



TECHNISCHE  
UNIVERSITÄT  
WIEN

Vienna University of Technology

## DIPLOMARBEIT

### Zweidimensionale Röntgenfluoreszenzanalyse zur Darstellung biologischer Proben in stark streuender Matrix

ausgeführt am

Atominstitut

der Technischen Universität Wien

unter der Anleitung von

Ao.Univ-Prof.i.R. Dipl.Ing.Dr. Peter Wobrauschek

durch

Alexander Utz BSc.

Rothof 3

4312 Ried/Rdm

Wien am 9. Mai 2012

## **Danksagung**

Ich möchte mich bei allen bedanken, die zur Entstehung dieser Diplomarbeit beigetragen haben:

Herrn Univ. Prof. DI. Dr. Peter Wobrauschek für die hervorragende Betreuung während dieser Diplomarbeit

Frau Univ. Prof. DI. Dr. Christina Streli für die stets bereitwillige und freundliche Hilfe

Herrn DI. Bernhard Pemmer für das Einbringen seines umfangreichen Wissens

Herrn Stephan Smolek für die Unterstützung im Bereich der Programmierung

Herrn Univ.Prof.Dr. Gert Lubec für die Bereitstellung der Proben

der gesamten ROELAB-Arbeitsgruppe für das gute Arbeitsklima

# 1 Inhaltsverzeichnis

1.	Einleitung.....	6
1.1	Das Bohr'sche Atommodell .....	7
1.2	Röntgenspektren .....	7
1.3	Wechselwirkung von Elektronen mit Materie.....	9
1.3.1	Die Bremsstrahlung:.....	9
1.3.2	Charakteristische Strahlung.....	11
1.4	Wechselwirkung von Strahlung mit Materie.....	14
1.4.1	Photoeffekt .....	14
1.4.2	Elastische Streuung .....	15
1.4.3	Inelastische Streuung.....	16
1.5	Schwächung der Strahlung .....	17
2	Röntgenfluoreszenzanalyse.....	19
2.1	Instrumentierung.....	19
2.1.1	Erzeugung der Röntgenstrahlung. ....	19
2.1.2	Sekundärtarget.....	21
2.1.3	Energiedispersive Detektoren .....	21
2.1.4	Aufbau von Halbleiter- Detektoren.....	22
2.2	Signalverarbeitung.....	26
2.3	Detektorartefakte .....	27
2.3.1	Escape Peak.....	27
2.3.2	Summen- Peaks .....	28
2.3.3	Compton Kante .....	28
3	Gelelektrophorese .....	29
4	Messaufbau .....	30
4.1	Röntgenfluoreszenz Messanlage .....	30
4.1.1	Strahlenquelle für die Anregung .....	30
4.1.2	Sekundärtarget.....	31
4.1.3	Detektor zum Erfassen der Röntgenfluoreszenz .....	32

4.1.4	Signalverarbeitung .....	34
4.2	Koordinatentisch.....	35
4.2.1	Manueller Koordinatentisch.....	35
4.2.2	Antrieb durch Schrittmotoren.....	36
4.2.3	Stromversorgung für den Schrittmotor .....	37
4.2.4	I/O Schnittstellen für den Schrittmotor .....	38
4.2.5	Die Platine .....	38
4.2.6	Steuerung des Schrittmotors.....	40
4.2.7	Kopplung von Koordinatentisch und Schrittmotor .....	40
4.3	Kombination von Koordinatentisch und XRF Anlage .....	42
4.3.1	Der mechanische Aufbau .....	42
4.3.2	Software .....	43
4.4	Das Programm Datamove.....	47
5	Einstellen der Mess- Anordnung.....	48
5.1	Die Geometrie.....	48
5.2	Justierung des Detektors .....	49
5.3	Bestimmen der Auflösung .....	51
5.4	Bestimmen des Leerwerts.....	56
6	Messungen mit dem XRF Scanner: .....	59
6.1	Messung einer definierten Probe .....	59
6.2	Messung von Elektrophorese-Gel. ....	61
6.2.1	Scannen des Elektrophorese-Gels mit der Bezeichnung „Gel 1“.....	61
6.2.2	Scannen des Elektrophorese Gel mit der Bezeichnung „Muera“ .....	64
7	Zusammenfassung und Ausblick .....	67
8	Abbildungsverzeichnis.....	69
9	Tabellenverzeichnis .....	71
10	Literaturverzeichnis .....	72
11	Anhang.....	73
11.1	Steuerprogramm .....	73
11.2	Datamove.....	96
11.3	Koordinatentisch.....	98

11.4	Schrittmotor .....	99
11.5	Schaltnetzteil: .....	100

## Kurzfassung

In den letzten Jahren wurde der entscheidende Einfluss von Metalloproteine auf physiologische und pathologische Prozesse festgestellt. Durch Gelelektrophorese werden Metalloproteine in ihre molekularen Eigenschaften im Elektrophorese - Gel aufgespalten und durch Marker sichtbar gemacht.

Für die weitere Untersuchung des stark streuenden Elektrophorese- Gels wurde ein Röntgenfluoreszenz Scanner entwickelt, wie in dieser Arbeit beschrieben ist. Um die starke Streuung der primären Röntgenstrahlung der leichten Matrix zu reduzieren wird ein Sekundärtarget eingesetzt, dass die anregende Strahlung linear polarisiert. Durch einen flexiblen Aufbau kann das Sekundärtarget und die Röntgenröhre, für die vermuteten Elemente im Elektrophorese-Gel, optimiert werden. Der Scan des Gels ist durch geeignete Wahl eines Kollimators mit der Pixelgröße von 3mm und 6mm möglich. Je nach gewünschter Auflösung kann die Kollimatorgröße gewählt werden. Der Aufbau kann auch für spektrale Röntgenoptiken eingesetzt werden, die einen Strahldurchmesser von 100µm aufweisen, da der Koordinatenschlitten über eine Positionsgenauigkeit von 20µm verfügt.

Der zweidimensionalen Röntgenfluoreszenz Scanners kann eine Vielzahl von biologische Proben mit Flüssigkeits- und Luft einschlüssen untersuchen, aber auch andere (technische) Proben von Interesse. Besonders hervorzuheben ist die scanbare Größe von 120mm x 180mm, wodurch sich die Anlage von kommerziellen XRF Scannern unterscheidet. Die Kombination von computergesteuerten Koordinatentisch und Röntgenfluoreszenz Anlage ermöglicht einen vollautomatischen Scan. Während der Messung wird ständig das Spektrum des aktuell gemessenen Punkts aktualisiert. Bei Besonderheiten kann jederzeit in das Programm eingegriffen werden und die Schrittweite vergrößert oder verringert werden. Nach dem Scan werden die Spektren vollautomatisch nach Elementen entfaltet und ausgewertet (Intensität, Element und Ort). Schließlich werden den ausgewerteten Elementen Positionen im zweidimensionalen Raum zugeordnet. Die Verteilung der Elemente über die zweidimensionale Oberfläche kann durch Säulen oder Farbcodes dargestellt werden. Neben dem qualitativen Scan kann durch weitere Schritte ein quantitativer Scan realisiert werden, der durch den Zusammenhang von Intensität und Konzentration über röntgenphysikalische Grundlagen gegeben ist.

Die Probe wird auf einer wechselbaren PMMA Platte angebracht. Die Position des Koordinatentisches kann auch ohne den Computer verändert werden, was einen schnellen Wechsel von Proben erlaubt.

Das Spektrometer zeichnet sich durch einen kompakten stabilen Aufbau aus und ist auch Strahlenschutzmäßig sicher ausgelegt. Durch die ausgefeilte Software des XRF Scanners können Scans der gleichen Probe übereinander gelegt werden und so die zweidimensionale Auflösung verbessert werden.

# 1. Einleitung

Beobachtet und in einer wissenschaftlichen Arbeit verfasst wurde die Röntgenstrahlung erstmals vom deutschen Physiker Wilhelm Conrad Röntgen im Jahre 1895. Die Röntgenstrahlung oder auch als X-Strahlung bezeichnet, ist ein Teil des elektromagnetischen Spektrums. Zu dem elektromagnetischen Spektrum gehören unter anderen auch Radiowellen,  $\gamma$ -Strahlung oder sichtbares Licht wie in Abbildung 1 dargestellt ist. Das elektromagnetische Spektrum wird in Wellenlängen oder in Energie eingeteilt. Die Röntgenstrahlung befindet sich zwischen 100eV und einigen MeV bzw. im Nano- und Pikometer Bereich. Im Bereich der Röntgenfluoreszenzanalyse ist ein Energieintervall von einigen 100eV bis 100keV ausreichend um chemische Elemente in einer Probe effizient anzuzeigen.

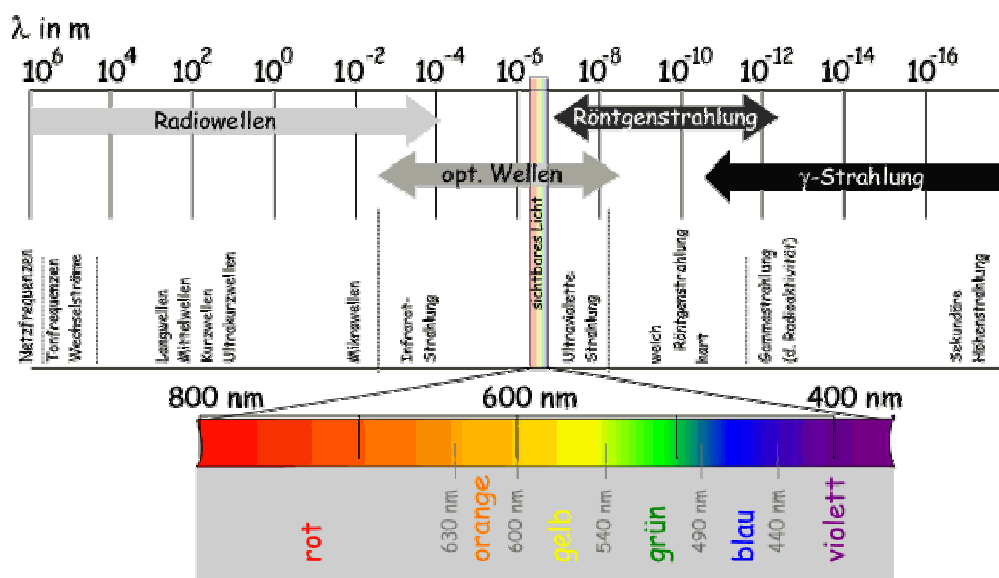


Abbildung 1: Elektromagnetisches Spektrum (Rothhardt, 2011)

Röntgenstrahlung wird zumeist durch die Beschleunigung eines Elektrons und anschließendem Abbremsen durch ein Target erzeugt. Die entstandene Röntgenstrahlung liegt zwischen 100eV und 100keV und lässt sich in

- schwache, ( 100 eV bis 1keV)
- klassische ( 1keV bis 100keV)
- und harte ( größer 100keV)

Röntgenstrahlung unterteilen.

## 1.1 Das Bohr'sche Atommodell

In diesem Modell bewegen sich Elektronen in kreisförmigen Bahnen um einen Atomkern. Im Atom befinden sich  $Z$  Protonen, welche positiv geladen sind, und  $A-Z$  Neutronen.  $Z$  steht in dieser Beschreibung für die Ordnungszahl und  $A$  für die Massenzahl des Elements. Die Elektronen in der Annahme des Bohr'schen Atommodells kreisen aufgrund der elektrischen Anziehung und der entgegengesetzten Zentrifugalkraft um den Kern. Da es sich hierbei um eine ständige gleichmäßige Beschleunigung handelt, müsste ständig Energie abstrahlt werden. Durch die Energieabgabe ist das System nicht mehr geschlossen und die Elektronen würden in den Kern stürzen. Stabile Atome wären nicht möglich.

Aus diesem Grund wurden die Bohr'schen Postulate aufgestellt, die folgendes Besagen:

- Die Atome liegen in bestimmten Zuständen mit diskreten Energien  $E_0, E_1, E_2, \dots$  vor
- Beim Übergang von einem solchen stationären Zustand mit der Energie  $E_a$  in einen anderen Zustand  $E_e$  wird der Energieunterschied in Form eines Photons emittiert ( $E_a > E_e$ ) oder absorbiert ( $E_a < E_e$ ) (Reineker and Schulz, 2007).

Diese Postulate erklären die Stabilität und die Linienspektren eines Atoms.

Die kreisförmigen Bahnen der Elektronen werden auch Schalen genannt und sind nach Barkla mit K, L, M, N, ... bezeichnet.

## 1.2 Röntgenspektren

Bei einem Spektrum wird die Intensität  $I$  der Strahlung gegenüber der Energie  $E$  oder der Wellenlänge  $\lambda$  dargestellt. Die Form des Spektrums ändert sich, wenn die Energie gegen die Wellenlänge vertauscht wird (siehe Abbildung 3 und Abbildung 2) die Information bleibt jedoch dieselbe.

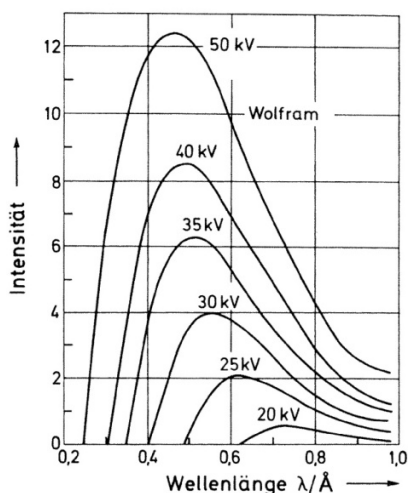


Abbildung 3: wellenlängenaufgelöstes Röntgenspektrum (Haken and Wolf, 2004)

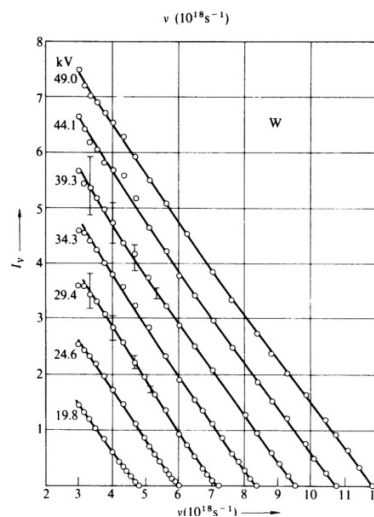


Abbildung 2: energieaufgelöstes Röntgenspektrum (Dyson, 1990)



Es besteht folgender Zusammenhang zwischen Energie und Wellenlänge:

$$E = h\nu = \frac{h \cdot c_0}{\lambda}$$

Formel 1

mit

$$c_0 = \nu \cdot \lambda$$

Formel 2

Die Energie kann in Elektronenvolt [eV] angegeben werden. Dies vereinfacht die Formel und es gilt:

$$E[keV] = \frac{12,39}{\lambda[\text{\AA}]} = \frac{1,239}{\lambda[\text{nm}]}$$

Formel 3

	Wert	Einheit	Bezeichnung
<b>h</b>	$6,626 \cdot 10^{-34}$	Js	Plancksches Wirkungsquantum
<b>c<sub>0</sub></b>	$299 \cdot 10^6$	m/s	Lichtgeschwindigkeit
<b>ν</b>	-	s <sup>-1</sup>	Frequenz
<b>λ</b>	-	m	Wellenlänge
<b>E</b>	-	J	Energie

Tabelle 1: Bezeichnung der Variablen für die Formel 1, Formel 2, Formel 3

Die Intensität der Strahlung wird in

$$I \left[ \frac{W}{cm^2} \right] = \frac{\text{Energie der Strahlung [J]}}{\text{Fläche}[cm^2] \cdot \text{Zeit}[s]}$$

Formel 4

angegeben.

Für die Anwendung in der Röntgenfluoreszenzanalyse wird die Anzahl der Impulse bzw. der Photonen gemessen. Die Impulse werden in “counts per second” (cps) angegeben.

Atomare Stoßprozesse produzieren Charakteristische Röntgenstrahlung, die ein diskretes Spektrum besitzt. Sie wird überlagert von Kontinuierlicher Bremsstrahlung, die durch Abbremsung, unelastisches Streuen von geladenen Teilchen im Kernfeld, erzeugt werden.

## 1.3 Wechselwirkung von Elektronen mit Materie

Wird ein Elektron durch eine Spannung beschleunigt und trifft auf ein Atom kommt es zu Wechselwirkungen. Durch diese Wechselwirkungen können Elektronen oder Photonen aus dem beschossenen Atom emittiert werden. Diese Wechselwirkungen werden in den nachfolgenden Kapiteln erläutert.

### 1.3.1 Die Bremsstrahlung:

Wird ein geladenes Teilchen durch das Coulomb-Feld eines Atomkerns von seiner Bahn abgelenkt, so wird Bremsstrahlung freigesetzt (siehe Abbildung 4). Beim Ablenken wird dem Teilchen kinetische Energie entzogen, welche anschließend in Form von Bremsstrahlung abgegeben wird.

Die Ablenkung der Teilchen nimmt zu, je leichter das Projektil im Vergleich zum Target ist und je größer die Differenz der Ladung zum Projektil ist.

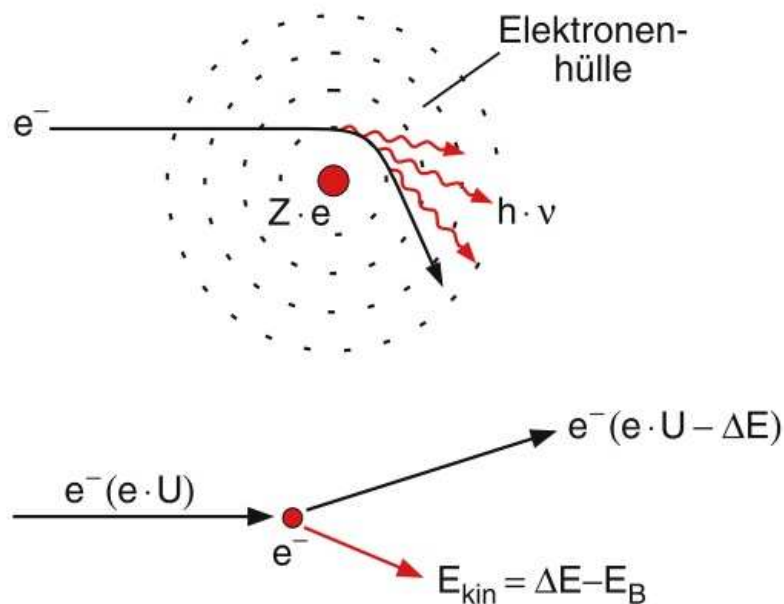


Abbildung 4: Bremsstrahlung(Demtröder, 2009)

Diese Eigenschaften werden in einer Röntgenröhre genutzt. Als Projektil wird ein Elektron, das eine geringe Masse und eine hohe Ladung besitzt, verwendet. Durch ein E- Feld wird das Elektron beschleunigt und anschließend an einem Target, mit einer hohen Ordnungszahl, gestreut. Da der Kern eine um wesentliche höhere Masse besitzt, kann jeder Impuls des wechselwirkenden Elektrons aufgenommen werden. Der Energieverlust der Elektronen ist durch einen Maximalwert beschränkt, der gleich der Energie des beschleunigten Elektrons ist. Dabei wird das Elektron durch einen einzigen Stoß zum Stillstand gebracht.

Die nachfolgende Formel veranschaulicht das Energieerhaltungsgesetz wobei  $E_{kin}^V$  die Energie vor und  $E_{kin}^N$  die Energie nach dem Einfluss der Coulomb-Kraft ist.

$$E_{ph} = hv = E_{kin}^V - E_{kin}^N$$

Formel 5

Energie der beschleunigten Elektroden wird durch die angelegte Hochspannung in der Röntgenröhre bestimmt.

$$E_{kin}^V = e \cdot U$$

Formel 6

Die Intensität der Bremsstrahlung, durch die Streuung von Elektronen, wurde durch Kulenkampff und Kramers in einer Formel dargestellt:

$$I(v) = C(Z \cdot (v_{max} - v) + bZ^2)$$

Formel 7(Kulenkampff, 1923, Kramers, 1923)

In der Formel 7 sieht man, dass die Intensität von der Ordnungszahl und der Energie abhängig ist, die Variablen C und b sind Konstanten. Um die Gesamtintensität der Bremsstrahlung zu erhalten wird über den gesamten Energiebereich der Bremsstrahlung integriert.

