parameters				
Current position	All current position information	Current relative position in rad	Current absolute position in rad	Current reference point in rad	Current relative position in m	Current absolute position in m	Current reference point in m
Measurement results	All triggered results	Number of triggered results	Get available triggered results	Get one specific triggered result	Clear triggered result buffer		

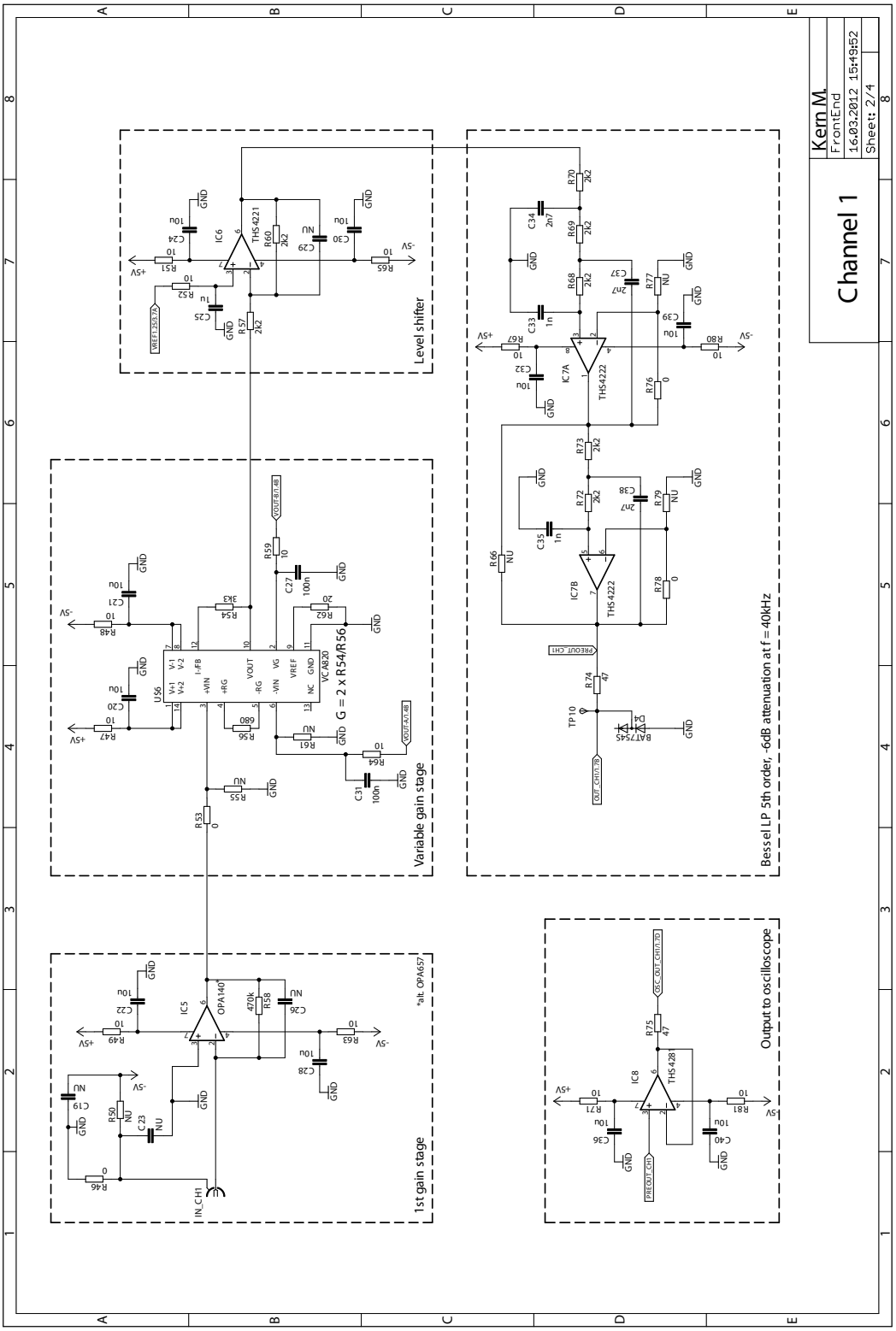
read parameter | write parameter | read/write parameter