Die Gesamtintensität der Bremsstrahlung einer Röntgenröhre lässt sich durch Formel 8 näherungsweise berechnen.

$$I_{ges} \approx C \cdot i \cdot U^2 \cdot Z$$

Formel 8

	Wert	Einheit	Bezeichnung
<b>I</b>	-	W/m <sup>2</sup>	Gesamtintensität
<b>i</b>	-	A	Röhrenstrom
<b>U</b>	-	V	Beschleunigungsspannung
<b>Z</b>	1,2,3...	-	Ordnungszahl
<b>e</b>	1,6 10 <sup>19</sup>	As	Ladung
<b>C,b</b>	-	-	Konstanten
<b>C</b>	1,4 10 <sup>-19</sup>	V <sup>-1</sup>	Konstante (Wobrauschk)

Tabelle 2: Bezeichnung der Variablen für Formel 8, Formel 7, Formel 6

### 1.3.2 Charakteristische Strahlung

Von Interesse in der Röntgenfluoreszenzanalyse (RFA) ist die Charakteristische Strahlung, die durch atomare Stoßprozesse erzeugt wird. Durch einen Stoß wird ausreichend Energie auf ein Hüllenelektron übertragen, um die Bindungsenergie zu überschreiten und ein Elektron aus der Hülle zu schlagen. Dieser Vorgang wird als Ionisation bezeichnet und ist in Abbildung 5 dargestellt.

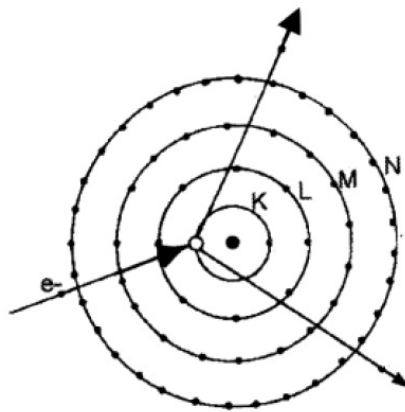


Abbildung 5 : Darstellung des Photoeffekts und dessen Folgereaktionen (Krieger and Petzold, 1988)

Werden die Elektronen der inneren Schale durch ausreichend Energie ionisiert, so treten Folgereaktionen auf, die die entstandene Elektronen-Lücke füllen. Für das Schließen der Lücke gibt es zwei rivalisierende Prozesse:

- Auger- Effekt
- Fluoreszenz- Effekt

#### 1.3.2.1 Fluoreszenz- Effekt

Bei Kernladungszahlen die größer als 30 sind, wird die frei gewordene Elektronen-Lücke aufgrund ihrer niedrigen energetischen Lage durch ein Elektron aus einem höheren Energieniveau besetzt. Die dabei freiwerdende Energie wird überwiegend in Form eines Photons abgegeben (siehe Abbildung 6). Diese Strahlung wird als Fluoreszenzstrahlung bezeichnet.

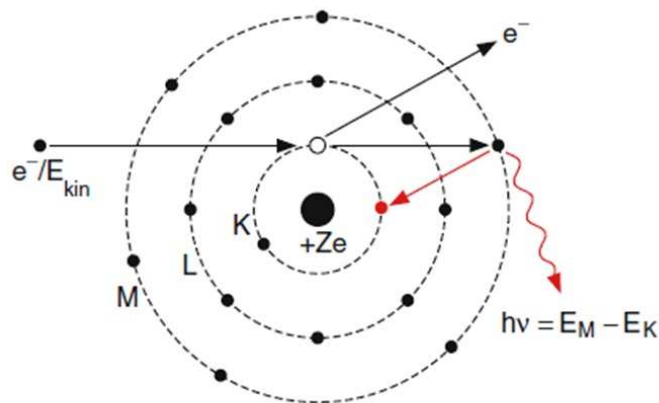


Abbildung 6: Schematische Darstellung des Fluoreszenz- Effekts (Demtröder, 2009)

Die freigeworden Energie vom Übergang eines Elektrons aus der L- auf die K- Schale lässt sich folgendermaßen berechnen:

$$E_{kin}^{Ph} = E_K - E_L$$

Formel 9

Abkürzung	Bezeichnung
$E_{kin}^{Ph}$	Bewegungsenergie des Photons
$E_L$	Elektronenenergie in der L- Schale
$E_K$	Elektronenenergie in der K- Schale

Tabelle 3: Bezeichnung der Variablen für Formel 9

Bei Übergängen der Elektronen zwischen den Schalen müssen die Auswahlregeln erfüllt sein

$$\Delta n \neq 0$$

$$\Delta l = \pm 1$$

$$\Delta j = 0, \pm 1$$

	Wert	Bezeichnung
<b>n</b>	1,2,3...	Hauptquantenzahl
<b>l</b>	$l=0,1,2,...,n-1$	Bahndrehimpulsquantenzahl
<b>J</b>	$1, \pm 1/2$	Gesamtimpuls

Tabelle 4: Bezeichnung der Variablen

### 1.3.2.2 Auger- Effekt

Beim Auger- Effekt (dominiert bei  $Z < 30$ ) wird die freigewordene Elektronen-Lücke durch ein Elektron aus einem höheren Energieniveau besetzt. Die dabei frei werdende Energie wird auf ein Elektron in der Schale übertragen, welches das Atom als Auger- Elektron verlässt (siehe Abbildung 7).

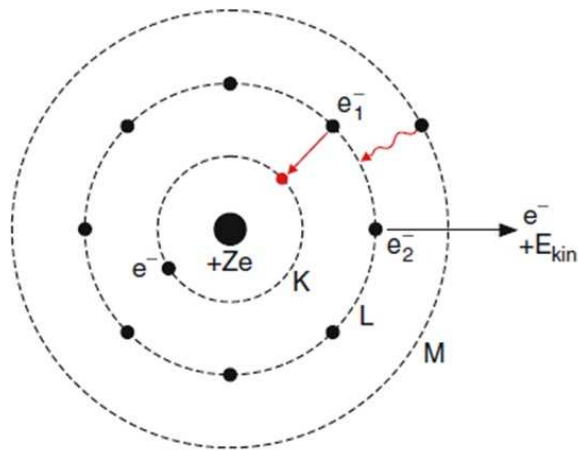


Abbildung 7: Darstellung des Auger-Effekts (Demtröder, 2009)

## 1.4 Wechselwirkung von Strahlung mit Materie

Die im vorigen Kapitel beschriebene Brems- und Charakteristische Strahlung wird als Röntgenstrahlung bezeichnet. Trifft Röntgenstrahlung auf ein Material, so kommt es zu weiteren Wechselwirkungen, die in diesem Kapitel beschrieben werden.

### 1.4.1 Photoeffekt

Kommt es zu einem Stoß zwischen einem gebunden Elektron und einem Photon, so kann das Elektron, unter der Bedingung, dass die Photonenenergie größer ist als die Bindungsenergie, die gesamte Energie absorbieren. Abhängig von der Differenz zwischen Photonenenergie und Elektronenbindungsenergie, kann das Elektron von der K- Schale in eine höhere Schale oder sogar ins Kontinuum gehoben werden. Diese Elektronen werden als Photoelektronen bezeichnet.

$$E_{kin}^{Ph} = E_{kin}^T - E_k - W$$

Formel 10

Abkürzung	Bezeichnung
$E_{kin}^T$	Energie des einfallenden Teilchens
$W$	Austrittsarbeit

Tabelle 5: Bezeichnung der Variablen für Formel 10

Die frei gewordenen Elektronen Plätze müssen wieder nachbesetzt werden. Die Folgereaktionen sind identisch wie in Kapitel Charakteristische Strahlung auf Seite 11.

### 1.4.2 Elastische Streuung

Wird das Photon an einem Elektron gestreut, welches im Atom gebunden ist, so spricht man von elastischer bzw. Rayleigh- Streuung. Das Photon ändert lediglich seine Richtung, die Energie und die Phasenlage zwischen einlaufendem und gestreutem Photon bleiben gleich. Im Atom werden die Elektronen lediglich zu einer gedämpften Schwingung angeregt, es kommt jedoch zu keiner Ionisierung bzw. Streuung.

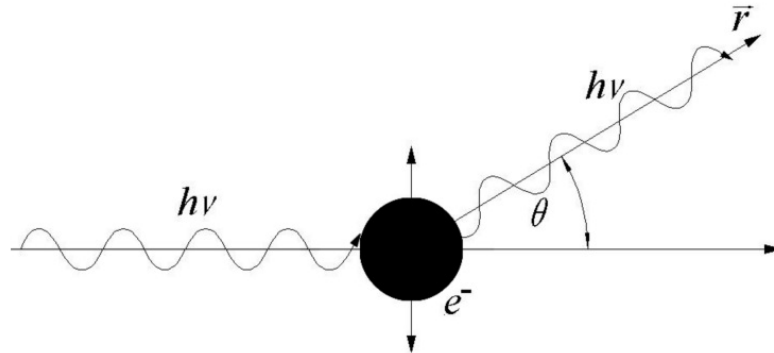


Abbildung 8: Darstellung der elastischen Streuung (Marosi, 2002)

Der differentielle elastische Streuquerschnitt für unpolarisierte Photonen ist gegeben durch:

$$\frac{\partial \sigma_{el}(\theta, E)}{\partial \Omega} = \frac{r_0^2}{2} \cdot (1 + \cos^2 \theta) \cdot |F(x, Z)|^2$$

Formel 11

$$x = \frac{\sin \theta}{2} \cdot \frac{1}{\lambda}$$

Formel 12

Abkürzung	Bezeichnung
$r_0$	Energie des einfallenden Teilchens
$\theta$	Streuwinkel (Winkel zwischen einfallenden und gestreuter Strahlung)
$\frac{1 + \cos^2 \theta}{2}$	Polarisationsfaktor (gibt den Grad der Polarisation der gestreuten Winkel an )
$F(x, Z)$	Formfaktor der sich aus der Integration über die Ladungsverteilung der Z Elektronen ergibt. Es wird berücksichtigt, dass alle Elektronen im realen Atom zur Streuung beitragen

Tabelle 6: Bezeichnung der Variablen für Formel 11 und Formel 12



### 1.4.3 Inelastische Streuung

Anstelle inelastischer Streuung kann auch als inkohärente- oder Compton- Streuung bezeichnet werden. Dieser Prozess bezeichnet die Wechselwirkung mit einem äußeren, nur schwach gebundenen Elektron. Die Energie und der Impuls des Photons werden teilweise auf das schwach gebundene Elektron übertragen. Ist die Bindungsenergie des Elektrons geringer als die übertragene Energie des Photons, so kann das Elektron das Atom verlassen (Compton Elektron). Das gestreute Photon ändert dabei seine Richtung, seine Energie und seinen Impuls.

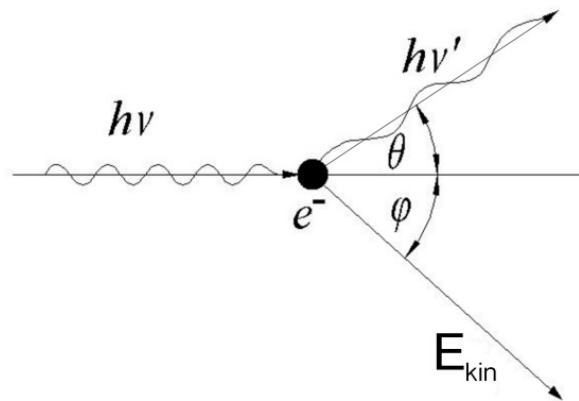


Abbildung 9: Darstellung der inelastischen Streuung (Marosi, 2002)

Die Energie nach dem inelastischen Stoß des Photons lässt sich folgendermaßen berechnen:

$$E_{Phf} = \frac{E_{Phi}}{1 + \frac{E_{Phi}}{m_e \cdot c_0^2} \cdot (1 - \cos\theta)}$$

Formel 13

Abkürzung	Bezeichnung
$E_{Phi}$	Energie des einfallenden Photons
$E_{Phf}$	Energie des gestreuten Photons
$m_e$	Masse des Photons

Tabelle 7: Bezeichnung der Variablen für Formel 13

Aus Formel 13 ist ersichtlich, dass bei kleineren Streuwinkel des Photons, die Energieänderung geringer wird, und so auch die Erzeugung von Compton Elektronen.

## 1.5 Schwächung der Strahlung

Tritt eine Strahlung durch ein Medium, so entstehen Wechselwirkungsprozesse die zu einer Abnahme der Intensität führen. Dies ist in Abbildung 10 dargestellt. Das Beer Lambert'sche Gesetz (siehe Formel 14) beschreibt den Intensitätsverlust von Photonen durch eine Materialschicht.

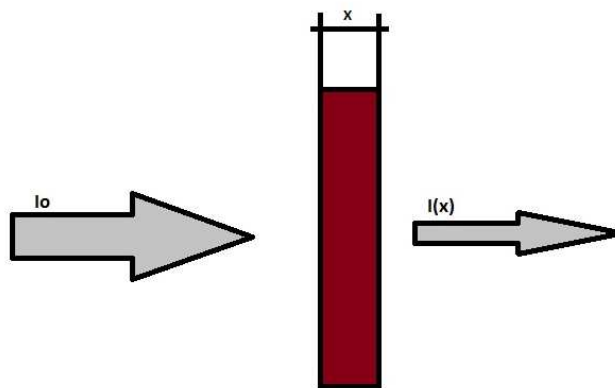


Abbildung 10: Intensitätsabnahme, wenn Strahlung durch einen Körper tritt

$$I(x) = I_0 \cdot e^{-\mu(E) \cdot x} = I_0 \cdot e^{-\left(\frac{\mu_m}{\rho}\right) \cdot \frac{m}{F}}$$

Formel 14

$$m = \rho \cdot F \cdot x$$

Formel 15

Abkürzung	Einheit	Bezeichnung
$\mu(E)$	$\text{cm}^{-1}$	linearer Schwächungskoeffizient
$\mu_m/\rho$	$\text{cm}^2/\text{g}$	Massenschwächungskoeffizient
$x$	g	Masse der Materialschicht
$F$	$\text{cm}^2$	Fläche der Materialschicht
$\rho$	$\text{kg}/\text{m}^3$	Dichte

Tabelle 8: Bezeichnung der Variablen für Formel 14 und Formel 15

Der Massenschwächungskoeffizient  $\mu_m$  ergibt sich auch dem Verhältnis des linearen Schwächungs- Koeffizienten  $\mu$  zur Dichte  $\rho$ . Dieser Schwächungskoeffizient besteht wiederum aus dem Beitrag  $\tau$  für den photoelektrischen,  $\sigma_{el}$  für den elastischen und  $\sigma_{in}$  für den inelastischen Streukoeffizienten(Wobrauschek, 2012).

Der Intensitätsverlust ist abhängig von der Dichte des absorbierenden Materials, der chemischen Zusammensetzung und der Energie der Strahlung. Ist die Energie ausreichend um ein Atom zu ionisieren, so steigt der Massenschwächungskoeffizient sprunghaft an. Mit steigender Energie werden auch die inneren Schalen ionisiert (siehe Abbildung 11). Diese Sprünge werden als Absorptionskanten bezeichnet.

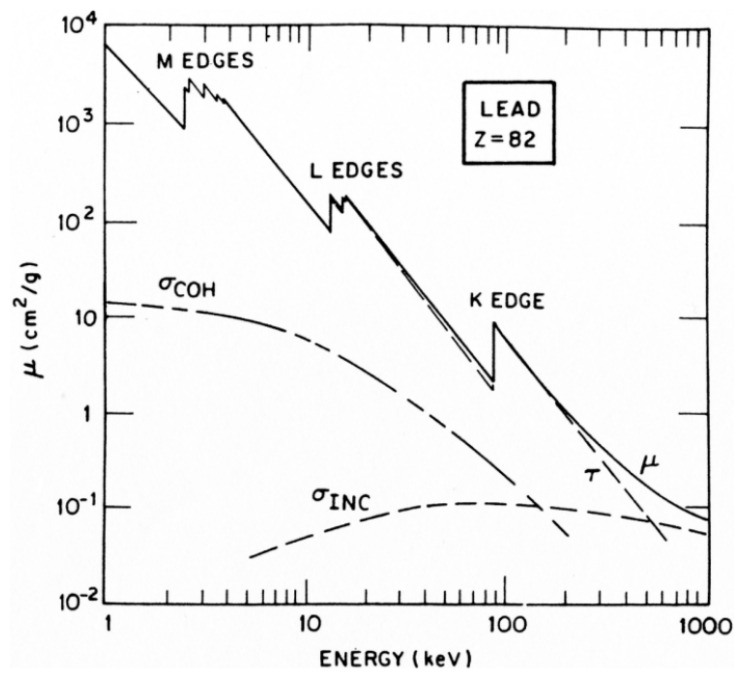


Abbildung 11: Absorptionskanten von Blei (Jenkins et al., 1981)

## 2 Röntgenfluoreszenzanalyse

Die Röntgenfluoreszenzanalyse (RFA, eng XRF) ist ein spektroskopisches Verfahren zur Elementanalyse von unbekannten Proben.

Dieses Verfahren ermöglicht neben dem flexiblen Aufbau eine Nachweisgrenze für Spurenelemente bis in den  $\mu\text{g/g}$  Bereich ( $1:10^6$ ). Sie wird für die zerstörungsfreie Untersuchung von Silizium-Halbleitern oder Stählen verwendet.

Die RFA führt bei der Untersuchung von lebendem Gewebe jedoch zu DNA- und Protein-Schäden.

Atome der Probe werden durch ionisierende Strahlung zur Emission von Fluoreszenz- Strahlung angeregt. Diese für jedes Element eindeutige Strahlung wird mittels wellenlängendispersiven oder energiedispersiven Methoden gemessen und anschließend analysiert.

Das Gesetz von Moseley beschreibt den Zusammenhang der Wellenlänge von charakteristischen Strahlungen und der Ordnungszahl  $Z$ , damit ist qualitative Analyse möglich. Da die Intensität der charakteristischen Strahlung von der Konzentration der Probe abhängt, ist eine quantitative Analyse ebenfalls durchführbar.

Aufgrund der schlechten Fluoreszenzausbeute und der starken Absorption leichter Elemente, eignet sich die RFA besser für die Elemente mit höheren Ordnungszahlen ( $_{6}\text{C}$  bis  $_{92}\text{U}$ ).

Für die energiedispersive Röntgenfluoreszenzanalyse werden Detektoren verwendet, welche die Photonenenergie erfassen und diese am Detektorausgang darstellen. Als EDRFA Detektoren werden Silizium –Lithium Detektoren, Germanium Detektoren und Silizium –Drift Detektoren, die im nachfolgenden Kapitel erklärt werden, verwendet.

### 2.1 Instrumentierung

Dieses Kapitel beschäftigt sich mit den technischen Geräten und ihren Funktionen, die für die Methode der EDRFA benötigt werden.

Für den einfachsten Aufbau eines EDRFA – Systems wird eine Strahlenquelle (Röntgenröhre, Radioisotop,...) für die Anregung der Probe benötigt. Die angeregte Probe emittiert Fluoreszenzstrahlung, die durch einen energiedispersiven Detektor registriert und anschließend elektronisch verarbeitet wird. Die Rohsignale, werden durch einen MCA (Multi Channel Analyzer) nach ihrer Energie sortiert und als Spektrum am PC dargestellt.

#### 2.1.1 Erzeugung der Röntgenstrahlung.

Für die Erzeugung von Röntgenstrahlung wird in den meisten Fällen eine Röntgenröhre eingesetzt. Die Röntgenröhre besteht aus einem hochevakuierten Glaszylinder in dem sich eine Kathode und eine Anode gegenüberstehen. An der Kathode werden Elektronen durch einen Wolfram Glühfaden erzeugt, die anschließend durch einen Wehneltzylinder gebündelt werden. Durch das E- Feld zwischen Kathode und Anode liegt, werden die Elektronen Richtung Anode beschleunigt. Werden die Elektronen an der Anode abgebremst, so kommt es zur Bremsstrahlung und Charakteristischen Strahlung, wie im Kapitel Wechselwirkung von Elektronen mit Materie

auf Seite 9 beschrieben ist. Damit die Strahlung nicht allzu sehr geschwächt wird befindet sich im Glaszylinder ein 100-300µm dickes Beryllium-Fenster. Beryllium besitzt ausreichend mechanische Stärke um das Vakuum im Zylinder aufrecht zu erhalten und zugleich einen möglichst geringen Anteil der Röntgenstrahlung zu absorbieren. Durch Elektronenbeschuss, gestreuter Elektronen, des Berylliumfensters kann es erodieren und sogar brechen, was unerwünscht ist.

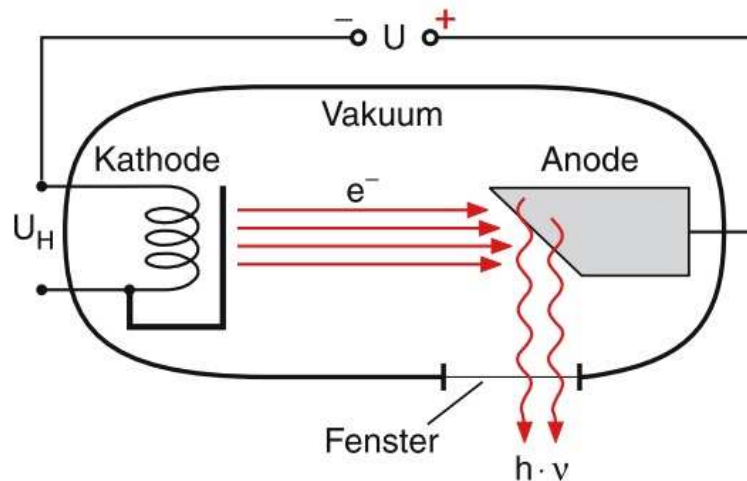


Abbildung 12: Schematische Darstellung einer Röntgenröhre (Demtröder, 2009)

Die Hochspannung zwischen Anode und Kathode, welche das E-Feld erzeugt, bewegt sich im keV-Bereich, wobei bei Standardröhren die Anode auf Erdpotential liegt. Die Energie der Elektronen berechnet sich aus der Beschleunigungsspannung (in kV) und der Elementarladung des Elektrons (siehe Formel 16). Die Einheit wird in keV angegeben.

$$E_{kin} = e \cdot U$$

Formel 16

Röntgenröhren besitzen einen Wirkungsgrad von etwa 1%. Für die Bestimmung des Wirkungsgrads wird die Leistung der Strahlung durch die dafür benötigte Energie dividiert.

Der größte Teil der Energie wird in Form von Wärme abgegeben. Um diese Energie abzuführen, ist eine Kühlung für die Anode notwendig. Für die Kühlung der Anode kann Wasser oder Luft (Drehanode) verwendet werden. Die Anode selbst muss gut wärmeleitend und hitzebeständig sein. Die der Kathode zugewandten Seite ist mit dem eigentlichen Anodenmaterial beschichtet. Die Anzahl der Elektronen, die auf die Anode treffen, werden durch die Geometrie des Wehneltzylinders und den Kathodenheizstrom, der sich in µA bis mA bewegt, geregelt.

### 2.1.2 Sekundärtarget

Um die Anregungsbedingung für die Proben zu verbessern und die Analysequalität zu erhöhen, werden Sekundärtargets verwendet, wie es in Abbildung 13 dargestellt ist.

Das Sekundärtarget soll so gewählt werden, dass das gesuchte Element optimal angeregt wird. Daher ist es empfehlenswert, dass das Sekundärtarget leicht getauscht werden kann um damit alle möglichen Elemente, die sich in der Probe befinden können, anzuregen. Es führt zu einer linearen Polarisierung der primären Röntgenstrahlung und eliminiert die Brems- sowie Charakteristische Strahlung der Röntgenröhre. Damit wird ein niedriger Hintergrund erzielt und eine quasi monochromatische Anregung ( $K_\alpha$   $K_\beta$  Strahlung des Targets) erzeugt.

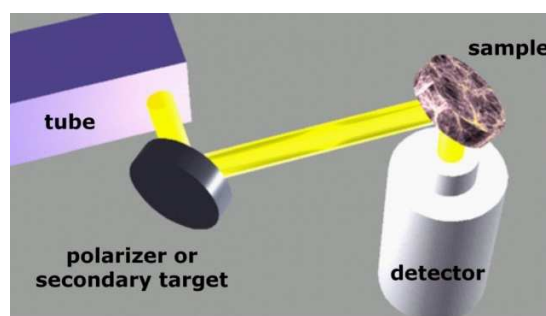


Abbildung 13: Schematische Darstellung der Versuchsanordnung mit Sekundärtarget (Cernohlawek, 2005)

### 2.1.3 Energiedispersive Detektoren

Energiedispersive Detektoren werden aus Halbleitermaterial hergestellt. Ionisierende Strahlung trifft auf den Detektor und erzeugt dabei Elektronenionen und ("Loch")- Paare im empfindlichen Volumen. Ein elektrisches Feld sammelt die Ladungsträger und erzeugt ein entsprechendes Spannungssignal. Dieses Signal ist proportional zur Anzahl der Ladungsträger und dadurch proportional zur Strahlungsenergie.

Halbleiter können als Leiter oder als Isolatoren betrachtet werden. Ihre Struktur ist kristallin und die Leitfähigkeit ist von der Temperatur abhängig. Bei niedrigeren Temperaturen ist die Leitfähigkeit geringer, wohingegen mit ansteigender Temperatur auch die Leitfähigkeit erhöht. Bei der Erklärung hilft das Bändermodell, wie es in Abbildung 14 dargestellt ist, mit Leitung und Valenzband.

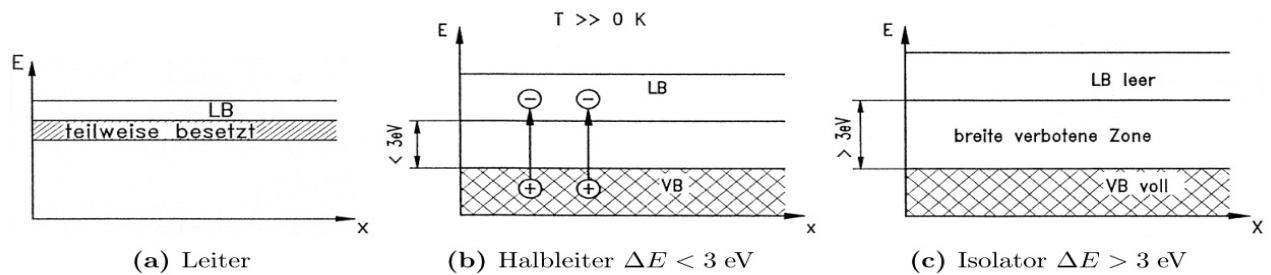


Abbildung 14: Bändermodell für Leiter, Halbleiter und Isolator (Küick, 2002)

#### 2.1.3.1 Leiter (a)

Bei Leitern überlagern sich beide Bänder. Die Elektronen des Valenzbandes befinden sich ebenfalls im Leitungsband, dies führt zu einer Leitfähigkeit der Elektronen.

#### 2.1.3.2 Halbleiter (b)

Bei Halbleitern hingegen überlappen sich die beiden Bänder. Bei Raumtemperatur ist allein durch die gegebenen Bedingungen keine Leitung möglich, da ein Abstand von ca. 1 eV zwischen beiden Bändern besteht.

#### 2.1.3.3 Isolator (c)

Bei Raumtemperatur beträgt der Abstand der Bänder ist größer 3 eV. Dieser Abstand kann nicht mehr überbrückt werden und es ist schwer möglich eine Leitfähigkeit herzustellen.

### 2.1.4 Aufbau von Halbleiter- Detektoren

Die Herstellung eines hochreinen Halbleiters ist sehr aufwendig. Aus diesem Grund werden Halbleiterkristalle mit anderen Atomen z.B. Li gezielt verunreinigt (dotiert), um die Isolationsfähigkeit des Halbleiters zu verbessern. Die Dotierung hat die Funktion freie Ladungsträger an sie zu binden. Dabei gibt es zwei Varianten:

#### 2.1.4.1 n-Leiter

Für die Produktion eines Überschuss- oder n-Leiters wird der Silizium Kristall mit einem Element der fünften Hauptgruppe verunreinigt. Die Fremdatome werden als Akzeptoren bezeichnet. Diese Elemente besitzen um ein Valenzelektron mehr als Silizium und ermöglichen dem Elektron sich frei im Kristallgitter zu bewegen.

#### 2.1.4.2 p-Leiter

Im Vergleich zum n Leiter wird der Silizium- Kristall mit einem Element der dritten Hauptgruppe, das ein Valenzelektron weniger als Silizium besitzt, verunreinigt. Im Valenzband

fehlt ein Elektron und es entsteht ein Loch. Die Löcher können sich im Atomgitter „weiterbewegen“, da die fehlenden Bindungs- Elektronen von anderen ersetzt werden können. Die Fremdatome werden als Akzeptoren bezeichnet.

Mit Ausnahme des Ge(HP)- Detektors sind Halbleiterdetektoren aus Dioden aufgebaut. Dioden besitzen typischerweise eine Sperrschicht zwischen dem Übergang von zwei Halbleiterarten.

### 2.1.4.3 Die p-n Struktur

Für eine p-n Struktur werden die vorher erwähnten p- und n- Leiter mit dem gleichen Grundgitter in Kontakt gebracht. Die unterschiedliche Ladungsträgerdichte in den p- und n- Bereichen lässt die Elektronen aus dem n Teil in den p Teil wandern und die Löcher begeben sich in den n-Teil. Das durch die Ladungswanderung entstandene elektrische Feld, das vom n- zum p- Leiter gerichtet ist, begrenzt einen weiteren Ladungsaustausch. An der Übergangsschicht zwischen p- und n- Leiter stellt sich ein Ladungsgleichgewicht ein.

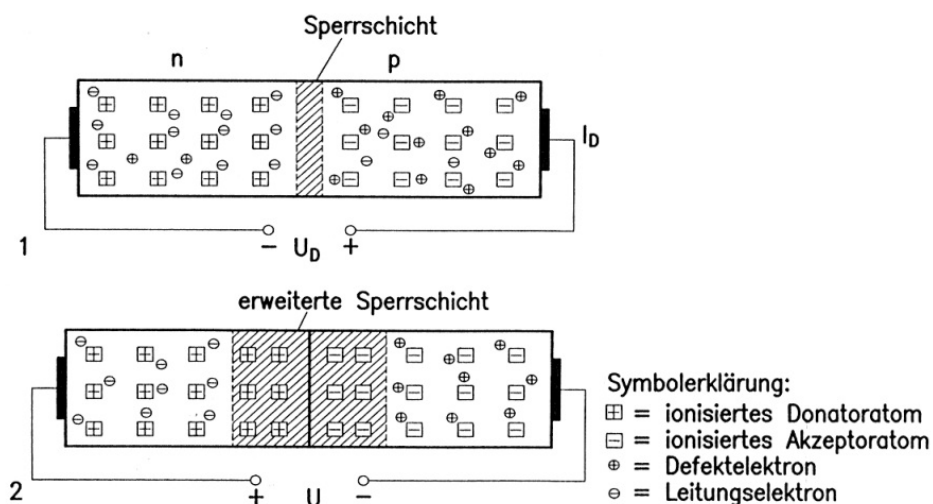


Abbildung 15: p-n Übergang mit Polung in Durchlassrichtung (1) und in Sperrrichtung (2) (Kück, 2002)

Um eine größere Sperrschicht zu erzeugen werden PIN (*positive intrinsic negative diode*) Strukturen verwendet. Bei diesen Strukturen befindet sich zwischen p- und n-Schicht eine dritte Schicht, die sogenannte *intrinsic-(i)- Schicht*. Diese Schicht ist ein Halbleiter der undotiert ist und daher nur eigenleitend. Die daraus ergebende Vergrößerung der Sperrschicht führt zu einer höheren Strahlenempfindlichkeit. Wird die *intrinsic-(i)- Schicht* durch einen schwach dotierten Halbleiter ersetzt, so handelt es sich um eine PSN Struktur. Diese wird bei SD Detektoren verwendet



#### 2.1.4.4 Silizium- Lithium Detektor

Die vorhin erwähneter PIN Struktur findet beim Silizium-PIN Detektor Anwendung. Der Silizium-Kristall, wie in Abbildung 16 dargestellt, besitzt einen Querschnitt zwischen 10 und 80 mm<sup>2</sup> und eine Dicke von 3-5mm. In der Sperrrichtung wird eine Spannung von 500- 1000 V angelegt.

Für die Kontaktierung der p- und n-Schicht wird eine 200 Angström dicke Goldschicht aufgedampft. Um Photonen zu detektieren, muss erst die Goldschicht durchdrungen werden, dabei wird vorwiegend niederenergetische Strahlung absorbiert. Erreichen die Photonen das sensitive Volumen, so werden Elektronen-Loch-Paare erzeugt. Die dabei frei gesetzten werdenden Elektronen können gemessen werden und geben Auskunft über die Photonenenergie. Mit flüssigem Stickstoff wird der Detektor auf 77K gekühlt, um den Leck- Strom im Si- Kristall und das Rauschen des Detektors zu reduzieren.

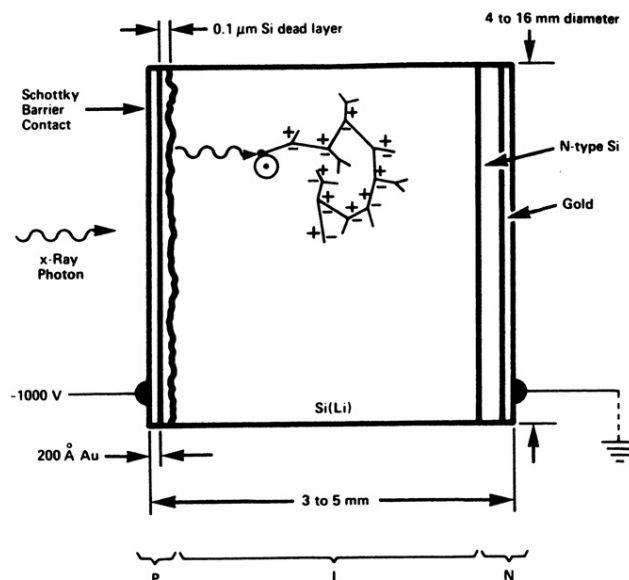


Abbildung 16: Schematische Darstellung des SI(Li) Detektors (Jenkins et al., 1991)

#### 2.1.4.5 Silizium-Drift Detektor

Moderne SD Detektoren bieten den Vorteil, dass sie nicht mehr auf 77K gekühlt werden müssen und mehr Photonen bei gleicher Fläche in gleicher Zeit detektieren können.

Der SDD ist in einer PEN Struktur aufgebaut. Der dünne Aufbau (300µm - 450µm) Detektor bietet den Vorteil, dass bereits Spannungen von 70Volt in Sperrrichtung ausreichen, um den Detektor zu betreiben. Daher ist eine Aluminium-Kontaktierung ausreichend, die durchlässiger für niederenergetische Strahlung ist. Die Kühlung erfolgt über ein Peltier-Element.

Um die Leistungsfähigkeit des SD Detektors zu steigern, wird die Sperrschicht mit drei Kontakten seitlich ausgeweitet. Diese Veränderung nennt sich Seitwärtsdepilation. Der schwach

n-dotierte Silizium-Kristall wird mit zwei großen p- und einem kleinen n-Kontakt bestückt, wie in Abbildung 17 dargestellt. Die negative Spannung wird an den beiden n- Schichten so lange erhöht, bis sich die beiden Sperrschichten in der Mitte berühren.

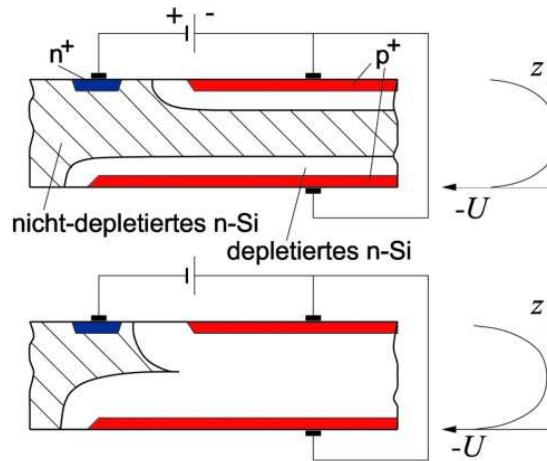


Abbildung 17: Schematische Darstellung des Prinzips der Seitwärtsdepletion (Pahlke, 2004)

Elektronen, die aufgrund von Strahlung entstehen, müssen durch ein elektrisches Feld zur Anode geleitet werden. Aus diesem Grund ist die p-Schicht, auf der sich die Anode befindet, in Form von Ringen aufgebaut, wie Abbildung 19 zeigt. Je geringer der Radius der Ringe, desto höher ist die negative Spannung am Ring. Dies erzeugt ein elektrisches Feld entlang der schwach n-dotierten Schicht in Richtung des Detektormittelpunktes, wo sich die Anode mit kleiner Kapazität befindet. Das entstandene E-Feld wird als Driftfeld bezeichnet (siehe Abbildung 18).

Trifft nun ein Photon auf die leicht n-dotierte Schicht in der Mitte, so werden Elektronen erzeugt, die durch das Driftfeld zur Anode bewegt werden. Um das elektrische Rauschen, erzeugt durch Kapazitäten und Wärme, zu verringern, ist in der Anode ein Verstärker integriert.

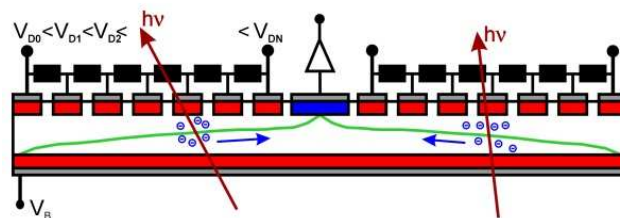


Abbildung 18: Seitliche schematische Darstellung eines SD Detektors mit Driftfeld (Pemmer, 2007)

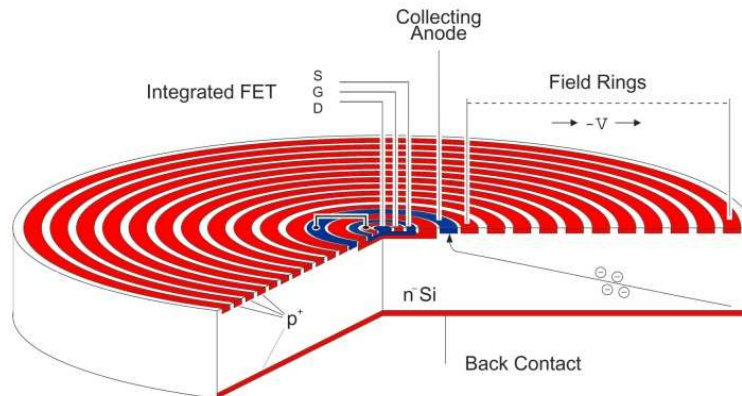


Abbildung 19: dreidimensionale Darstellung eines SD Detektors mit integrierten FET(Eggert, 2004)

## 2.2 Signalverarbeitung

Das Signal am Detektorausgang wird durch weitere Schritte derart verarbeitet, dass schließlich ein Spektrum mit der Anzahl der Photonen und der dazugehörigen Energie dargestellt werden kann. Dabei durchläuft das Signal einige Verarbeitungsschritte, wie sie in Abbildung 20 dargestellt sind.

Nach der Verstärkung im Detektor erfolgt neuerlich eine Verstärkung außerhalb. Dabei wird durch einen Bandpass (Hoch und Tiefpass) das „Signal to Noise Ratio“ verbessert. Das Signal wird weitergeleitet zum Shaper, der das Stufensignal in ein gaußförmiges Signal umwandelt.

Zwei zeitlich knapp aufeinanderfolgende Pulse können bei der weiteren Signalverarbeitung falsch interpretiert und als Summen-Peak dargestellt werden. Um dies zu verhindern, wird nach dem Eintreffen eines Signals für eine definierte Zeit vom Pulse-Pileup- Rejector kein Signal durchgelassen. Die Zeit, in der kein Signal durchgelassen wird, wird als Totzeit bezeichnet.

Anschließend folgt ein Diskriminator, der wie der Verstärker das Signal to Noise Ratio verbessert.

Nun erfolgt die digitale Bearbeitung: Dafür wird das Signal durch einen Analog/Digital-Konverter (ADC) in ein Digitalsignal umgewandelt. Der Multi- Channel-Analyser (MCA) ordnet die Signale nach ihrer Größe in unterschiedliche Kanäle. Es ergibt sich ein Spektrum, das nach Kanal-Nummern sortiert ist. Durch Energiekalibrierung werden den Kanälen Energien zugeordnet.

Hersteller von energiedispersiven Detektoren bieten bereits Einheiten an, in denen die gesamte Signalverarbeitung integriert ist.

Im Computer wird das digitale Signal als Spektrum dargestellt. Die X-Achse zeigt die Kanalnummer und die Y-Achse die Anzahl der detektierten Signale. Durch Energiekalibrierung, wird jeder Kanal-Nummer ein Energiewert zugeordnet.

Dekonfluenzprogramme reduzieren softwaretechnisch das „Signal to Noise Ratio“ und erkennen Escape-Peaks und andere Detektorartefakte.

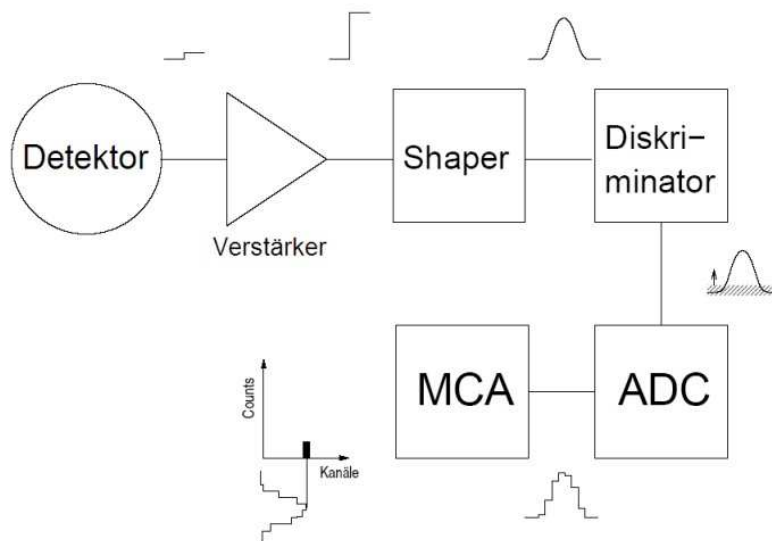


Abbildung 20: Signalverarbeitungskette einer EDRFA Mess- Systems (Eggert, 2004)

## 2.3 Detektorartefakte

Kommt es zu Wechselwirkungen im Detektor, die nicht direkt absorbierten Photonen zugeordnet werden, so spricht man von Detektorartefakten. Sie bergen das Risiko, das Spektrum falsch zu interpretieren. Häufige Detektorartefakte sind der Escape- Peak, der Summen-Peak, und die Compton-Kante, die in diesem Kapitel genauer erklärt werden.