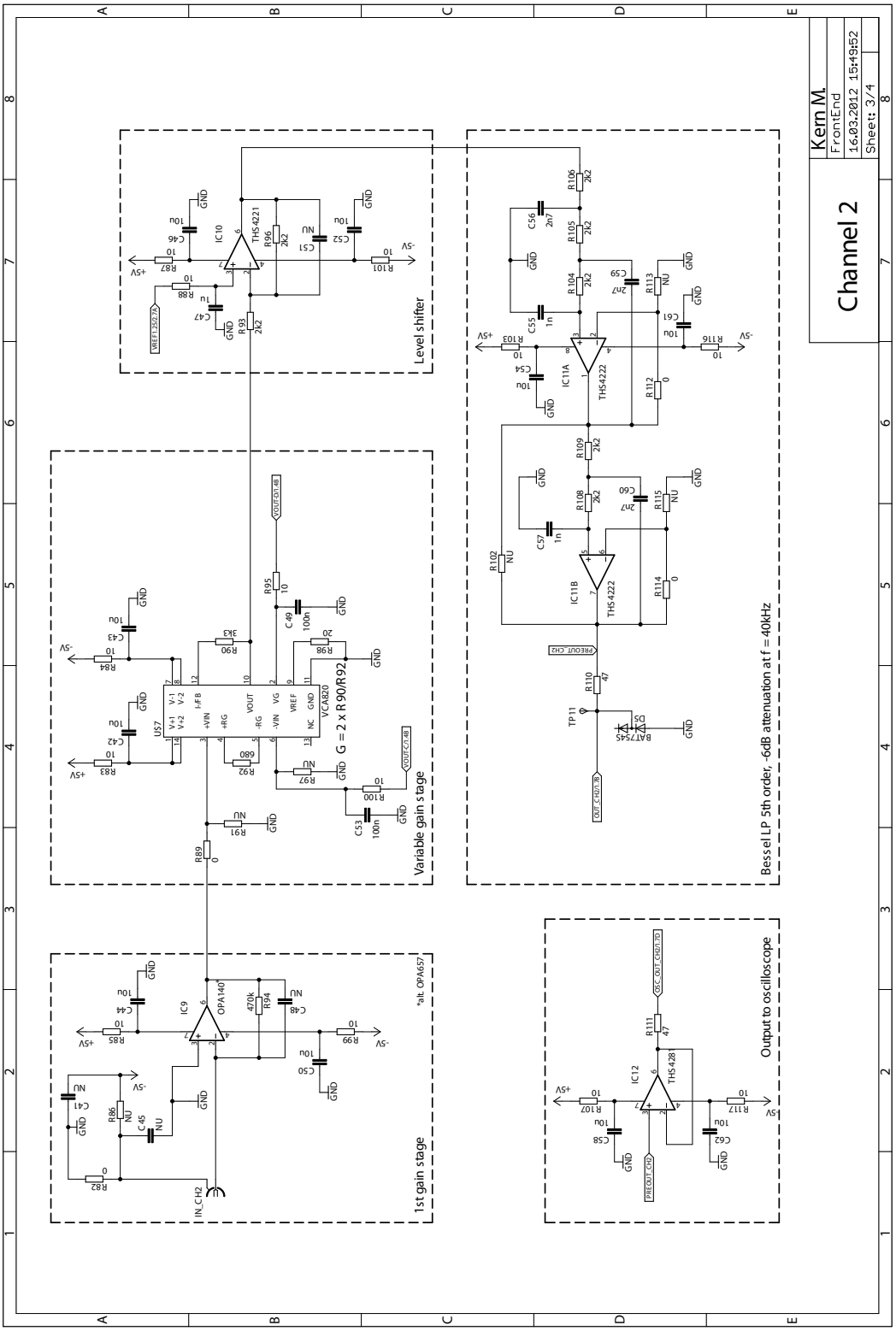
A.3 Schematics

A.3.1 Front End

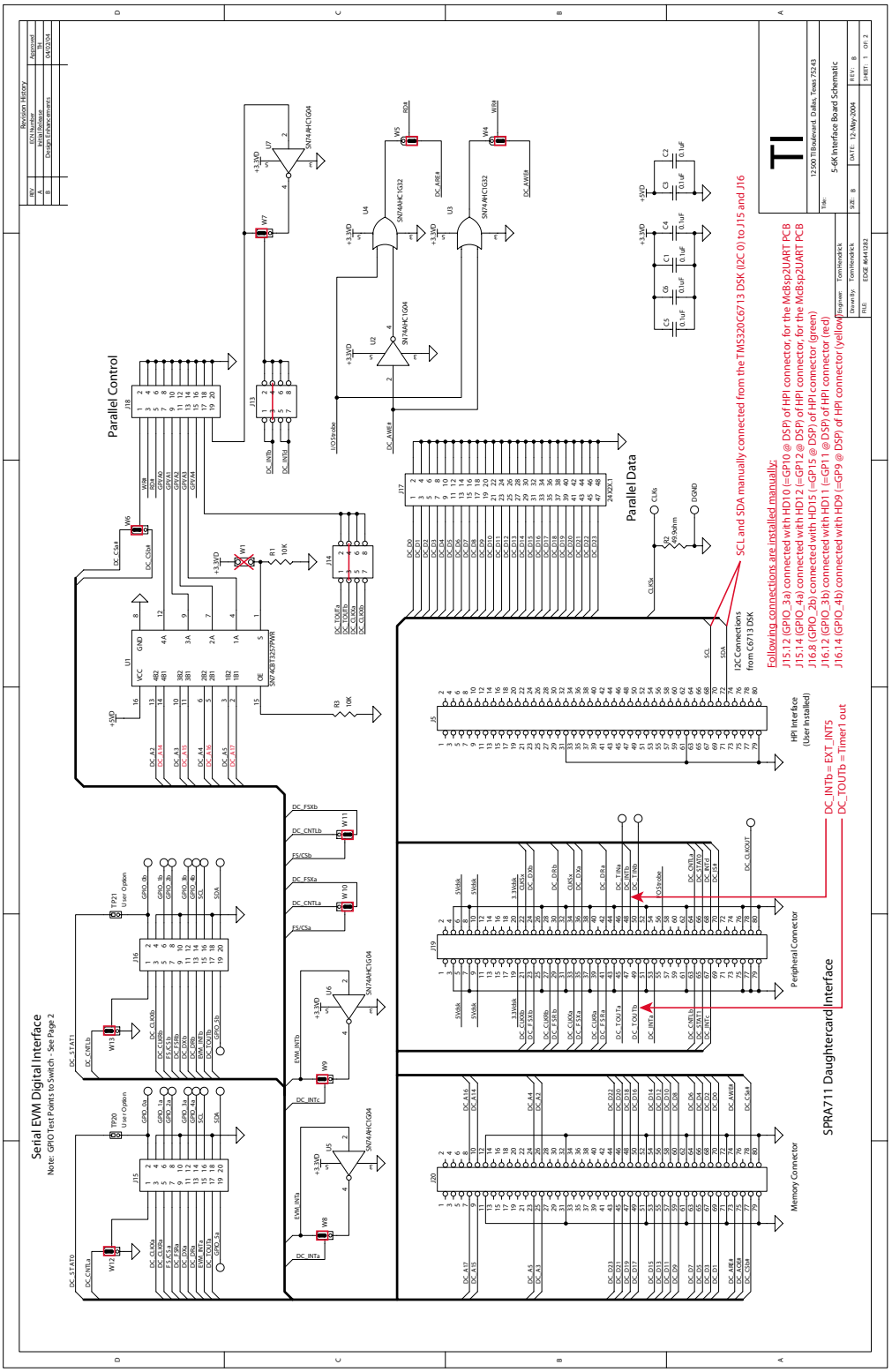


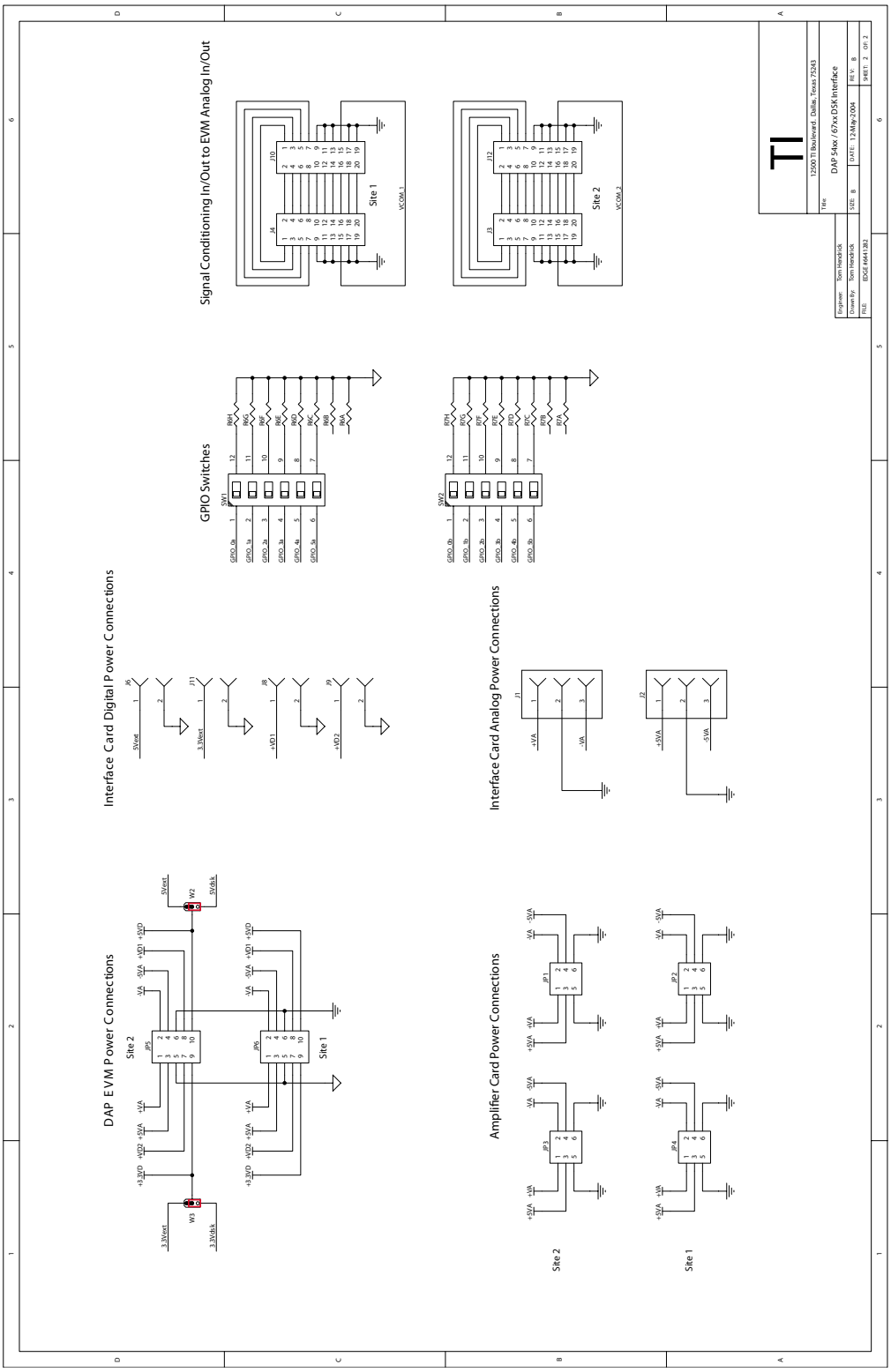

Interface to 5-6k EVM (McBSP)
Interface to THS 1206 EVM (analog input)
Interface to BNC jack for oscilloscope
DAC8534 circuit
Trigger channel
Level shifter
Interfaces, DAC and trigger
Kern M.
FrontEnd
16.03.2012 15:49:52
Sheet: 1/4
0V to VREFH (2.048V) analog outputs
Do not connect pin 2 and 3 of SJ1 if EVM_INT strip line is floating!
Note: REFOUT from THS1206 EVM is not used!
Detailed information: THS1206 Modular EVM schematic.
SCLK
SDI
ESX
GPIO0
GPIO2
IN_TRIGGER
SDA
SCL


Channel 1
Kem M.
FrontEnd
16.03.2012 15:49:52
Sheet: 2/4
Level shifter
Variable gain stage
Bessel LP 5th order, -6dB attenuation at f = 40kHz
1st gain stage
Output to oscilloscope
G = 2 x R54/R56
*alt. OPA657
IC6
THS4221
IC7A
THS4222
IC78
THS4222
IC5
OPA140
IC8
THS4281
U56
VCA820
TP10
D4
BAT754S
IN_CH1
PREOUT_CH1
OSC_OUT_CH1/78
VREF1.25/78
VOUT-AX1-48
VOUT-CH1/78