### 2.3.1 Escape Peak

Tritt der Photoeffekt im sensitiven Volumen des Detektors auf, so werden die Atome des Detektors angeregt und emittieren Fluoreszenzphotonen, wie bereits im Kapitel 1.3.2.1 beschrieben. Diese Photonen werden im Idealfall neuerlich im Detektor absorbiert. Die Summe der absorbierten Photonen verhält sich proportional zur Energie des primären Photons. Verlässt jedoch das Fluoreszenzphoton den Detektor, so entsteht ein Peak, der um die  $K_\alpha$  Energie des Detektormaterials zur niedrigeren Energie verschoben ist. Bei einem Silizium-Detektor befindet sich der Escape- Peak 1,74KeV unterhalb des ursprünglichen Peaks.

### 2.3.2 Summen- Peaks

Wenn zwei Signale innerhalb der Breite des Pulses kommen können sie vom Pulse Pile Rejector akzeptiert werden und zu einem Signal aufsummiert werden. Der Summen-Peak entsteht an der Summenenergie dieser beiden nicht unterschiedenen Photonen. Haben die beiden Photonen dieselbe Energie, dann liegt der Summen-Peak bei der doppelten Energie.

### 2.3.3 Compton Kante

Kommt es zu einer inelastischen Streuung eines hochenergetischen Photons um  $180^\circ$ , so gelangt ein Compton- Elektron in den Detektor und gibt dort die Energie ab.

### 3 Gelelektrophorese

Die Gelelektrophorese ist eine analytische Methode, die verwendet wird um Moleküle z.B. Proteine zu trennen. in einem elektrischen Feld werden geladene Moleküle durch ein Gel, das als Molekularsieb wirkt, bewegt.

Abhängig von der Ladung und der Größe der Moleküle bewegen sie sich mit unterschiedlicher Geschwindigkeit durch die Gel-Matrix. Die Wandergeschwindigkeit der Moleküle ist bei der Elektrophorese proportional zur Feldstärke zwischen der Kathode und der Anode, die sich auf beiden Seiten des Gels befinden, und der Ionenladung  $Q$  der Moleküle, wie man in Abbildung 21 sehen kann. Die Geschwindigkeit verringert sich bei großen Molekülen oder großer Viskosität. Dies führt zu einer Auftrennung abhängig von der elektrophoretischen Mobilität.

Die Porenweite des Gels hat Auswirkung auf die Auftrennung nach der Molekülgröße. So hemmt eine kleine Porenweite die Wandergeschwindigkeit von großen Molekülen. Abhängig von der Anwendung werden unterschiedliche Gele verwendet. So wird Agarose-Gel (Porengröße = 150 - 500 nm) zur Auftrennung von DNA- Fragmenten eingesetzt, während Proteine in Polyacrylamid-Gele (Porengröße = 3-6nm) aufgetrennt werden. Polyacrylamid oder auch  $C_3H_5NO$  besitzt Elemente, mit niedriger Ordnungszahl.

Für die Auftrennung von Proteinen werden Verfahren wie die SDS Page (sodium dodecyl sulfate polyacrylamide gel electrophoresis) und der Western Plot verwendet.

Metalloproteine nehmen eine wichtige Rolle sowohl in physiologischen als auch pathologischen Prozessen ein und werden ebenfalls mit Gelelektrophorese aufgetrennt. Metalloproteine beinhalten ein oder mehrere Metalle. Befinden sich diese Proteine mit ihren Metallen im Elektrophorese Gel, so sollten sie nachweisbar sein, wenn ausreichend Metalle vorhanden sind.(Rehm and Letzel, 2010)

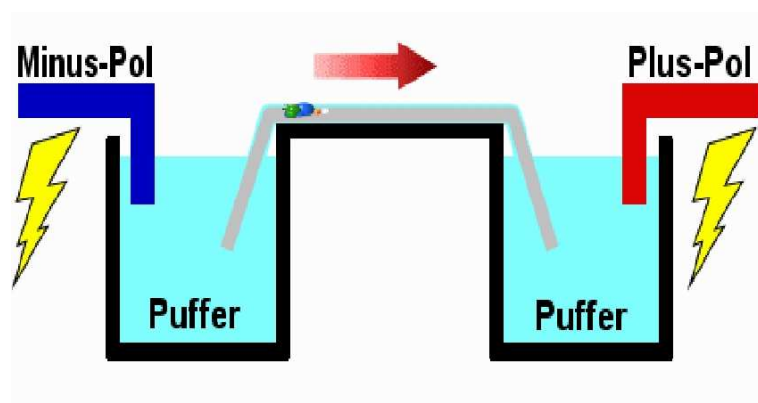


Abbildung 21: Prinzip der Gelelektrophorese (Hübl and Milosavljevic, 2011)

## 4 Messaufbau

Um zweidimensionale biologische Proben nach unterschiedlichen Elementen im Millimeterbereich zu scannen, wurde ein Messaufbau entwickelt, der sich durch seine Flexibilität und Automatismus auszeichnet. Dabei wurde eine Röntgenfluoreszenz Messanlage mit einem PC gesteuerten Koordinatentisch kombiniert. In einer dafür entwickelten Software werden die beiden Komponenten aufeinander abgestimmt bedient.

### 4.1 Röntgenfluoreszenz Messanlage

Für die Messung von Röntgenfluoreszenz im Gel benötigt man folgende Komponenten

- eine Strahlenquelle
- ein Sekundärtarget mit Kollimator
- einen Detektor
- Signalverarbeitungssoftware und Hardware

Dieses Kapitel soll Einblick in die konkrete Abstimmung der Komponenten geben.

#### 4.1.1 Strahlenquelle für die Anregung

Röntgenstrahlung wird durch eine Panalytical Röntgenröhre in einer Röntgenanlage der Firma Siemens Model 5300 produziert. Dabei handelt es sich um eine wassergekühlte Anlage, die zeitweise mit einer Leistung von 2,2kW betrieben werden kann. Bei der Long Fine Focus Röntgenröhre des Typs PW2276720 von Panalytical ist die Anode mit Silber (Ag) beschichtet. In Abbildung 22 ist dargestellt mit welchem Verhältnis die Beschleunigungsspannung zum Röhrenstrom betrieben werden kann. Tabelle 9 zeigt die maximal zulässigen Betriebsströme und Spannungen. Der 2 Kilowatt Bereich wurde nie überschritten. Die Röntgenstrahlung verlässt die Röhre durch ein 200µm dickes Beryllium Fenster und einer Lochblende. Die Ag- Röhre strahlt Brems- und Charakteristische Strahlung aus. Da für unseren Messaufbau ein Sekundärtarget verwendet wird, ist hier die Energie der Charakteristischen Strahlung für Ag von besonderem Interesse. Der  $K\alpha$ -Peak der Charakteristischen Strahlung beträgt 22,16 keV und der  $K\beta$ -Peak beträgt 24,94 keV.

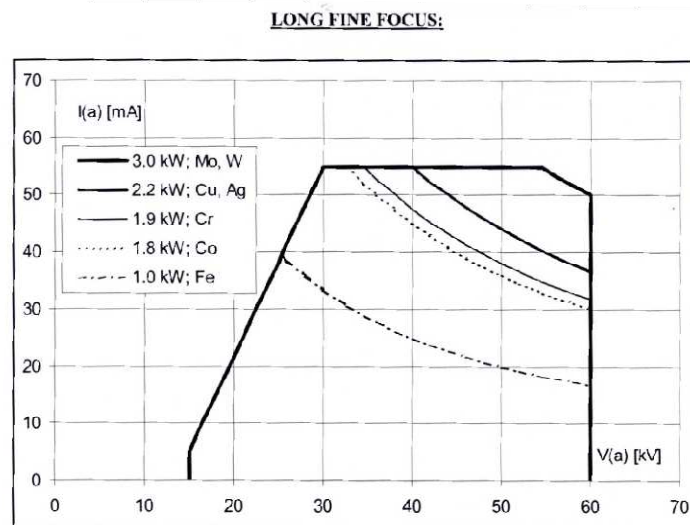


Abbildung 22: Grafische Darstellung der zulässigen Spannung für Long Fine Fochus Röntgenröhren von Panalytical (Tube, 2007)

2,2 kW Ag Röntgenröhre										
Spannung [kV]	15	20	25	30	35	40	45	50	55	60
Strom [mA]	5	22	38	55	55	55	38	33	40	37

Tabelle 9: maximal zulässige Strom und Spannungswerte der 2,2 kW Ag Röntgenröhre (Tube, 2007)

### 4.1.2 Sekundärtarget

Die Strahlung der Röntgenröhre trifft auf ein Sekundärtarget (siehe Abbildung 13) das  $45^\circ$  zu dem Röntgenstrahl angeordnet ist. Das Sekundärtarget besteht aus Molybdän und wird von der Ag-Röntgenröhre optimal angeregt, da die K-Kante bei 20keV liegt. Sie befindet sich unterhalb der Anregungsenergie der Charakteristischen Strahlung der Ag-Röntgenröhre. Da der Strahl nach dem Austritt vom Sekundärtarget die Richtung um  $45^\circ$  verändert, entsteht eine Gesamtumlenkung von  $90^\circ$ , die zu einer Polarisierung der kohärenteren Streuung führt. Dies bietet den Vorteil, dass die Streuung auf der Probe reduziert wird. Durch die Gesamtbrechung von  $90^\circ$  wird die Strahlung der Röntgenröhre reduziert, es trifft nur mehr Charakteristische Strahlung des Sekundärtargets auf die Probe.



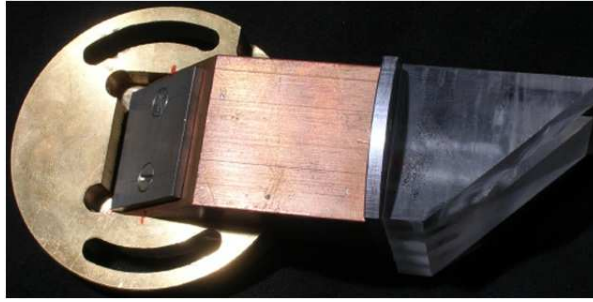


Abbildung 23: Strahlengang mit Sekundärtarget

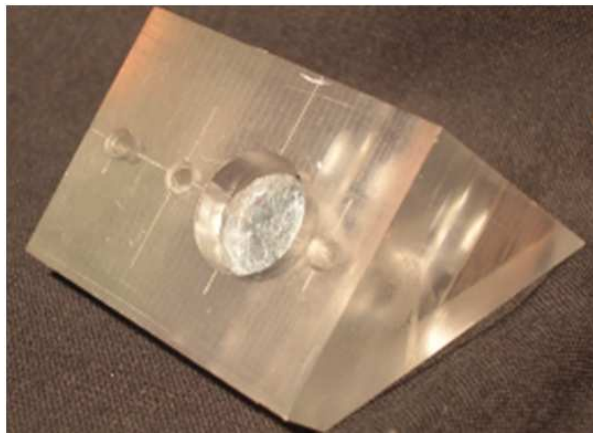


Abbildung 24 : PMMA Block um die Geometrie zwischen Detektor und Probe festzulegen (Pemmer, 2007)

Der Messaufbau, auf dem sich das Sekundärtarget befindet besteht aus reinem Kupfer (siehe Abbildung 23) und ist mit der Röntgenanlage verschraubt. Im Strahlengang wurde ein Polymethylmethacrylat [PMMA] Röhrchen mit einem Innendurchmesser von 5 mm eingesetzt. Der Grund dafür ist, dass PMMA eine geringe Ordnungszahl ( $Z \approx 6$ ) besitzt und daher bei einer Anregung kaum Photonen emittiert. Am Ende des Strahlenganges befindet sich ein dreieckförmiger PMMA Block (siehe Abbildung 24) der den Winkel zwischen Strahlengang, Probe und emittierten Photonen festlegt.

#### 4.1.3 Detektor zum Erfassen der Röntgenfluoreszenz

Unter  $45^\circ$  zu der Probe befindet sich ein energiedispersiver SD Detektor, der durch einen verstellbaren Arm gehalten wird. Der Arm ist an der Röntgenröhre angebracht, damit ausreichend Platz zur Positionierung der Probe (Elektrophorese-Gel) gegeben ist.

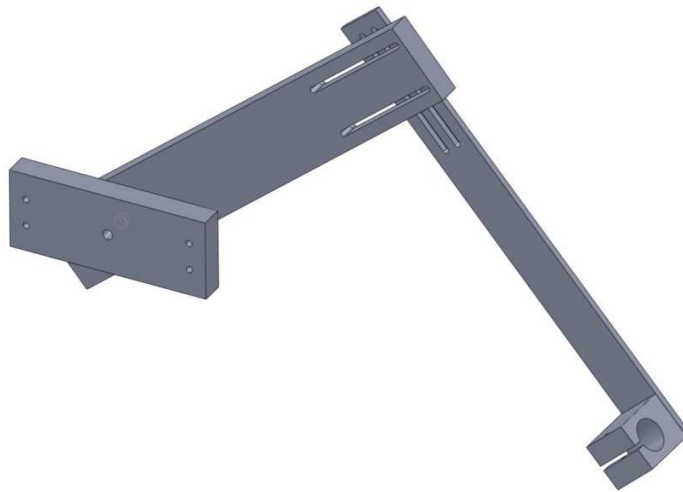


Abbildung 25: verstellbarer Detektorarm

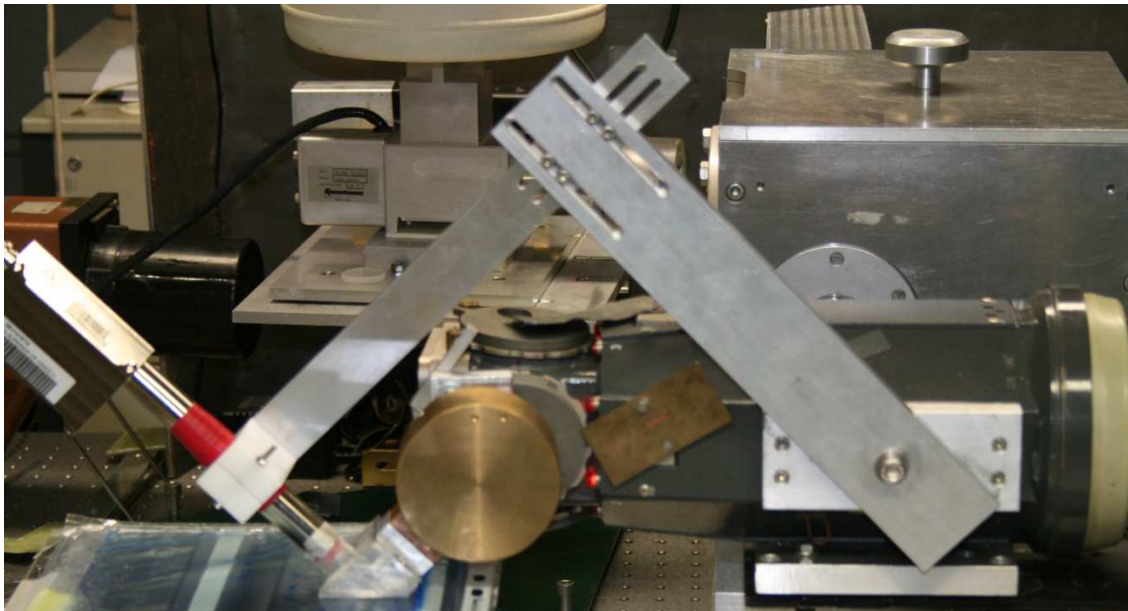


Abbildung 26: Röntgenröhre, Sekundärtarget und Detektor

Für die Anwendung wurden folgende SD Detektoren getestet:

- SD Detektor der Firma Amptek mit dem Modellnamen SuperSDD XR-100SDD X  
Detektorfläche:  $25 \text{ mm}^2$ / Kristalldicke :  $500 \text{ }\mu\text{m}$
- SD Detektor der Firma Ketek mit dem Modellnamen AXAS  
Detektorfläche:  $5 \text{ mm}^2$ / Kristalldicke :  $300 \text{ }\mu\text{m}$

#### 4.1.4 Signalverarbeitung

Das Signal des Detektors wird durch ein geschirmtes Kabel an den AFT Research Amplifier der Firma Canberra weitergeleitet, wie in Abbildung 27 dargestellt ist. Diese Komponente kann das Signal des Detektors verstärken, Filtern und Shapen. Die Shaping Time wurde auf 0,5 Sekunden eingestellt.



Abbildung 27: AFT Amplifier der Firma Canberra Modell:2025

Das analoge Signal wird zum Multi Channel Analyser (MCA 8000A) der Firma Amptek weitergeleitet (siehe Abbildung 28), wo die Signale digitalisiert werden und je nach ihrer Signalgröße in unterschiedliche Kanäle sortiert werden. Durch die serielle Schnittstelle kann man die Daten des MCA auslesen.



Abbildung 28: MCA 8000A der Firma Amptek

## 4.2 Koordinatentisch

Dieses Kapitel beschreibt, wie die Probe durch den computergesteuerten Koordinatentisch bewegt werden kann.

### 4.2.1 Manueller Koordinatentisch

Koordinatentische erlauben es, Objekte auf einer zweidimensionalen Ebene zu verschieben. Dabei wird ein Schlitten durch eine Gewindespindel entlang einer Führungsbahn bewegt.

Für die Vermessung der für die Arbeit relevanten Probe wurde der Koordinatentisch der Firma Dema mit einer Verstellbreite von  $X = 110\text{mm}$  und  $Y = 165\text{mm}$  verwendet. In Tabelle 10 sind die Parameter des Koordinatenschlittens dargestellt. Durch Modifikationen an den Gewindestangen könnte die Verstellbreite um 10 mm an beiden Achsen vergrößert werden. Somit können Oberflächen mit einer Größe bis zu  $120\text{mm} \times 180\text{mm}$  in einem Messdurchlauf gescannt werden.

Durch Schrauben an der Seite der Schlitten wurde das Spiel zwischen Schlitten und Führungsbahn so eingestellt, dass sich der Schlitten nur mehr in die gewünschte Achse bewegt. Zudem wurden die Kurbeln, für die Manuelle Bedienung abgenommen und so der Totwinkel, der bei wechselnder Richtung entsteht, auf annähernd  $0^\circ$  gebracht. Der Vorschub pro Umdrehung ist 2mm.



Abbildung 29 : Koordinatentisch der Firma Dema (Technik, 2012)

Parameter	Werte
Parallelität des Arbeitstisches	100:0,0254mm
Verstellbreite X- Achse	110mm
Verstellbreite Y- Achse	165mm
Abweichung:	0,02mm
Nettogewicht:	16 kg
Maße LxBxH	40 x 26 x12 cm

Tabelle 10: Parameter des Koordinatentisches der Firma Dema (Kreuztisch, 2007)

#### 4.2.2 Antrieb durch Schrittmotoren

Für eine vollautomatische Messung wurden Schrittmotoren des Typs PD2-O41 der Firma Nanotech gewählt (siehe Abbildung 30). Diese Schrittmotoren sind bereits mit einem Controller bestückt und können mit einem Bus-System angesprochen werden. Da die Proben das Gewicht von 50g nicht überschreitet, ist ein Schrittmotor mit einer Eingangsleistung von 30Watt ausreichend. Für die Steuerung sind neben den RS485 Bus System auch sechs Digitaleingänge und ein Analogeingang versehen. Für die Statusanzeige sind drei Open-Kollektor Ausgänge vorhanden. Weitere Informationen befinden sich im Kapitel 4.2.5. In Tabelle 11 sind die Parameter des Schrittmotors dargestellt.



Abbildung 30: Nanotech Schrittmotor / Typ PD2-O41

Parameter	Werte
Betriebsspannung	DC12 bis 24 V max.
Phasenstrom	2,34A
Schnittstelle	RS485 oder CANopen
Betriebsart	Takt- Richtung, Position, Drehzahl, Flagposition, Analog, Joystick
Taktfrequenz:	bis zu 1 MHz bei 1/64
Eingänge	6 Digitaleingänge, 1 Analogeingang
Ausgänge	3 Open-Kollektor
Programmiersprache	C++,C', Java, LabView

Tabelle 11: Parameter des Schrittmotors Typs PD2-O41 (Nanotech, 2010)

### 4.2.3 Stromversorgung für den Schrittmotor

Um eine ausreichende Stromversorgung und eine konstante Spannungsversorgung zu gewährleisten, wurde das Schaltnetzteil NTS-24V-5A von Nanotech (siehe Abbildung 31 ) gewählt. Dieses Netzteil liefert bis zu 5 A bei einer Ausgangsspannung von 24 V(vgl. Tabelle 12). Somit können bis zu drei Schrittmotoren des Typs PD2-O4118L1804 versorgt werden. Um kleinere Spannungsschwankungen zu dämpfen wird ein Elektrolytkondensator mit 47 $\mu$ F parallel zu den Ausgangsklemmen geschaltet.



Abbildung 31: Schaltnetzteil mit des Typs NTS-24V-5A der Firma Nanotech (Nanotech, 2011)

Parameter	Werte
Output Voltage	24V
Output Current Range	0-5A
Output Rated Power	120W
Input Voltage	230 V
Input Current	40A
Efficiency	80%

Tabelle 12: Parameter des Schaltnetzteils NTS-24V-5A(Nanotech, 2011)



#### 4.2.4 I/O Schnittstellen für den Schrittmotor

Für die Spannungsversorgung der Endschalter, der Bedienschalter und der Status-LEDs wurde eine Platine gefertigt. Die Aufgabe dieser Platine ist, die Eingangsspannung für die Schrittmotoren zu glätten und durch einen Spannungsregler eine Versorgung der Taster und der LEDs von 5 Volt zu garantieren. Um maximale Sicherheit zu gewährleisten, wurden Öffner als Taster und Endschalter gewählt. Kommt es zu einem Kabelbruch, so wird der Messvorgang unterbrochen.

Um zu ermöglichen, dass der PC auf des RS485 Bus System zugreifen kann, wurde ein USB to RS485 Konverter verwendet. Der letzte Schrittmotor im RS485-Bus-System wurde mit einem Abschlusswiderstand von 120kOhm ausgestattet, um ihn vor Induktivität zu schützen.

#### 4.2.5 Die Platine

Jeder Schrittmotor ist mit einer Platine ausgestattet um die I/O Schnittstellen mit Signalen zu versorgen. Die Spannungsversorgung der Platinen erfolgt über das Schaltnetzteil NTS-24V-5A. Wie im Kapitel 4.2.3 bereits erwähnt, benötigt das Netzteil einen Kondensator, um Spannungsschwankungen zu kompensieren. Auf jeder Platine befindet sich ein 45µF Kondensator. Da beide Platinen parallel am Netzteil hängen, wirken die Kondensatoren wie ein großer 90µF Kondensator.

Für die Platine werden keine 24 Volt benötigt, sondern lediglich 5 und 4,1 Volt. Der Spannungsregler 7805 stellt 5 Volt zur Verfügung. Diese ist notwendig um die Dioden mit einen 2,2 kΩ Vorwiderstand zu versorgen. Der Open Kollektor Ausgang vom Schrittmotor (JP1/Pin 9, 10,11) steuert die Dioden, an wie am Schaltplan in Abbildung 32 dargestellt ist.

Durch den Spannungsteiler fällt am 1 kΩ Widerstand die Spannung von 5V auf 4,1 Volt ab. Diese Spannung wird über den Taster (Öffner) zurück an den Schrittmotor (JP1/Pin 2, 3, 4, 5, 6,7) geleitet. Wird ein Taster betätigt, so fällt die Spannung am Schrittmotoreingang auf 0 Volt ab.

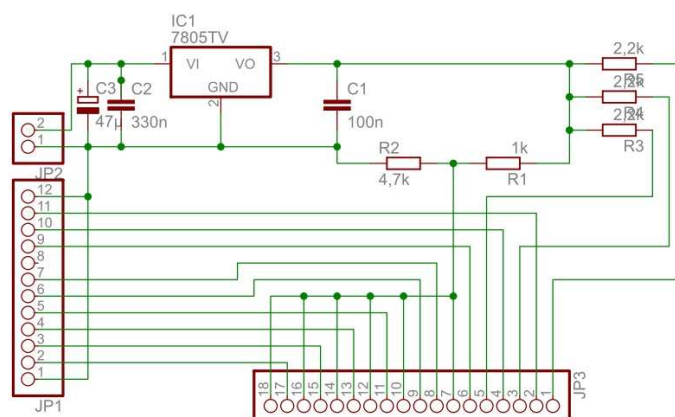


Abbildung 32: Schaltplan für die Stromversorgung der Ein und Ausgänge

Das Layout des Boards wie in Abbildung 33 präsentiert erfolgte im Program Eagle. Die Platine wurde gefräst und anschließend mit den elektrischen Bauteilen (siehe Tabelle 13 ) bestückt.

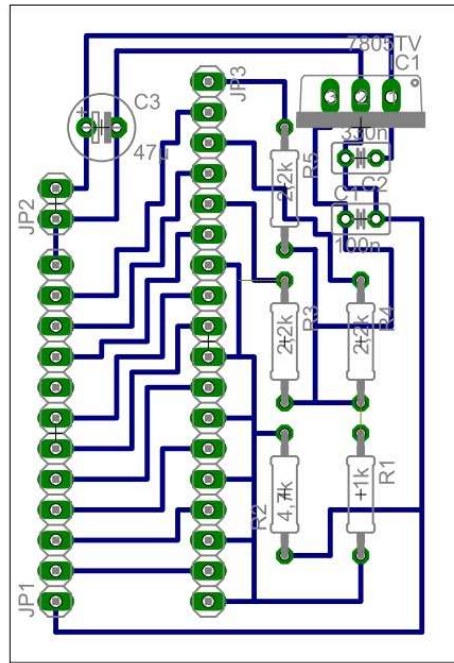


Abbildung 33: Layout der realisierten Platine für den Schaltplan in Abbildung 32

Anzahl	Bauteile	Eigenschaften
2	Platine	40mm x 50mm
2	Elektrolytkondensator	47µF
2	Tantalkondensator	330nF
2	Tantalkondensator	100nF
6	Widerstand	2,2kΩ
2	Widerstand	1kΩ
2	Widerstand	4,7kΩ

Tabelle 13: Verwendete elektrische Bauteile



## 4.2.6 Steuerung des Schrittmotors

Auf der Rückseite des Schrittmotors, ist bereits ein Controller angebracht. Seine Aufgabe ist es, Befehle von Schnittstellen entgegenzunehmen und die Magneten im Motor richtig anzusteuern. Der Schrittmotor von Nanotech besitzt einen  $\mu$ -Prozessor auf dem man bereits einfache Programme speichern kann. Durch ein Signal von den Tastern, können vorgespeicherte Bewegungen ausgeführt werden.

Über das RS485-Bus-System können Computerprogramme, die in Java, C++, C# oder LabView programmiert wurden komplexe Steuerbefehle für den Schrittmotor erteilen. Auch die Zustände der Ein und Ausgänge der Platine können in das Programm eingelesen werden.

## 4.2.7 Kopplung von Koordinatentisch und Schrittmotor

Dafür wurde ein Aluminium L-Profil mit den Schlitten und ein Aluminium L-Profil mit den Schrittmotoren verbunden. Durch eine Schraubenverbindung kann die Position der beiden Profile so verändert werden, dass die Gewindespindel und der Rotor des Motors auf einer Achse liegen. In Abbildung 34 und Abbildung 35 ist die Kopplung zwischen dem Schrittmotor und dem Koordinatentisch dargestellt. Durch die exakte Montage ist keine Kupplung nötig. In die Gewindespindel wurde mittig von der Drehachse ein Loch gebohrt und der Rotor des Schrittmotors darin eingebettet. Die beiden Achsen wurden mit Wurmschrauben verbunden. Um zu vermeiden, dass die Welle berührt wird, wurde eine 1mm starke Blech-Abdeckung über die Welle angebracht. In dieser Abdeckung wurden die Taster sowie die Status-LEDs montiert. Eine Bewegung des Motors um 202 Schritte bewirkt am Kreuztisch eine Positionsveränderung um 1mm.

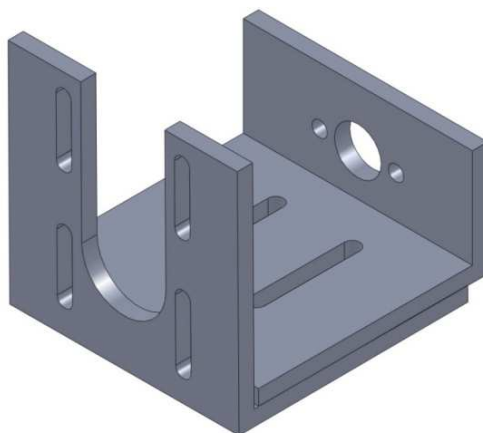
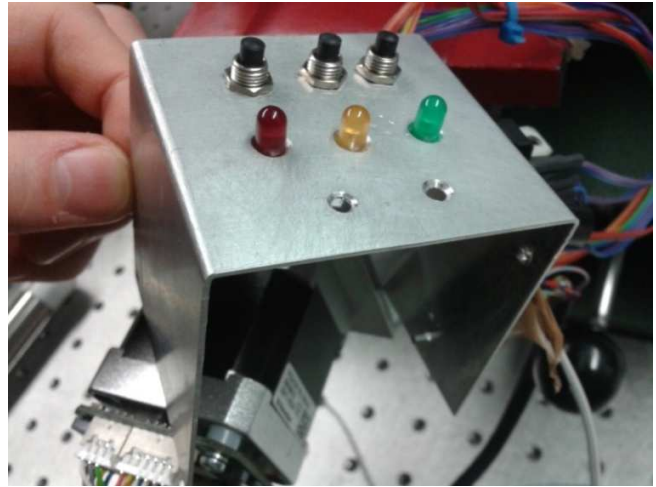
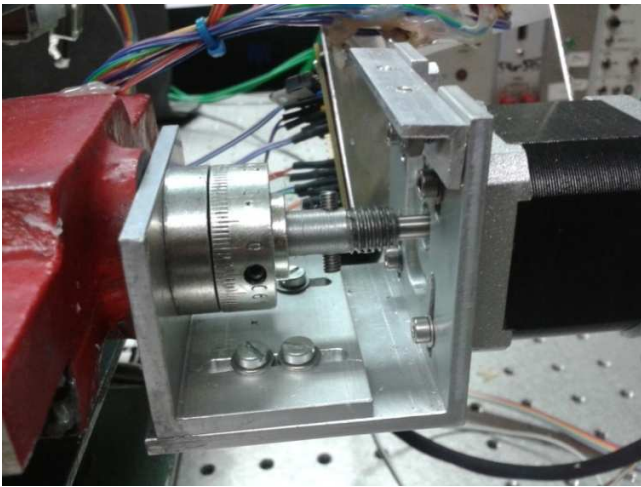


Abbildung 34: in Solid Works konstruierte Kopplung aus zwei L-Profilen



**Abbildung 35: Realisierte Kopplungshalterung mit Kopplung beider Wellen und Gehäuse mit Taster und LEDs**

Die Endschalter, die benötigt werden um die Position zu erkennen, werden auf der gegenüberliegenden Seiten der Schrittmotoren angebracht. Die Drähte sind mit Kabelbindern und einem Thermoplast befestigt, damit sie nicht durch Bewegungen des Koordinatentisches abgetrennt werden.

## 4.3 Kombination von Koordinatentisch und XRF Anlage

Die XRF Anlage muss mechanisch sowie softwaretechnisch mit dem Koordinatentisch abgestimmt werden.

### 4.3.1 Der mechanische Aufbau

Um die Entfernung zwischen RFA Messanlage und Probe einzustellen wird der automatisierte Koordinatentisch auf einen pneumatischen Labor-Boy befestigt. Der PMMA Block mit SD Detektor und Röntgenröhre befindet sich über den Koordinatentisch, wie Abbildung 36 zeigt. Auf dem Koordinatentisch befindet sich eine PMMA Platte, die anregende Röntgenstrahlung vom Eisen des Koordinatentisches abschirmt. Über eine serielle Verbindung ist der Koordinatentisch und der Multi Channel Analyser (MCA) mit dem Computer verbunden.

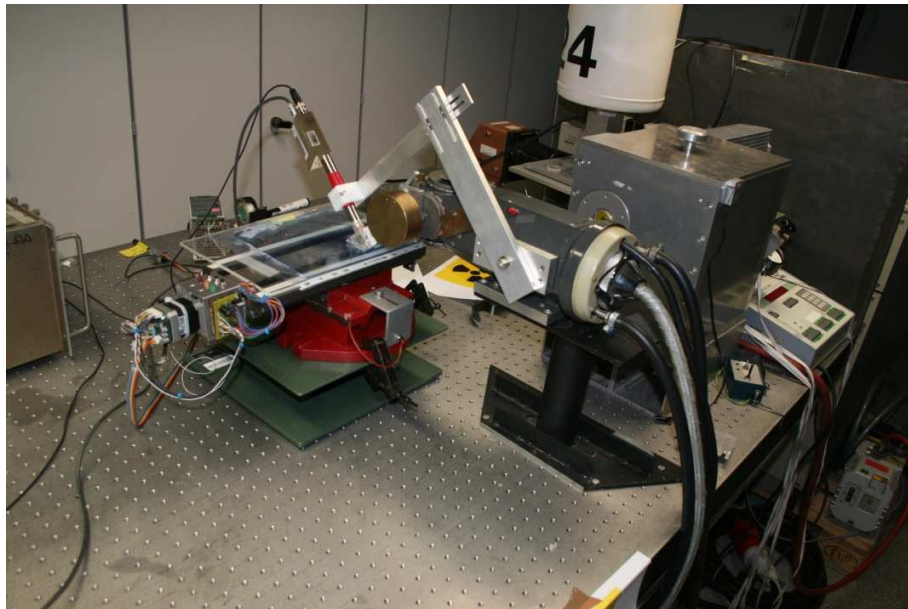


Abbildung 36: Messaufbau mit Koordinatenschlitten und XRF Messanlage

### 4.3.2 Software

Der Multi Channel Analyser (MCA) des Detektors und der Schrittmotor lassen sich softwaretechnisch mit unterschiedlichen Programmiersprachen ansteuern. Für die Einbindung beider Komponenten wurde die weitverbreitete Programmiersprache C# verwendet.

Die Aufgaben des Programmes sind:

- Einzelmessungen mit XRF
- Graphische Darstellungen des gemessenen Spektrums
- Energiekalibrierung des Spektrums
- Numerische Darstellung des gemessenen Spektrums
- Speichern einzelner Spektren und Wiederaufrufen
- Erzeugen eines Summenspektrums
- Präzise Bewegung der zweidimensionalen Probe zu jeder gewünschten Stelle
- Präzise Angabe der Position des Schlitten
- Erzeugen eines Batch für die automatische Auswertung
- Rasterfunktion (Messen und Bewegen der Probe) mit Auswahl:
  - Wahl der zeitliche Messauflösung
  - Wahl der örtliche Messauflösung
  - Wahl der Größe der Oberfläche
  - Speichern in unterschiedlichen Formaten

Abbildung 37 gibt Übersicht über den Aufbau des Steuerprogrammes. An der linken Seite des Programmes können Bewegungen für den Schrittmotor übermittelt werden (blau markiert), an der rechten Seite Einstellungen für den MCA (grün markiert). Im roten Bereich erfolgt die Einstellung der Ports und der Darstellung des Spektrums. Das Spektrum ist im gelben Bereich zu erkennen.

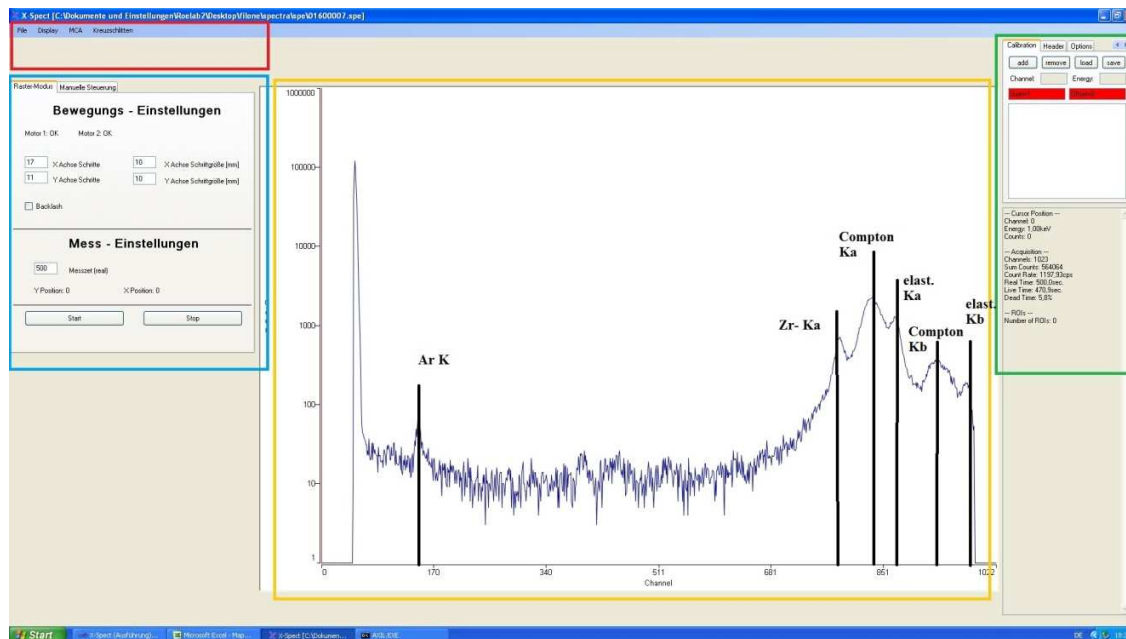


Abbildung 37: Benutzeroberfläche zum Steuern des MCA und des Koordinatenschlittens

Wird das Programm gestartet, so muss zuerst eine Verbindung zum MCA und zum Koordinatentisch aufgebaut werden. In der oberen Leiste, im roten Bereich, wird die Port-Nummer der beiden Komponenten gewählt. Der MCA befindet sich auf Port 1 und der Koordinatentisch, auch Kreuztisch genannt, auf Port 4. Nun kann mit der Messung begonnen werden.



Abbildung 38: Einstellungen um die Verbindung zum MCA und zum Koordinatentisch aufzubauen

Wählt man im blauen Bereich den Karteikartenreiter „Manuelle Steuerung“ so kann der Koordinatentisch unabhängig vom XRF Mess-System bewegt werden. Durch den Button „Start Position“ werden beide Endschalter (X- und Y-Achse) angefahren. Diese Position ist als 0/0-Position definiert. Kennt man die zu messende Koordinaten im zweidimensionalen Raum so kann man auf diese Position anfahren. Mit den Positionstasten kann der Koordinatentisch in jede Richtung um 2mm verschoben werden.

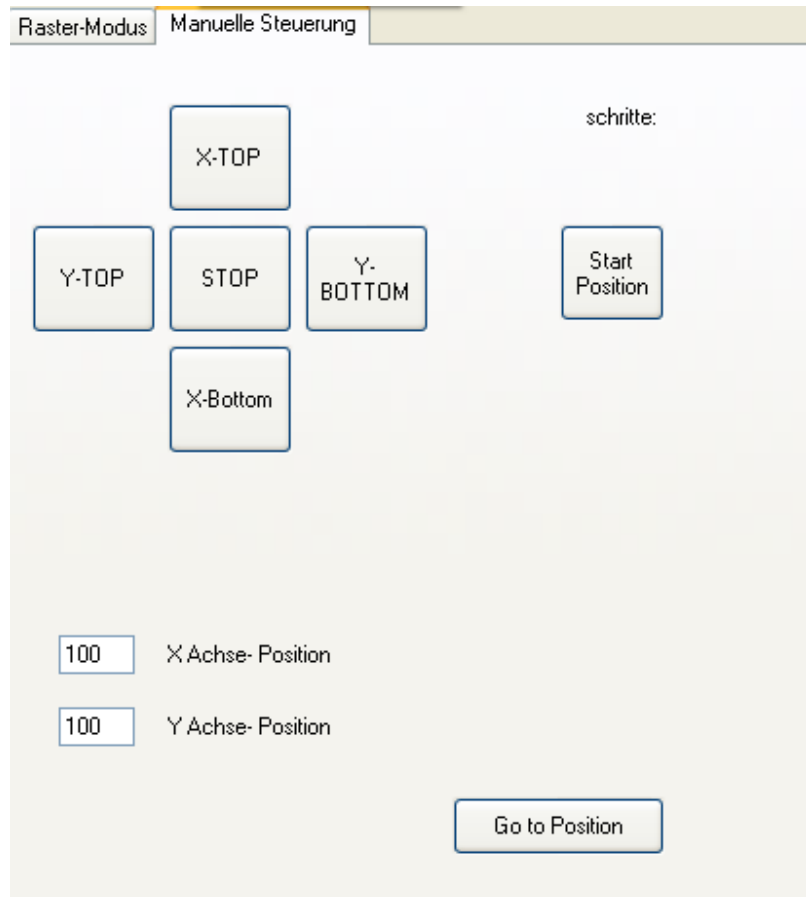


Abbildung 39: Grafische Oberfläche zum Bewegen des Koordinatentisches (ohne Messung)

Durch die Wahl des Karteikartenreiters „Raster Modus“ kann automatisch gemessen werden. Hierbei werden die Anzahl der Schritte und die jeweiligen Schrittgrößen eingegeben. Darunter wird die Messzeit eingetragen wie lange an jedem Punkt gemessen werden soll. Unter „Option“ kann der Speichermodus gewählt werden. Dabei wird jeder Messpunkt in Format: XXXXYYYYY.spe gespeichert. Die Platzhalter X und Y stehen für die Position in mm an der gemessen wird. Die SPE-Dateien werden auf den Desktop gespeichert und können mit weiteren Programmen bearbeitet werden.