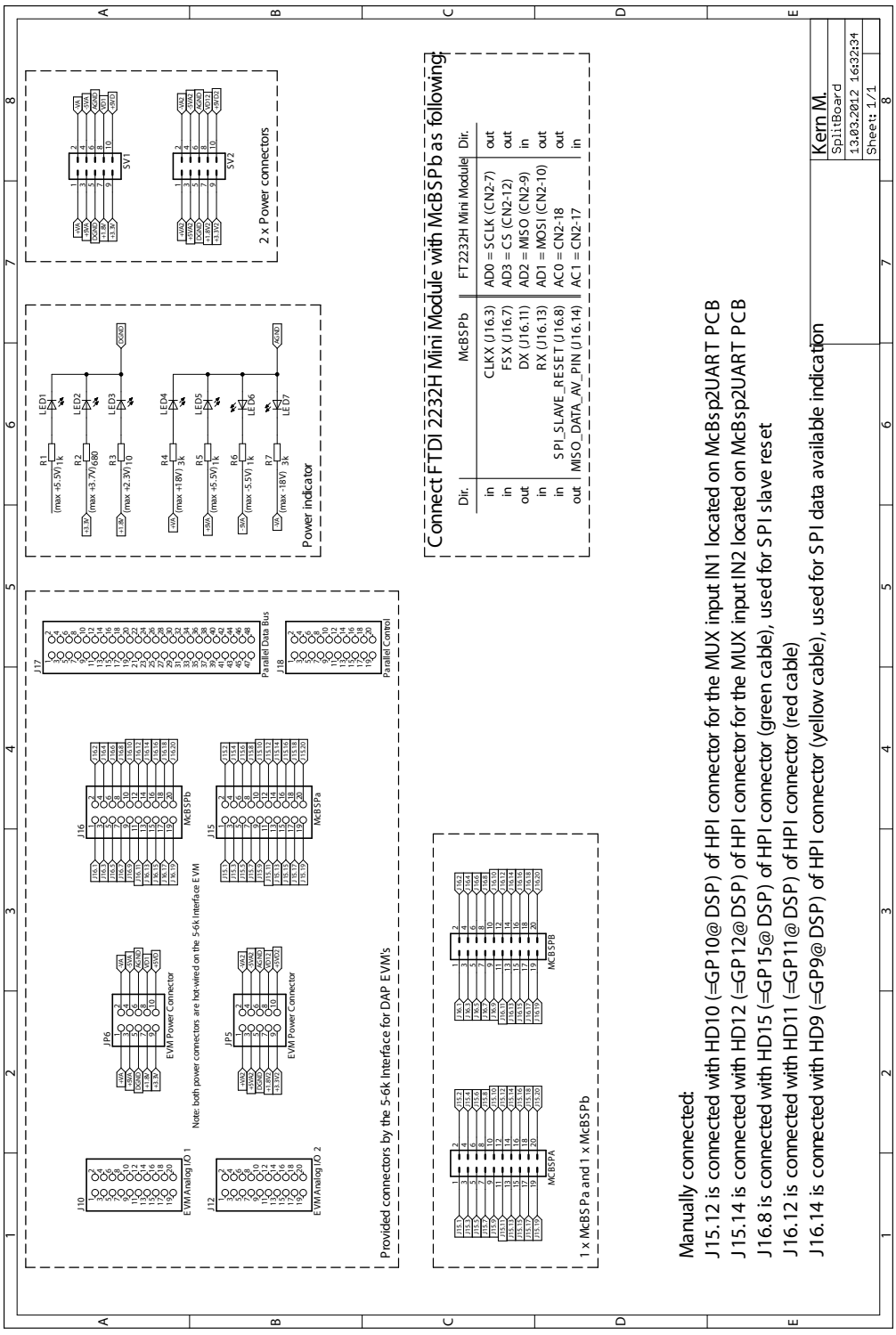

Channel 2
Kern M.
FrontEnd
16.03.2012 15:49:52
Sheet: 3/4
Level shifter
Variable gain stage
G = 2 x R90/R92
Bessel LP 5th order, -6dB attenuation at f = 40kHz
1st gain stage
Output to oscilloscope
*alt. OPA657
IC10
THS4221
IC9
OPA140
IC11A
IC11B
THS4222
IC12
THS4281
U57
VCA820
TP11
D5
BAT754S
IN_CH2
PREOUT_CH2
OSC_OUT_CH2/75
PREOUT_CH2

A.3.2 5-6K Interface