The image shows a software window titled 'Raster-Modus' with a sub-tab 'Manuelle Steuerung'. It contains two main sections: 'Bewegungs - Einstellungen' and 'Mess - Einstellungen'.

**Bewegungs - Einstellungen**

- Motor 1: OK      Motor 2: OK
- X Achse Schritte: 17      X Achse Schrittgröße [mm]: 10
- Y Achse Schritte: 11      Y Achse Schrittgröße [mm]: 10
- ☐ Backlash

**Mess - Einstellungen**

- Messzeit (real): 500
- Y Position: 0      X Position: 0
- Start      Stop

Abbildung 40: Grafische Oberfläche zum scannen einer Zweidimensionalen Probe

Zur Analyse wird das Programm QXAS von der International Atomic Energy Agency (IAEA) verwendet. Dieses Dekonvolution- Programm unterstützt bei der Zuordnung von Energie Peaks. Aufgrund des Mess-Aufbau entsteht immer ein Argon  $K_{\alpha}$ -Peak welches sich in der Umgebungsluft befindet und der Compton-Peak mit elastisch gestreuten Peak von Molybdän.

Aus den Peaks wird die Konzentration eines Elements im Vergleich zu anderen Elementen durch mathematische Algorithmen ermittelt. Die Summen- und Escape- Peaks werden in diesem Programm beachtet. Durch das im Steuerprogramm erzeugte Batch File werden alle Messpunkte ausgewertet und als ASR File gespeichert.

## 4.4 Das Programm Datamove

Durch das Programm Datamove, das in MatLab entwickelt wurde, kann für alle Elemente die im Programm QXAS gefunden wurden, eine zweidimensionale Karte erstellt werden. Dabei werden alle ASR Files aufgerufen, die vom Dekonfluenzprogramm erstellt wurden und analysiert. Durch die gespeicherte Position der Messpunkte im ASR File, ebenfalls im Format XXXXYYYYY.asr, kann jeder Intensitätswert eines Elements einer zweidimensionalen Position zugewiesen werden. Das Programm Datamove berücksichtigt auch die möglichen Fehler des Dekonfluenzprogramms und subtrahiert diese von den Endresultaten. Schließlich wird eine zweidimensionale Karte generiert, wo die Intensität eines Elements als Farbe sichtbar gemacht wird. Abbildung 41 enthält einen Ausschnitt des Programms Datamove. Durch Auswahl der Elemente können die jeweiligen Karten angezeigt werden

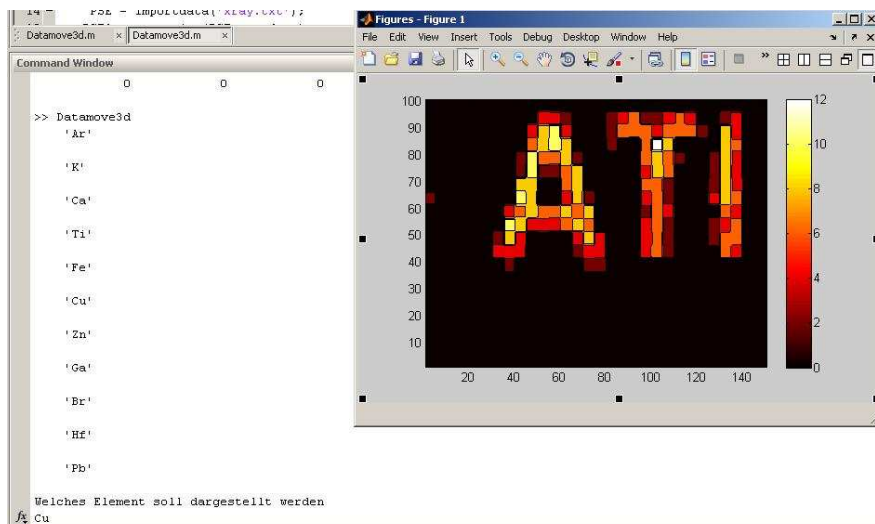


Abbildung 41: Mathlab Program Datamove3d



## 5 Einstellen der Mess- Anordnung

### 5.1 Die Geometrie

Das Elektrophorese-Gel ist eine flexible Struktur die bei mechanischer Belastung leicht einreißt. Um die Stabilität zu erhöhen wurde das Material in einer Kunststoffolie verpackt und mit der PMMA-Platte des Koordinatentischs verklebt. Da die Oberfläche der Kunststoffolie wellig ist und sich der Koordinatentisch bewegt, ist die Probe nicht mehr in der vom Plexiglas vorgegebenen Referenzfläche. Der Schnittpunkt der Strahl- und Detektorachse liegt bei einem fix montierten Detektor in der Referenzebene. Somit überlagert sich das Gel nicht mehr mit der vom Detektor messbaren Referenzfläche. Der Abstand  $d$  zwischen Referenzfläche und der Kunststoffolie verursacht eine Verschiebung des Schnittpunktes von der Detektor- und der Strahlen- Achse um  $\sqrt{2} \cdot d$ . Dies ist in Abbildung 42 veranschaulicht.

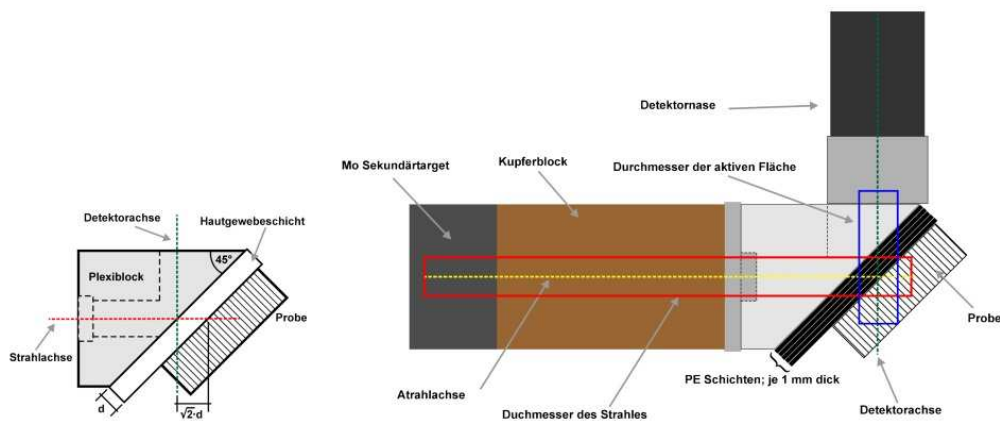


Abbildung 42: Geometrieproblem, da sich die Probe nicht in der Referenzfläche befindet (Pemmer, 2007)

Im XRF Scanner kann dieser Messfehler durch Verstellen des Labor-Boys, auf dem sich der Koordinatentisch befindet, reduziert werden. Das Elektrophorese- Gel wird dabei näher an die Referenzfläche gebracht. Das Gel darf nicht anliegen, um eine freie Bewegung des Koordinatentisches zu ermöglichen.

Eine weitere Möglichkeit diesen Messfehler zu reduzieren, ist das Verstellen der Detektorposition wie im kommenden Kapitel erklärt wird.

## 5.2 Justierung des Detektors

Die Messanordnung ist so justiert, dass die Strahlenquelle und der Detektor  $90^\circ$  versetzt sind, wie die Abbildung 17 zeigt. Die Probenfläche erstreckt sich auf horizontaler Ebene. Der PMMA - Block besitzt zwei Strahlengänge (ein Gang für die Anregung und einer für die Detektion der Photonen). Um das größtmögliche Messsignal zu erlangen, muss der Detektor parallel am PMMA Keil anliegen. Durch Langlöcher an der Detektorhalterung kann der Detektor justiert werden, wie in Abbildung 43 demonstriert. Für die Einstellung wurde eine Kupferplatte auf die Probenfläche gelegt. Das Kupfer wurde mit einer Röhrenspannung von 50keV und einem Röhrenstrom von 40mA mit einem Mo- Sekundärtarget angeregt. Das Signal am Detektor wurde 10 Sekunden gemessen. Danach wurde der Detektor entlang der X- Richtung um 1,8mm verschoben und erneut gemessen indem der Vorgang wiederholte wurde.

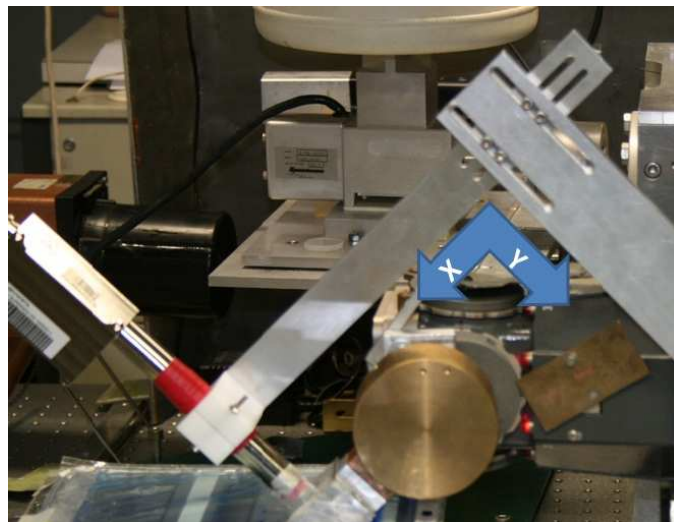


Abbildung 43: Position des Detektors zum PMMA Block einstellbar über den Detektorarm

Position	Cu K $\alpha$	Cu K $\alpha$
mm	Gross Area [cts* $\Delta E$ ]	Net Area [cts* $\Delta E$ ]
0	115566	114270
1,6	158009	156821
3,2	205912	203932
4,8	258636	256656
6,4	267964	265552
8	264278	262334
9,6	183137	181517

Tabelle 14: Gemessene Kupfer Intensität an der jeweiligen Position

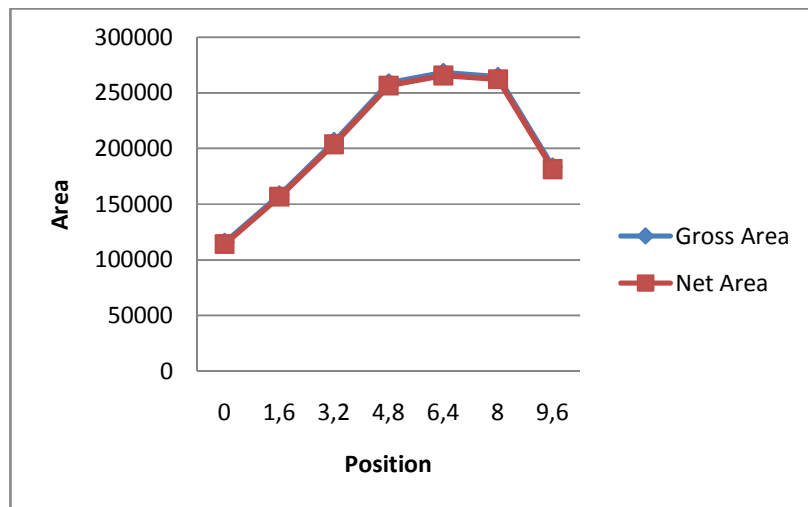


Abbildung 44: Darstellung der gemessenen Intensität von Kupfer bei Positionsänderung des Detektors entlang der X Achse

Bei der Messung wurde das Signal vom  $K_{\alpha}$ -Übergang von Kupfer ausgewertet. In die Tabelle 14 ist die gemessene gesamt Fluoreszenz-Strahlung (Gross Area) und die Strahlung ohne Hintergrundrauschen (Net Area) eingetragen. Befindet sich der Arm 9 mm von der definierten 0 Position entfernt, so wird das meiste Kupfer detektiert. Abbildung 44 veranschaulicht die Einstellung der Messpunkte und der detektierten Kupfer- Fluoreszenzstrahlung.

### 5.3 Bestimmen der Auflösung

Um die zweidimensionale Auflösung festzustellen wurde ein 1µm starker Kupferdraht in x und y Richtung auf eine PMMA Platte, wie in Abbildung 45 dargestellt, geklebt. Die PMMA Platte ist auf dem Kreutzisch befestigt und das XRF Messsystem so nahe wie möglich darüber gebracht. Der Koordinatentisch bewegt sich nun in 1mm Schritte in X Richtung über den in Y Richtung liegenden Draht. Zwischen jedem Schritt wurde die Photofluoreszenz für 100 Sekunden gemessen. Dieser Vorgang wurde auch in Y Richtung mit einem in X Richtung liegenden Draht wiederholt. Die Messdaten werden in Diagrammen eingetragen und die Halbwertsbreite gemessen

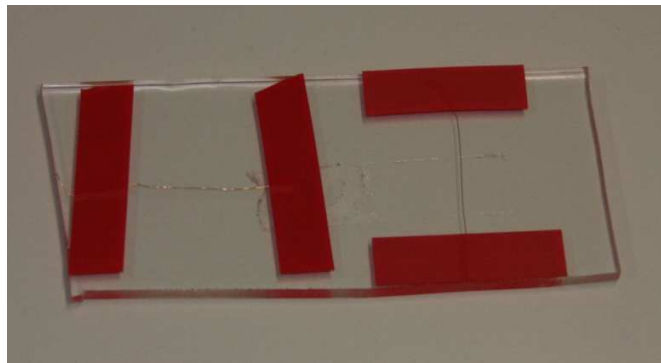


Abbildung 45: PMMA Platte mit Kupfer Draht in X und Y Richtung

Position	X- Achse	Y- Achse
mm	cts	cts
1	0,060972	10,75
2	0	0
3	0	7,16
4	0,24389	7,16
5	0,12194	0,71
6	0	0
7	0,18292	10,037
8	0,67069	0
9	0	0
10	0,24389	0
11	0,060972	4,3
12	11,95	10,75
13	76,703	7,88
14	182,18	0
15	327,36	0
16	413,39	2,15
17	456,37	7,89

18	423,88	22,23
19	346,99	111,84
20	229,44	224,48
21	116,15	281,76
22	34,754	346,28
23	4,329	342,71
24	0	285,34
25	0	182,81
26	0	124,75
27	0,18292	52,34
28	0	16,49
29	-	8,6
30	-	9,32
31	-	4,3
32	-	0,71692
33	-	0
34	-	5,74
35	-	0
36	-	10,04

**Tabelle 15: Intensitätsänderung bei Verfahren der RFA Messanlage normal zum Kupferdraht**

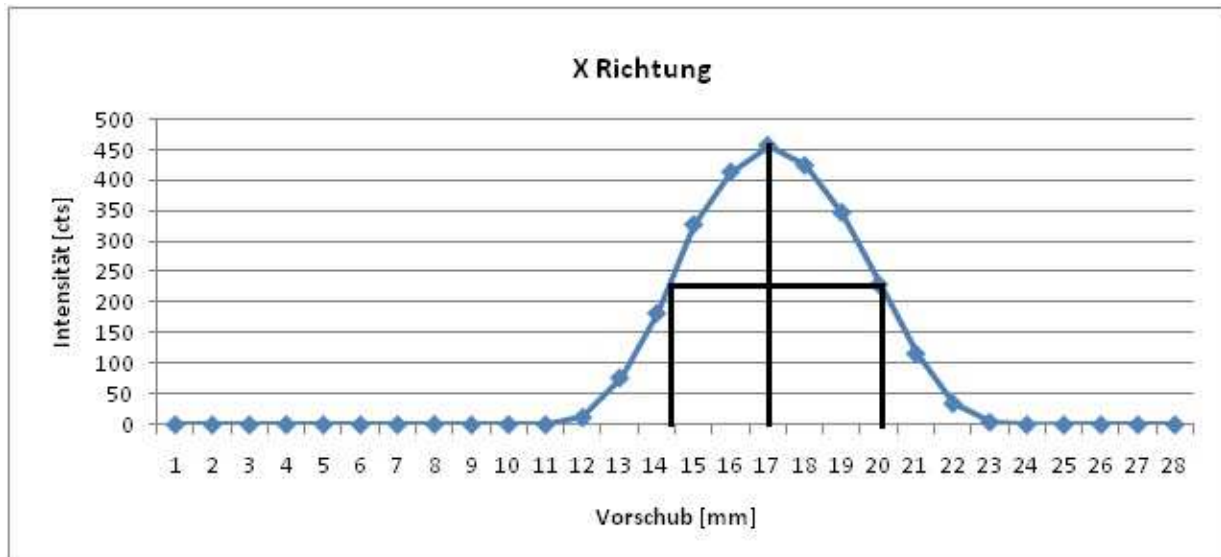


Abbildung 46: Auflösung in X Richtung des XRF Scanners ohne Kollimator

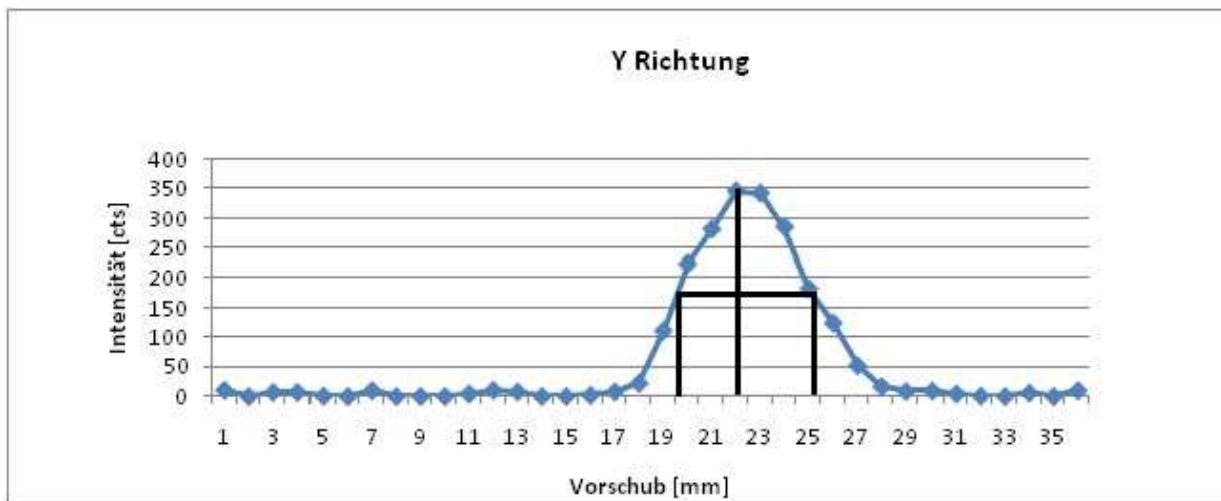


Abbildung 47: Auflösung in Y Richtung des XRF Scanners ohne Kollimator

Diese Methode macht die Größe der Fläche sichtbar, in der das Kupfer detektiert wird. Die Halbwertsbreite gibt die Differenz zwischen den beiden Punkten auf der horizontalen Achse an, wo das Maximum um die Hälfte gefallen ist. Für die X-Achse und Y-Achse beträgt die Halbwertsbreite, wie sich aus Abbildung 47 und Abbildung 46 ablesen lässt, ca. 6 mm.

Für eine bessere zweidimensionale Auflösung muss ein Kollimator in den Strahlengang gegeben werden. Dabei wird der Brennfleck auf der Probe kleiner und die Intensität der Strahlung verringert.

Ein Kollimator (siehe Abbildung 48) mit 2mm Durchmesser wurde 4 cm nach dem Sekundärartarget eingesetzt. Der Kollimator besteht aus Molybdän, da die Strahlung nach dem Sekundärartarget nicht mehr genug Energie hat um ihn anzuregen. Wie vorher bereits beschrieben wurde erneut die Auflösung bestimmt. Die Halbwertsbreite beträgt nun nur mehr 3 mm.