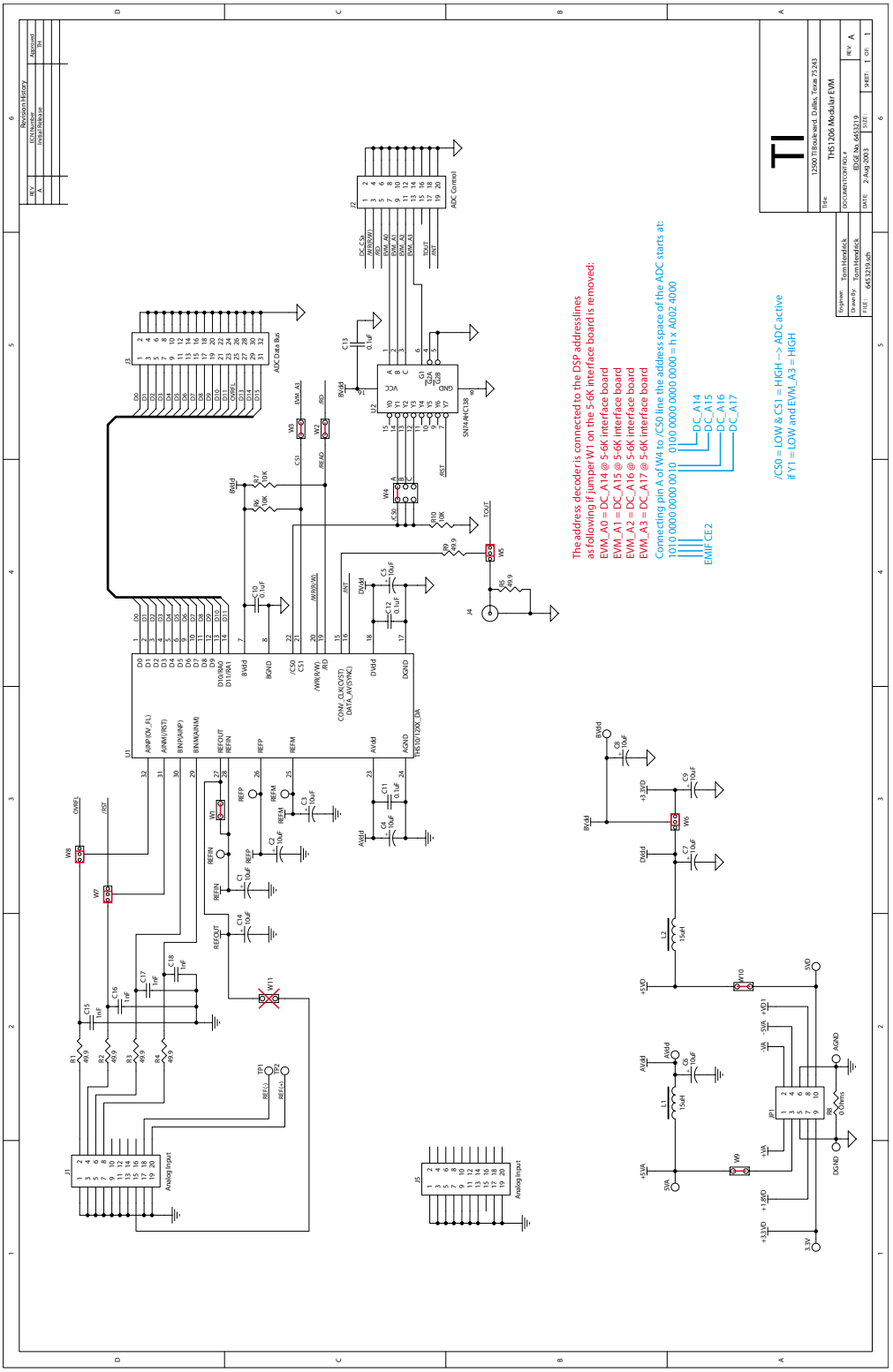



Signal Conditioning In/Out to EVM Analog In/Out
GPIO Switches
Interface Card Digital Power Connections
Interface Card Analog Power Connections
DAP EVM Power Connections
Amplifier Card Power Connections
Site 1
Site 2
VC_CM_1
VC_CM_2
TI
12500 TI Boulevard, Dallas, Texas 75243
DAP-54xx / 67xx DSK Interface
Tom Hendrick
Tom Hendrick
IDSGE HAM1-280
DATE: 12-May-2004
SHEET: 2 OF 2