Abbildung 48: Kollimator mit 2mm Durchmesser

Position	X- Achse	Y- Achse
mm	cts	Cts
1	0	0,30486
2	0,36583	0
3	0	0,24389
4	0	0
5	0,12194	0
6	0,12194	0
7	0,60972	0,18292
8	0,12194	0,060972
9	0	0
10	0,24389	0,060972
11	0	0
12	0,24389	0
13	0	0,12194
14	0	0
15	8,2312	0,12194
16	76,825	0,18292
17	189,93	0,060972
18	251,51	0
19	217,12	0
20	123,22	0
21	40,241	0,67069
22	9,5116	1,2194
23	4,4509	2,7437
24	2,8608	40,546
25	0,67069	158,59
26	0,36583	283,03
27	0,24389	236,75
28	0,12194	86,824
29	0,060942	15,914
30	0	0,91458

Tabelle 16: Intensitätsänderung bei Verfahren der RFA Messanlage normal zum Kupferdraht mit Kollimator

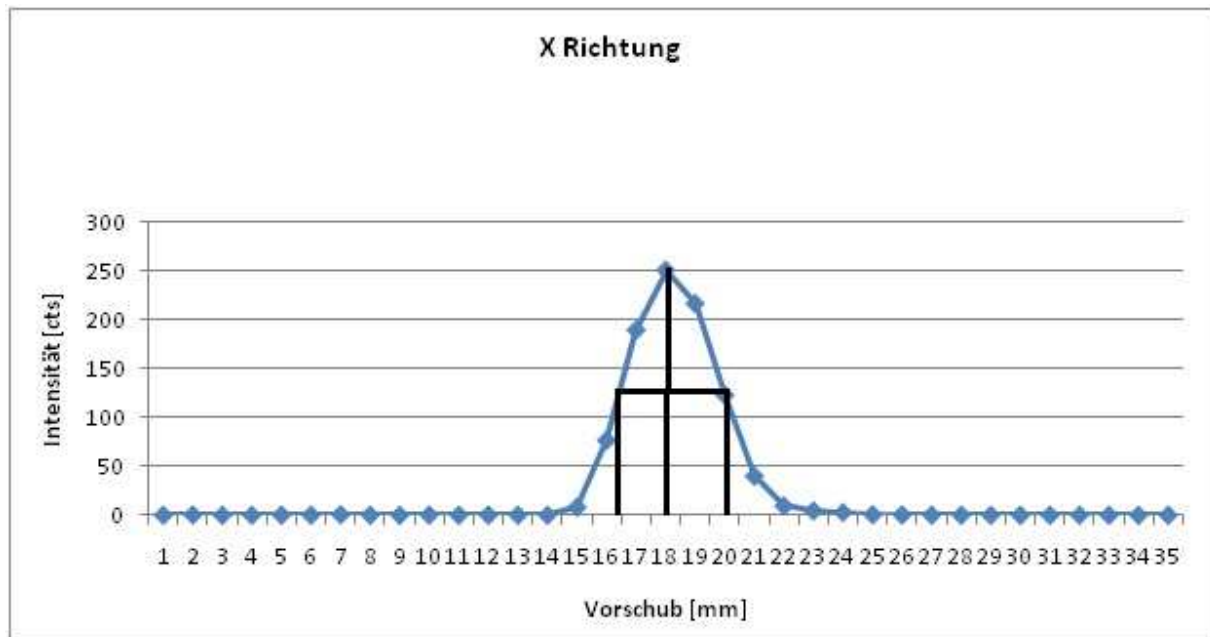


Abbildung 49: Auflösung in X-Richtung des XRF Scanners mit Kollimator

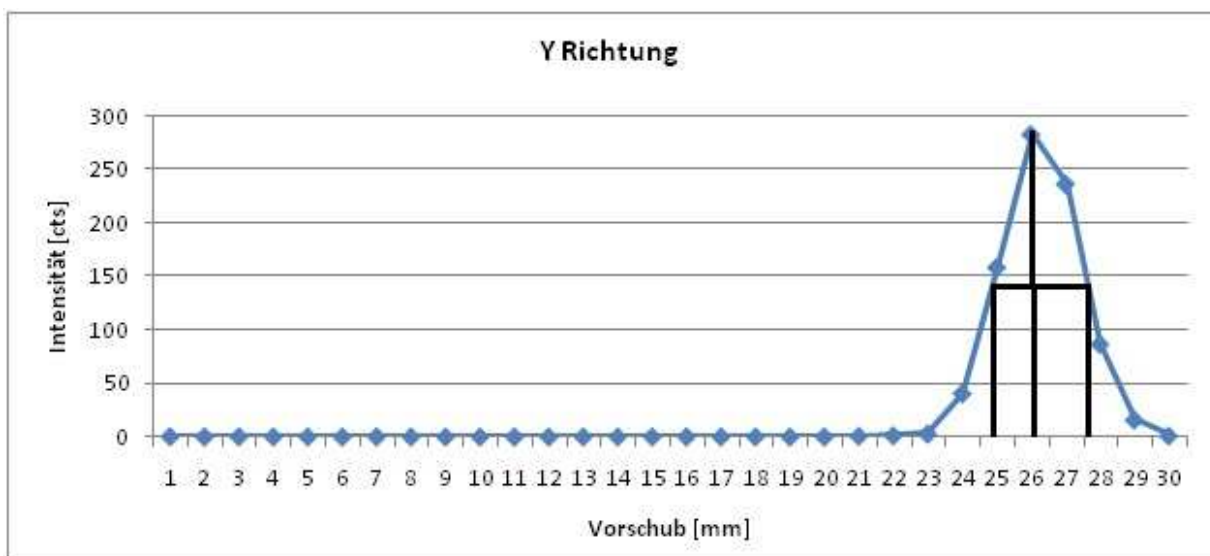


Abbildung 50: Auflösung in Y-Richtung des XRF Scanners mit Kollimator

Aus Abbildung 49 und Abbildung 50 ist zu erkennen, dass die Halbwertsbreite nun nur mehr 3 mm beträgt.



## 5.4 Bestimmen des Leerwerts

Für die Untersuchung von Spurenelementen in Elektrophorese-Gel muss sichergestellt werden, dass die Fluoreszenzstrahlung ausschließlich von der Probe stammt und nicht von Teilen des Spektrometers. Elemente, mit Ausnahme des Materials des Sekundärtargets, die gemessen werden, obwohl keine Probe vorhanden ist, werden als Leerwerte bezeichnet. Materialien aus leichten Elementen wie beispielsweise Plexiglas oder Polypropylen können nicht vom Detektor erfasst werden, da die Intensität der Fluoreszenzstrahlung zu gering ist. Aus strahlenschutztechnischen Gründen muss das Spektrometer aus Elementen mit einer hohen Ordnungszahl gebaut werden, um das bedienende Personal zu schützen. Im XRF-Scanner wurden folgende Materialien verwendet, die vom Detektor erfasst werden können:

Elemente	Abkürzung	Position im XRF-Scanner
Argon	Ar	Umgebungsluft
Aluminium	Al	Strahlengang / Detektor
Eisen	Fe	Koordinatentisch
Kupfer	Cu	Strahlengang
Molybdän	Mo	Sekundärtarget

Tabelle 17: Mögliche Leerwerte im XRF Scanner

Wird Acrylamid- oder Agarose-Gel in der Luft gemessen, soll idealerweise kein Element, ausgenommen Argon, detektiert werden. Um dies zu überprüfen, wird ein Blank- Gel vermessen mit folgenden Messeinstellungen:

Messeinstellungen	Parameter
X Achse Schritte:	12
Y Achse Schritte	19
X Achse Schrittgröße	6mm
Y Achse Schrittgröße	6mm
Messzeit (realtime):	350 sek.
Messzeit insgesamt	22h
Speicherung :	SPE File, TXT File, SUM File, ASR File, Batch File
Röhreneinstellungen	50KV/40mA
Sekundärtarget	Molybdän
Detektor	25mm <sup>2</sup> Amptek
Kollimator	nein

Tabelle 18: Messeinstellungen für die Leerwert-Messung

In der Blank-Messung wurden folgende Leerwerte gefunden:

Element	Intensität	Toleranz
	Cts	Cts
Ar	1030	19
K	295	16
Fe	221	16
Ca	174	16
Ni	1673	22
Cu	277	18
Au	228	16
Pb	486	19

Tabelle 19: gemessene Leerwerte im Elektrophorese- Gel

Eisen (Fe), Kalzium (Ca), Nickel (Ni) und Gold (Au) wurden zusätzlich zu den vermuteten Elementen gefunden. In Abbildung 51 und Abbildung 52 ist das Blank-Spektrum logarithmisch und linear dargestellt.

Die Leerwertbestimmung muss für jede Anlage gemacht werden, um eine falsche Interpretation der Messergebnisse auszuschließen. Werden Leerwerte entdeckt, müssen sie von einer Messung abgezogen werden. Dabei ist zu beachten, dass die Messbedingungen für die Leerwertmessung gleich wie die der zu untersuchenden Probe sein müssen.

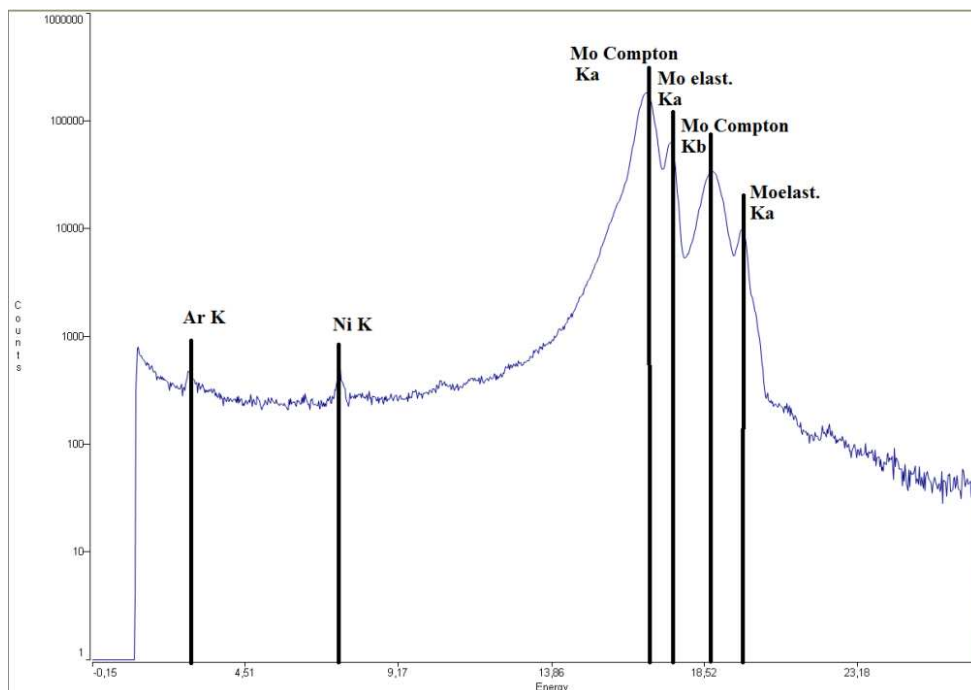


Abbildung 51: Logarithmisches Summenspektrum von der Blank-Messung

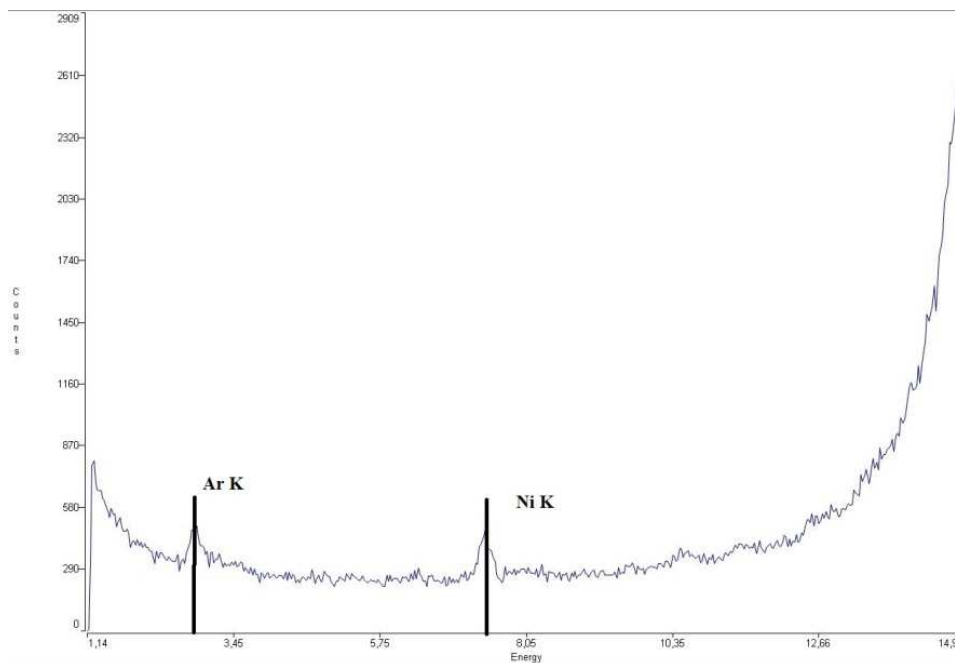


Abbildung 52: Lineares Summenspektrum von der Blank-Messung

## 6 Messungen mit dem XRF Scanner:

### 6.1 Messung einer definierten Probe

Um die Funktion des XRF Scanners zu überprüfen, wird eine Oberfläche mit unterschiedlicher Elementverteilung, die auch optisch sichtbar ist, vermessen. Dafür eignet sich am besten ein Papier, das mit der blauen Farbe eines Laserdruckers (Cyan) bedruckt wird. Im Toner der Farbe Cyan befinden sich neben den Farbpigmenten und Harzen auch Kupferpartikel. Diese werden durch den Laserdrucker auf das Papier aufgeschmolzen. So kann ein blaues Kupfermuster auf ein Papier aufgedruckt werden, wie in Abbildung 53 dargestellt.



Abbildung 53: Blaue Buchstaben auf weißem Papier, mittels Laser-Drucker bedruckt

Das bedruckte Blatt wurde auf die PMMA Platte geklebt. Danach wurde das Papier durch den Labor-Boy in die Höhe der Referenzfläche gebracht und gescannt. Es wurde ein Detektor mit 5mm<sup>2</sup> Fenster von der Firma Ketek verwendet.

Messeinstellungen	Parameter
X Achse Schritte:	12
Y Achse Schritte	19
X Achse Schrittgröße	6mm
Y Achse Schrittgröße	6mm
Messzeit (realtime):	350 sek.
Messzeit insgesamt	22h
Speicherung :	SPE File, TXT File, SUM File, ASR File, Batch File
Röhreneinstellungen	50KV/40mA
Sekundärtarget	Molybdän
Detektor	5mm <sup>2</sup> Ketek
Kollimator	nein

Tabelle 20: Messeinstellung für die Messung einer definierten Probe

Abbildung 54 zeigt das Kupfermuster auf dem Papier. An den Rändern des Schriftzuges ATI erkennt man, dass weniger Kupfer detektiert wurde. Hier befand sich die Kante des Schriftzuges in der Mitte des Detektors. Durch die Skala an der Seite erkennt man die Position der einzelnen Messpunkte. Dieser Versuch bestätigt die Fähigkeit des XRF-Scanners, Elemente zweidimensional darzustellen. Spiegelt und dreht man den Scan, so kann der Schriftzug ATI gelesen werden.

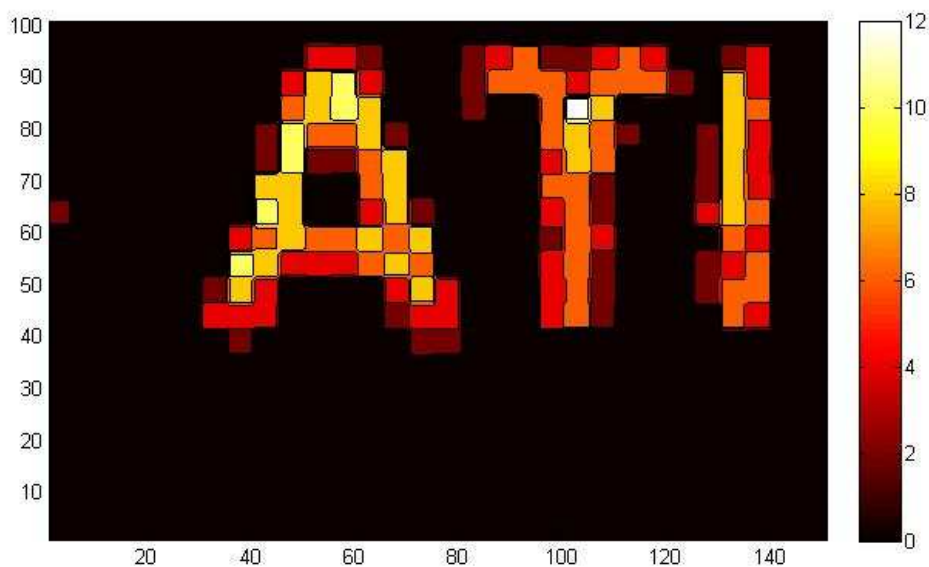


Abbildung 54 : Kupfer Scan des bedruckten Papiers

## 6.2 Messung von Elektrophorese-Gel.

Dieses Kapitel beschreibt, wie Elektrophorese-Gel gemessen wird, mit der dazugehörige Präparation und den Messeinstellungen.

### 6.2.1 Scannen des Elektrophorese-Gels mit der Bezeichnung „Gel 1“

Der XRF-Scanner kann verwendet werden um Elektrophorese-Gele, die mit Metalloproteinen bestückt wurden, nach Elementen zu untersuchen. Wie im Kapitel 3 bereits erwähnt, besteht das zu untersuchende Gel aus Elementen mit einer niedrigen Ordnungszahl. Dies hat den Vorteil, dass die Matrix in der die Proteine eingebettet sind, nicht detektiert wird, da Elemente mit niedriger Ordnungszahl eine geringe Fluoreszenzausbeute besitzen. Die leichten Elemente der Matrix führen jedoch zu einer Streuung im Medium. Das Hintergrundrauschen wird erhöht und die gemessene Position der schweren Elemente kann verfälscht werden. Um Verunreinigungen auf der Kunststoffolie auszuschließen, wurde diese mit Isopropanol gereinigt wie in Abbildung 55 dargestellt ist.



Abbildung 55: Reinigung der Probe

Für die Messung des Elektrophorese- Gels mit der Bezeichnung Gel 1 wurden eine Größe von 120 x 180 mm gescannt. Der Abstand zwischen den einzelnen Messpunkten beträgt 10 mm und jeder Messpunkt wurde 500sec (Realtime) spektroskopiert. Daraus ergeben sich 26 Stunden Messzeit, in der 216 SPE Files generiert wurden. In der nachfolgenden Tabelle befinden sich die Messeinstellungen:

Messeinstellungen	Parameter
X Achse Schritte:	17
Y Achse Schritte	11
X Achse Schrittgröße	10mm
Y Achse Schrittgröße	10mm
Messzeit (realtime):	500 sek.
Messzeit insgesamt	26h
Speicherung :	SPE File, TXT File, SUM File, ASR File, Batch File
Röhreneinstellungen	50KV/40mA
Sekundärtarget	Molybdän
Detektor	5mm <sup>2</sup> Ketek
Kollimator	nein

**Tabelle 21: Messeinstellungen für Gel 1**

Für die Analyse wurden alle Spektren aufaddiert, da die Einzelmessungen für sich keine repräsentative Aussagekraft haben. Tabelle 22 bietet einen Überblick über die Gesamtauswertung der Datenbasis. Die erste Spalte zeigt die Messergebnisse der Elemente im Gel in der Einheit counts/85990sec. Um diese mit dem Blank zu vergleichen werden die Messergebnisse auf counts/440sec umgerechnet, da die Blankmessung für 440 Sekunden erfolgte. In der dritten und vierten Spalte befindet sich die Blankmessung mit dem berücksichtigten Korrektur. Für eine konservative Betrachtung wird der Fehler zur Blankmessung in Spalte „Korrigiert Blank.ASR“ dazu addiert. Nun wird der Wert der Spalte „SUM.ASR“ von der Spalte „Korrigiert Blank“ abgezogen und in Spalte „Gel Messwerte“ dargestellt. Um das Verhältnis der Elemente zueinander zu ermitteln, wird der Gel Messwert durch den Korrektur Faktor, bezogen auf Mo-Strahlung, dividiert.

Messwerte in der letzten Spalte zeigen die Verteilung der Elemente im Gel. So wurde annähernd gleich viel Eisen und Hafnium detektiert. Diese Elemente kommen mit hoher Wahrscheinlichkeit aus der PMMA Platte und den Koordinatenschlitten, die sich unter der Probe befinden. Das detektierte Argon stammt aus der Umgebungsluft

	SUM ASR [counts/8590sec]	SUM ASR- Korrigiert [counts/440sec]	BLANK ASR [counts/440sec]	+/- Fehler [counts]	Korrigiert Blank ASR [counts/440sec]	Gel Messwerte [counts/440sec]	I kor = I/Kf	Normierte Messwerte
Argon	67319.000	344.463	260.000	11	271.000	73.463	48.748	45.584
Kalium	5469.000	27.984	-10.000	7	0	27.984	12.920	12.081
Kalzium	4789.000	24.505	17.000	7	24.000	0.505	0.139	0.130
Titan	371.000	1.898	-32.000	6	0	1.898	0.523	0.489
Eisen	5903.000	30.205	13.000	6	19.000	11.205	1.167	1.091
Kupfer	7671.000	39.252	52.000	8	60.000	n/a	n/a	n/a
Zink	4031.000	20.626	15.000	7	22.000	n/a	n/a	n/a
Gallium	2198.000	11.247	-20.000	7	0	11.247	0.520	0.486
Brom	-278.000	-1.422	3.000	8	11.000	n/a	n/a	n/a
Haftium	21254.000	108.754	52.000	6	58.000	50.754	1.069	1.000
Blei	10265.000	52.525	40.000	6	46.000	6.525	0.057	0.053
Livetime	85990	Korrekturfaktor: 195.4318182	Livetime 440					

Tabelle 22: Auswertung für das GEL mit der Bezeichnung „Gel 1“



Aufgrund der kurzen Messzeit ist in den einzelnen Messpunkten das Hintergrundrauschen so sehr ausgeprägt so, dass Elemente nicht eindeutig detektiert werden konnten. Es ist keine zweidimensionale Darstellung möglich.