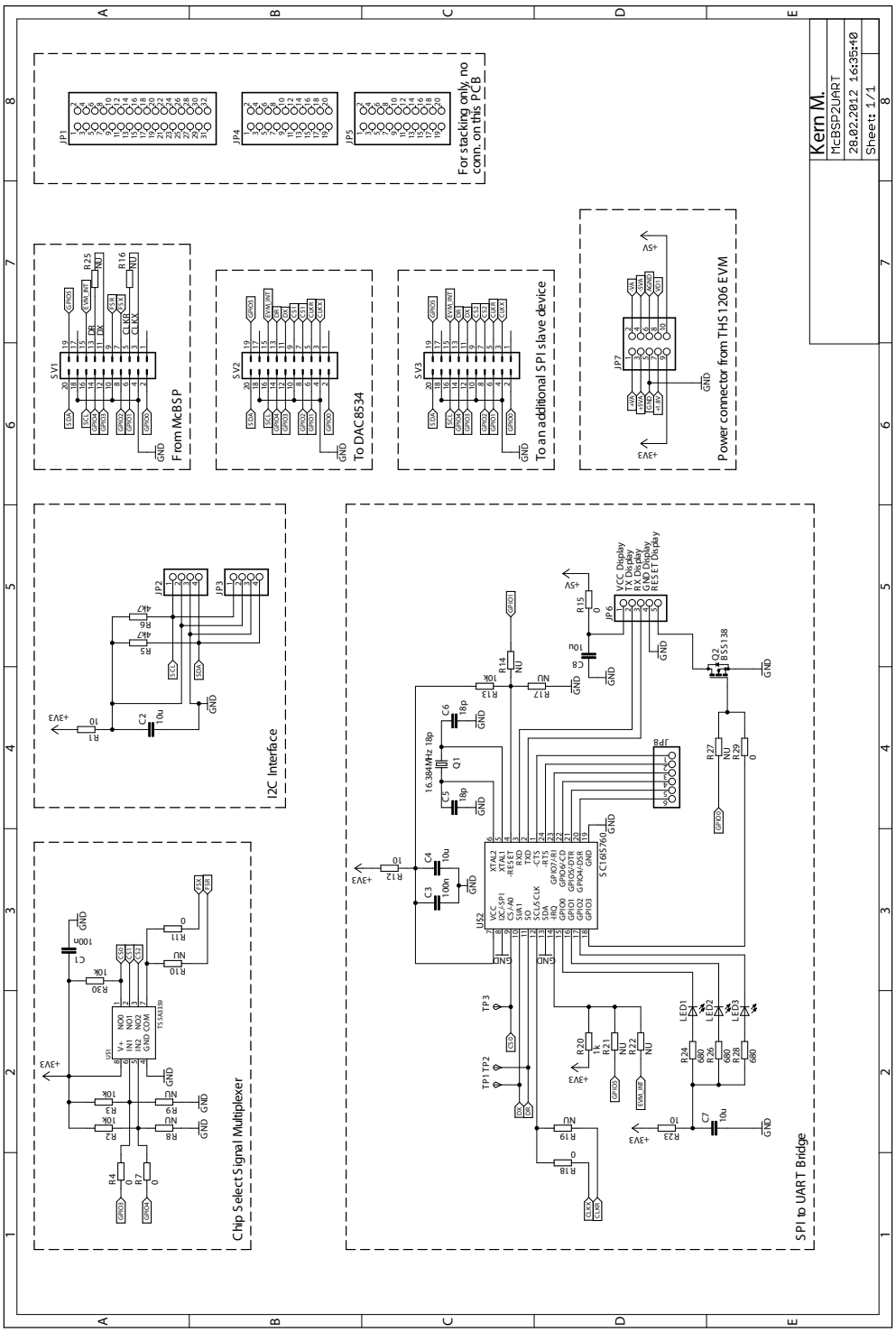
A.3.3 Split Board



A.3.4 THS1206 EVM



A.3.5 McBsp2UART



Acronyms

AAF	Anti Aliasing Filter
ADC	Analog-Digital-Converter
C/V	Current to Voltage
CCS	Code Composer Studio
CP	Circularly Polarised
CPLD	Complex Programmable Logic Device
CPU	Central Processing Unit
DAC	Digital-Analog-Converter
DSP	Digital Signal Processor
EDMA	Enhanced Direct Memory Access
EMIF	External Memory Interface
EVM	Evaluation Module
FAT	File Allocation Table
FET	Field Effect Transistor
FIFO	First In - First Out
FIR	Finite Impulse Response
GPIO	General Purpose Input/Output
GUM	Guide to the Expression of Uncertainty in Measurement
GUM-S1	Guide to the Expression of Uncertainty in Measurement - Supplement 1
HDD	Hard Disk Drive
HMI	Homodyne Michelson Interferometer

HP	Horizontally Polarised
HPI	Host Port Interface
HWI	Hardware Interrupt
I/O	Input/Output
I^2C	Inter-Integrated Circuit
IC	Integrated Circuit
IDE	Integrated Development Environment
JCGM	Joint Commitee for Guides in Metrology
LCP	Left Circularly Polarised
LED	Light-Emitting Diode
LP	Linearly Polarised
McBsp	Multichannel Buffered Serial Port
MPSSE	Multi-Protocol Synchronous Serial Engine
MU	Measurement Uncertainty
NIST	National Institute of Standard and Technology
NPBS	Non Polarising Beam Splitter
OPA	Operation Amplifier
OPD	Optical Path Difference
PBS	Polarising Beam Splitter
PC	Personal Computer
PCB	Printed Circuit Board
PDF	Probability Density Function
RCP	Right Circularly Polarised
RLS	Recursive Least Square
SD	Secure Digital
SDRAM	Synchronous Dynamic Random Access Memory

SNR	Signal-to-Noise Ratio
SPI	Serial Peripheral Interface
SPU	Signal Processing Unit
SWI	Software Interrupt
TCC	Transfer Complete Code
TSK	Task
UART	Universal Asynchronous Serial Receive Transmit
USB	Universal Serial Bus
VGA	Variable Gain Amplifier
VP	Vertically Polarised
WLI	White Light Interferometer

Bibliography

[1] 4D LABS. *µOLED-32028-P1 (SGC)*, 2010. Serial AMOLED display module, data sheet.

[2] 4D LABS. *PICASO-SGC command set*, 2011. Software interface specification.

[3] AGILENT. *Dynamic calibrator 5530*, February 2012. Data sheet.

[4] ANALOG DEVICES. OPA noise relationships, 2009.

[5] ATTIVISSIMO F., GIAQUINTO N. AND SAVINO M. Uncertainty evaluation in dithered ADC-based instruments. *Measurement 41* (2008), 364–370.

[6] AUER M. Hochauflösende optische Wegmessung. Master Thesis, Technische Universität Graz, 2011.

[7] BIRCH K. P. AND DOWNS M. J. An updated Edlén equation for the refractive index of air. *Metrologia 30* (1993), 155–162.

[8] BIRCH K. P. AND DOWNS M. J. Correction to the updated Edlén equation for the refractive index of air. *Metrologia 31* (1994), 315–316.

[9] BOSCH SENSORTEC. *BMP085*, 2009. Digital pressure sensor, Data sheet.

[10] BURR BROWN. Designing photodiode amplifier circuits with OPA 128, 1994. Technical report.

[11] DUMS K. Messtechnische Erfassung von Weißlichtinterferogrammen unter besonderer Berücksichtigung der Messunsicherheit. Master Thesis, Technische Universität Graz, 2011.

[12] EDLÉN B. The refractive index of air. *Metrologia 2*, 2 (1966), 71–80.

[13] FITZGIBBON A., PILU M. AND FISHER R. B. Direct least square fitting of ellipse. *IEEE Transactions on Pattern Analysis and Machine Intelligence 21* (1999), 476–480.

[14] FUTURE TECHNOLOGY DEVICES INTERNATIONAL LTD. *FT2232H*, 2010. Dual high speed USB to multipurpose UART/FIFO IC, data sheet.

[15] FUTURE TECHNOLOGY DEVICES INTERNATIONAL LTD. *LibMPSSE - SPI*, 2011. User's guide.

[16] FUTURE TECHNOLOGY DEVICES INTERNATIONAL LTD. *FT2232H evaluation module*, 2011. USB hi-speed evaluation module, data sheet.

[17] GREGORČIČ P., POŽAR T. AND MOŽINA J. Quadrature phase-shift error analysis using a homodyne laser interferometer. *Optical Express 17*, 18 (2009), 16322–16331.

[18] HAAGER W. *Regelungstechnik*, vol. 2.03. öbv hpt VerlagsgmbH & Co. KG, 1997.

[19] HEYDEMANN P. L. M. Determination and correction of quadrature fringe measurement errors in interferometers. *Applied Optics 20*, 19 (1981), 3382–3384.

[20] INFINEON. *BPX65*, 2000. Silizium PIN photo diode, data sheet.

[21] INTERNATIONAL BUREAU OF WEIGHTS AND MEASURES (BIPM). *Guide to the Expression of Uncertainty in Measurement*. JCGM, 2008.

[22] INTERNATIONAL BUREAU OF WEIGHTS AND MEASURES (BIPM). *Supplement 1 to the Guide to the Expression of Uncertainty in Measurement - Propagation of distributions using a Monte Carlo method*. JCGM, 2008.

[23] JAEGER G. Limitations of precision length measurements based on interferometers. *Measurement 43* (2010), 652–658.

[24] JENAER MESSTECHNIK GMBH. *Dual-frequency laser interferometer ZLM 700/800/900*, April 2011. Product description.

[25] KACKER R., SOMMER K. D. AND KESSEL R. Evolution of modern approaches to express uncertainty in measurement. *Metrologia 44* (2007), 513–529.

[26] NXP. *SC16IS740/750/760*, 2011. Single UART with I2C-bus/SPI interface, 64 bytes of transmit and receive FIFOs, IrDA SIR built-in support, data sheet.

[27] OPPENHEIM A. V., SCHAFER R. W. AND BUCK J. R. *Zeitdiskrete Signalverarbeitung*. Pearson Education, 2004.

[28] PETRŮ F. AND ČÍP. Problems regarding linearity of data of a laser interferometer with a single-frequency laser. *Precision Engineering 23* (1999), 39–50.

[29] RENISHAW. *XL-80 Laser System*, 2011.

[30] SEIFERT, T. *Verfahren zur schnellen Signalaufnahme in der Weißlichtinterferometrie*. Dissertation, Universität Erlangen-Nürnberg, 2005.

[31] SENSIRION. *SHT75*, 2008. Digital humidity and temperature sensor, data sheet.

[32] SIMON HAYKIN. *Adaptive filter theory (4th edition)*. Pearson Education, 2002.

[33] SIOS MESSTECHNIK GMBH. Stabilisierter He-Ne-Laser Serie SL 03, November 2007. Data sheet.

[34] SIOS MESSTECHNIK GMBH. Stabilisierter He-Ne-Laser Serie SL 03. Ser. -Nr. 181130110, February 2010. Test certificate.

[35] SOMMER K. D. Praxisgerechtes Bestimmen der Messunsicherheit nach GUM. *Technisches Messen 71* (2004), 52–66.

[36] SOMMER K. D. AND SIEBERT B. R. L. Systematic approach to the modelling of measurements for uncertainty evaluation. *Metrologia 43* (2006), 200–210.

[37] SPECTRUM DIGITAL, INC. *TMS320C6713 DSK*, 2003. data sheet.

[38] STONE, J. A. AND ZIMMERMAN, J. H. Refractive index of air calculator. http://emtoolbox.nist.gov, July 2011. National Institute of Standards and Technology (NIST).

[39] TEXAS INSTRUMENTS. *TMS320C6000 DSP/BIOS*, 2000. User's guide.

[40] TEXAS INSTRUMENTS. *TMS320C67x FastRTS library*, 2002. Programmer's reference.

[41] TEXAS INSTRUMENTS. *Modular THS1206EVM*, 2003. User's guide.

[42] TEXAS INSTRUMENTS. *ADC THS1206*, 2003. 12-bit, 4 analog input, 6MSPS, simultaneous sampling analog-to-digital converter, data sheet.

[43] TEXAS INSTRUMENTS. *DAC8534*, 2004. Quad channel, low power, 16-bit, serial input digital-to-analog converter, data sheet.

[44] TEXAS INSTRUMENTS. *5-6K interface board EVM*, 2004. User's guide.

[45] TEXAS INSTRUMENTS. *TMS320C6000 chip support library*, 2004. API reference guide.

[46] TEXAS INSTRUMENTS. *TMS320C6000 DSP general-purpose input/output (GPIO)*, 2004. Reference guide.

[47] TEXAS INSTRUMENTS. *TLV3501*, 2005. 4.5ns rail-to-rail, high speed comparator in microsize packages, data sheet.

[48] TEXAS INSTRUMENTS. *TMS320C6000 DSP 32-bit timer*, 2005. Reference guide.

[49] TEXAS INSTRUMENTS. *TMS320C6713B floating point digital signal processor*, 2006. data sheet.

[50] TEXAS INSTRUMENTS. *TMS320C6000 DSP enhanced direct memory access (EDMA) controller*, 2006. Reference guide.

[51] TEXAS INSTRUMENTS. *TMS320C6000 DSP multichannel buffered serial port (McBSP)*, 2006. Reference guide.

[52] TEXAS INSTRUMENTS. *TMS320C6000 DSP external memory interface (EMIF)*, 2008. Reference guide.

[53] TEXAS INSTRUMENTS. *VCA 820*, 2009. Wideband, > 40 dB adjust range, linear in dB variable gain amplifier, data sheet.

[54] TEXAS INSTRUMENTS. *TMS320C67x DSP library*, 2010. Programmer's reference guide.

[55] TEXAS INSTRUMENTS. *OPA140*, 2010. High-precision, low-noise, rail-to-rail output, 11MHz JFET operational amplifier, data sheet.

[56] TIETZE U. AND SCHENK CH. *Halbleiter-Schaltungstechnik*. Springer, 2002.

[57] WANG Y.C. *Präzisionsprüfgerät für Nanomesstaster*. PhD Thesis, Technische Universität Ilmenau, 2002.

[58] WU C. M. AND SU C. S. Nonlinearity in measurements of length by optical interferometry. *Measurement Science Technology 7* (1995), 62–68.

[59] WU C. M., SU C. S. AND PENG G. S. Correction of nonlinearity in one-frequency optical interferometry. *Measurement Science Technology 7* (1996), 520–524.