Um die Nachweisgrenze zu erhöhen kann die:

- Intensität der Röntgenstrahlung erhöht werden,
- ein energiedispersiver Detektor mit einem größerem Fenster verwendet werden,
- die Messzeit erhöht werden
- oder die Geometrie verbessert werden.

Da bereits mit maximaler Strahlleistung der Röntgenröhre gemessen wurde, konnte dieser Faktor nicht mehr verstärkt werden. Um dennoch die Nachweisgrenze zu erhöhen, wurde ein Detektor mit einem 25mm<sup>2</sup> großen Fenster gewählt.

## 6.2.2 Scannen des Elektrophorese Gel mit der Bezeichnung „Muera“

Mit den verbesserten Messbedingungen wurde das Gel mit der Bezeichnung Muera via XRF-Scanner vermessen. Das Gel ist 120x 70 mm groß. Für diesen Scan wird kein Kollimator verwendet und die Pixelgröße beträgt 6x6 mm. Der 5 fach größere Detektor der Firma Amptek (25mm<sup>2</sup>) erlaubt es die Messzeit auf 350 Sekunden zu reduzieren. Die Dauer der Messung erstreckt sich über 24 Stunden.

Messeinstellungen	Parameter
X Achse Schritte:	12
Y Achse Schritte	19
X Achse Schrittgröße	6mm
Y Achse Schrittgröße	6mm
Messzeit (realtime):	350 sek.
Messzeit insgesamt	22h
Speicherung :	SPE File, TXT File, SUM File, ASR File, Batch File
Röhreneinstellungen	50KV/40mA
Sekundärtarget	Molybdän
Detektor	25mm <sup>2</sup> Amptek
Kollimator	nein

Tabelle 23: Messeinstellungen für das Elektrophorese Gel mit der Bezeichnung Muera

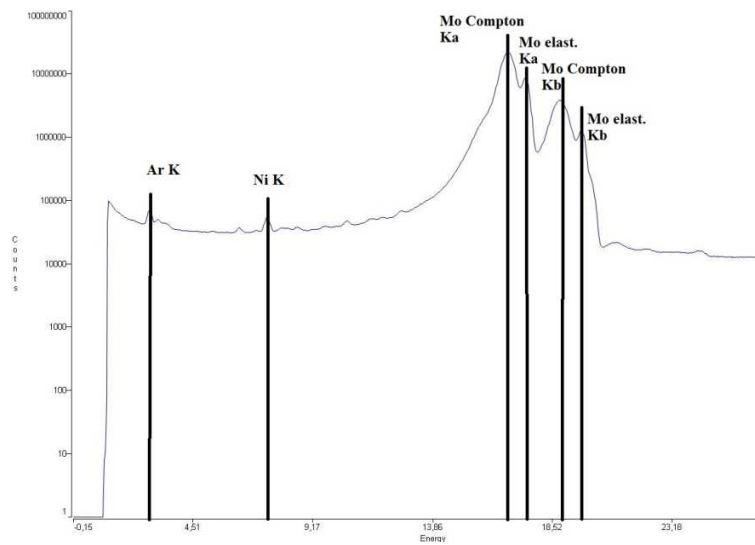


Abbildung 56: Logarithmisch dargestelltes Spektrum vom Elektrophorese Gel

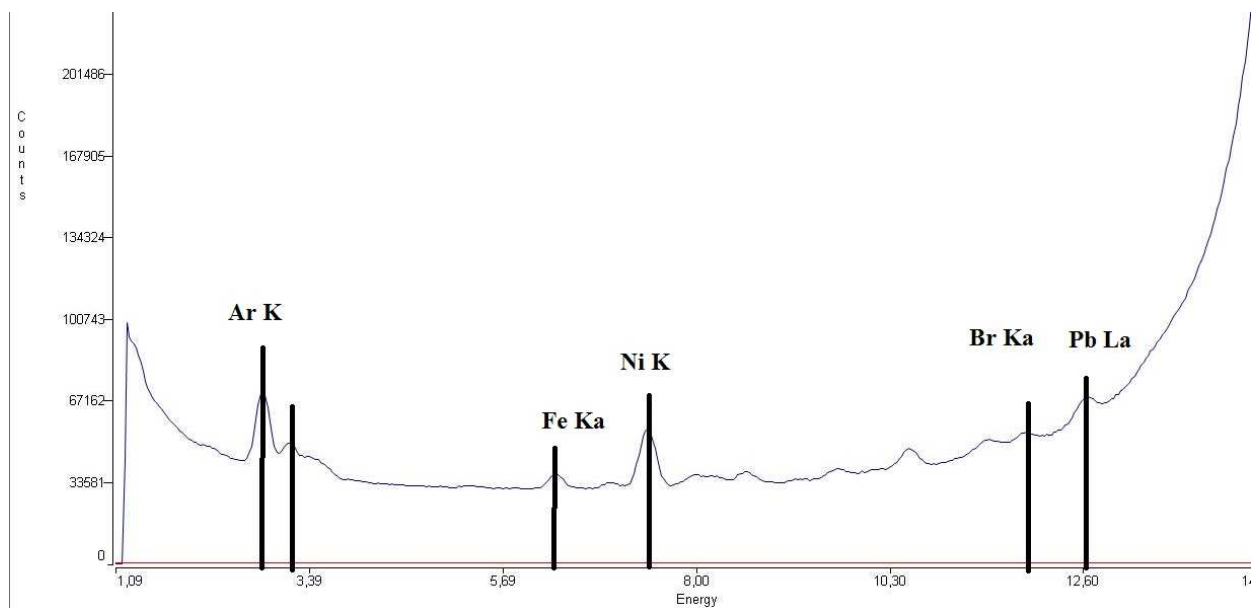


Abbildung 57: Lineares Spektrum vom Elektrophorese Gel

Abbildung 56 und Abbildung 57 zeigen das Summenspektrum des Elektrophorese-Gels mit linearer und logarithmischer X Achse. Bei 11,923 keV findet man im Summenspektrum den Peak der Charakteristischen Strahlung von Brom.

Durch das Programm Datamove wird eine farbcodierte zweidimensionale Karte des Broms im Elektrophorese- Gel generiert. In Abbildung 58 ist die BromKarte dargestellt.

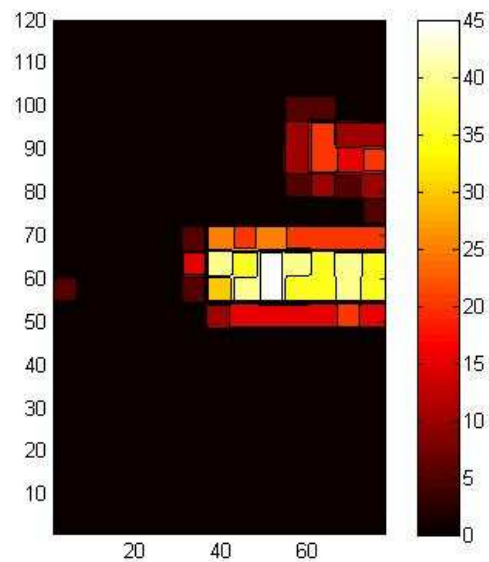


Abbildung 58: Brom Scan von Elektrophorese Gel

Dieser Scan zeigt, dass sich eine höhere Konzentration von Brom auf einem definierten Bereich befindet.

## 7 Zusammenfassung und Ausblick

Im Rahmen dieser Diplomarbeit wurde ein XRF-Scanner für biologische Proben entwickelt. Das Mess- System erlaubt Messungen in Umgebungsluft bei Normalbedingungen. Durch diese Eigenschaft können biologische Proben mit Flüssigkeits- und Luft- Einschlüssen gemessen werden.

Die Probenmatrix aus leichten Elementen erhöht die Streuung. Durch ein Sekundärtarget, das die intensive Primärstrahlung polarisiert, wird dieser Streuanteil drastisch reduziert. Durch adäquate Wahl des Sekundärtargets können die Messbedingung für die untersuchte Probe optimiert werden. Leichte Elemente (Na bis Sc) werden optimal durch eine Röntgenröhre mit einer Chrom Anode und einem Titan Sekundärtarget angeregt, dabei ist Helium Spülung vorausgesetzt. Die Automatisierung der XRF Anlage erlaubt lange Messzeiten, wodurch die Nachweisgrenze erhöht wird.

Durch die Kombination von Koordinatentisch und Schrittmotor ist es möglich eine Genauigkeit der Positionseinstellung mit einer maximalen Abweichung von 0,02 mm zu erlangen. Die Limitierung der zweidimensionalen Auflösung ist durch den Durchmesser des anregenden Strahls der Probe gegeben. Der anregende Strahl hat eine Halbwertsbreite von 6 mm, daher sollte die Schrittgröße zwischen den einzelnen Scan- Punkten nicht kleiner als 6 mm sein. Um dennoch die zweidimensionale Auflösung zu verbessern kann ein Kollimator verwendet werden. Wird ein Kollimator mit einen Durchmesser von 2mm kurz vor den Strahlenausgang eingesetzt, so verringert sich die Halbwertsbreite auf 3 Millimeter. Scans mit einer Pixelgröße von 3 mm sind möglich.

Die Software ermöglicht an interessanten Punkten, wo bereits mit großer Schrittweite gescannt wurde, erneut mit einem Kollimator mit kleineren Schrittweiten zu scannen. Dies reduziert die Messzeit und führt zu einer höheren zweidimensionalen Auflösung. Es können auch einzelne Punkte nachgemessen werden.

Während der Messung wird ständig das Spektrum des gemessenen Punktes aktualisiert. Bei Besonderheiten kann daher jederzeit in das Programm eingegriffen werden. Durch Taster am Kreuzschlitten kann dieser auch bedient werden, wenn der Computer nicht eingeschaltet ist. In der Steuer-Software können komplexe Bewegungen des Koordinatentisches realisiert werden.

Für die Auswertung der Spektren wird das zuverlässige Programm QXAS der IAEA eingesetzt. Dieses Programm wertet alle Spektren im batch-mode aus und speichert die Intensität der vorhandenen Elemente. Für die graphische zweidimensionale Auswertung wird das dafür entwickelte Programm Datamove genutzt. Neben der graphischen Darstellung sind auch quantitative Analysen durch das Programm Datamove möglich wenn ausreichend Informationen über die zu untersuchende Matrix gegeben sind. Eine sensitivity Korrektur auf Mo K bezogen wurde berücksichtigt, die es erlaubt Elemente relativ miteinander zu verknüpfen.

Im Kapitel „Messung einer definierten Probe“ wird die korrekte Funktion des XRF Scanners demonstriert. Der blaue Schriftzug in dem sich Kupfer befindet kann durch eine zweidimensionale Kupfer Elementkarte dargestellt werden.

Die zu untersuchenden Proben können zweidimensional sein und die Elemente müssen über der Nachweisgrenze liegen. Liegt die Probe nicht parallel zur Referenzebene, wie sie in Kapitel 5.1 beschrieben ist, so sinkt die Nachweisgrenze des XRF Scanners. Eine automatisierte Justierung der Z Achse würde dieses Problem lösen. Die Probe wird bei jeder Messung gegen den PMMA Block des Scanners gedrückt.

Um die optischen Auffälligkeiten der Probenoberfläche zu betrachten, kann in den PMMA Block eine Kamera ausgestattet werden, die bei jeder Messung ein Photo aufnimmt. Zu Ende des Scans könnten die Bilder zusammengefügt und die Elementintensität darüber projiziert werden.

Elektrophorese-Gel, in dem sich Metalloproteine befinden, eignet sich zum Scannen. In Kapitel 6.2 wurde eine Elektrophorese-Gel-Probe gescannt, dabei wurde in einem Bereich eine erhöhte Anzahl von Brom gefunden. Durch den XRF Scan des Elektrophorese-Gels wird die molekulare Analyse von Proteinen um die atomare Analyse ergänzt. Der Einfluss von Metalloproteinen auf physiologische und pathologische Prozesse im menschlichen Körper kann dadurch besser verstanden werden.

## 8 Abbildungsverzeichnis

Abbildung 1: Elektromagnetisches Spektrum (Rothhardt, 2011) .....	6
Abbildung 2: energieaufgelöstes Röntgenspektrum (Dyson, 1990) .....	7
Abbildung 3: wellenlängenaufgelöstes Röntgenspektrum(Haken and Wolf, 2004) .....	7
Abbildung 4: Bremsstrahlung(Demtröder, 2009) .....	9
Abbildung 5 : Darstellung des Photoeffekts und dessen Folgereaktionen (Krieger and Petzold, 1988).....	11
Abbildung 6: Schematische Darstellung des Fluoreszenz- Effekts (Demtröder, 2009).....	12
Abbildung 7: Darstellung des Auger-Effekts (Demtröder, 2009).....	13
Abbildung 8: Darstellung der elastischen Streuung (Marosi, 2002) .....	15
Abbildung 9: Darstellung der inelastischen Streuung(Marosi, 2002).....	16
Abbildung 10: Intensitätsabnahme, wenn Strahlung durch einen Körper tritt.....	17
Abbildung 11: Absorptionskanten von Blei (Jenkins et al., 1981) .....	18
Abbildung 12: Schematische Darstellung einer Röntgenröhre (Demtröder, 2009) .....	20
Abbildung 13: Schematische Darstellung der Versuchsanordnung mit Sekundärtarget (Cernohlawek, 2005) .....	21
Abbildung 14: Bändermodell für Leiter, Halbleiter und Isolator (Kück, 2002) .....	22
Abbildung 15: p-n Übergang mit Polung in Durchlassrichtung (1) und in Sperrrichtung (2) (Kück, 2002).....	23
Abbildung 16: Schematische Darstellung des SI(Li) Detektors (Jenkins et al., 1991).....	24
Abbildung 17: Schematische Darstellung des Prinzips der Seitwärtsdepletion (Pahlke, 2004) ....	25
Abbildung 18: Seitliche schematische Darstellung eines SD Detektors mit Driftfeld (Pemmer, 2007).....	25
Abbildung 19: dreidimensionale Darstellung eines SD Detektors mit integrierten FET(Eggert, 2004).....	26
Abbildung 20: Signalverarbeitungskette einer EDRFA Mess- Systems (Eggert, 2004) .....	27
Abbildung 21: Prinzip der Gelelektrophorese (Hübl and Milosavljevic, 2011) .....	29
Abbildung 22: Grafische Darstellung der zulässigen Spannung für Long Fine Focus Röntgenröhren von Panalytical (Tube, 2007) .....	31
Abbildung 23: Strahlengang mit Sekundärtarget .....	32
Abbildung 24 : PMMA Block um die Geometrie zwischen Detektor und Probe festzulegen (Pemmer, 2007) .....	32
Abbildung 25: verstellbarer Detektorarm.....	33
Abbildung 26: Röntgenröhre, Sekundärtarget und Detektor .....	33
Abbildung 27: AFT Amplifier der Firma Canberra Modell:2025 .....	34
Abbildung 28: MCA 8000A der Firma Amptek .....	34
Abbildung 29 : Koordinatentisch der Firma Dema (Technik, 2012) .....	35

Abbildung 30: Nanotech Schrittmotor / Typ PD2-O41 .....	36
Abbildung 31: Schaltnetzteil mit des Typs NTS-24V-5A der Firma Nanotech (Nanotech, 2011).....	37
Abbildung 32: Schaltplan für die Stromversorgung der Ein und Ausgänge.....	38
Abbildung 33: Layout der realisierten Platine für den Schaltplan in Abbildung 32.....	39
Abbildung 34: in Solid Works konstruierte Kopplung aus zwei L-Profilen.....	40
Abbildung 35: Realisierte Kopplungshalterung mit Kopplung beider Wellen und Gehäuse mit Taster und LEDs.....	41
Abbildung 36: Messaufbau mit Koordinatenschlitten und XRF Messanlage.....	42
Abbildung 37: Benutzeroberfläche zum Steuern des MCA und des Koordinatenschlittens .....	44
Abbildung 38: Einstellungen um die Verbindung zum MCA und zum Koordinatentisch aufzubauen .....	44
Abbildung 39: Grafische Oberfläche zum Bewegen des Koordinatentisches (ohne Messung)....	45
Abbildung 40: Grafische Oberfläche zum scannen einer Zweidimensionalen Probe.....	46
Abbildung 41: Mathlab Program Datamove3d .....	47
Abbildung 42: Geometrieproblem, da sich die Probe nicht in der Referenzfläche befindet (Pemmer, 2007).....	48
Abbildung 43: Position des Detektors zum PMMA Block einstellbar über den Detektorarm .....	49
Abbildung 44: Darstellung der gemessenen Intensität von Kupfer bei Positionsänderung des Detektors entlang der X Achse.....	50
Abbildung 45: PMMA Platte mit Kupfer Draht in X und Y Richtung .....	51
Abbildung 47: Auflösung in Y Richtung des XRF Scanners ohne Kollimator .....	53
Abbildung 46: Auflösung in X Richtung des XRF Scanners ohne Kollimator .....	53
Abbildung 48: Kollimator mit 2mm Durchmesser .....	54
Abbildung 49: Auflösung in X-Richtung des XRF Scanners mit Kollimator .....	55
Abbildung 50: Auflösung in Y-Richtung des XRF Scanners mit Kollimator .....	55
Abbildung 51: Logarithmisches Summenspektrum von der Blank-Messung .....	57
Abbildung 52: Lineares Summenspektrum von der Blank-Messung .....	58
Abbildung 53: Blaue Buchstaben auf weißen Papier, mittels Laser-Drucker bedruckt.....	59
Abbildung 54 : Kupfer Scan des bedruckten Papiers.....	60
Abbildung 55: Reinigung der Probe.....	61
Abbildung 56: Logarithmisch dargestelltes Spektrum vom Elektrophorese Gel.....	65
Abbildung 57: Lineares Spektrum vom Elektrophorese Gel .....	65
Abbildung 58: Brom Scan von Elektrophorese Gel.....	66

## 9 Tabellenverzeichnis

Tabelle 1: Bezeichnung der Variablen für die Formel 1, Formel 2, Formel 3.....	8
Tabelle 2: Bezeichnung der Variablen für Formel 8, Formel 7, Formel 6.....	10
Tabelle 3: Bezeichnung der Variablen für Formel 9.....	12
Tabelle 4: Bezeichnung der Variablen.....	12
Tabelle 5: Bezeichnung der Variablen für Formel 10.....	14
Tabelle 6: Bezeichnung der Variablen für Formel 11 und Formel 12 .....	15
Tabelle 7: Bezeichnung der Variablen für Formel 13.....	16
Tabelle 8: Bezeichnung der Variablen für Formel 14 und Formel 15 .....	17
Tabelle 9: maximal zulässige Strom und Spannungswerte der 2,2 kW Ag Röntgenröhre (Tube, 2007).....	31
Tabelle 10: Parameter des Koordinatentisches der Firma Dema (Kreuztisch, 2007) .....	35
Tabelle 11: Parameter des Schrittmotors Typs PD2-O41 (Nanotech, 2010) .....	36
Tabelle 12: Parameter des Schaltnetzteils NTS-24V-5A(Nanotech, 2011).....	37
Tabelle 13: Verwendete elektrische Bauteile.....	39
Tabelle 14: Gemessene Kupfer Intensität an der jeweiligen Position.....	49
Tabelle 15: Intensitätsänderung bei Verfahren der RFA Messanlage normal zum Kupferdraht...	52
Tabelle 16: Intensitätsänderung bei Verfahren der RFA Messanlage normal zum Kupferdraht mit Kollimator .....	54
Tabelle 17: Mögliche Leerwerte im XRF Scanner .....	56
Tabelle 18: Messeinstellungen für die Leerwert-Messung .....	56
Tabelle 19: gemessene Leerwerte im Elektrophorese- Gel.....	57
Tabelle 20: Messeinstellung für die Messung einer definierten Probe .....	59
Tabelle 21: Messeinstellungen für Gel 1.....	62
Tabelle 22: Auswertung für das GEL mit der Bezeichnung „Gel 1“ .....	63
Tabelle 23: Messeinstellungen für das Elektrophorese Gel mit der Bezeichnung Muera .....	64



## 10 Literaturverzeichnis

- CERNOHLAWEK, N. 2005. *Parameterstudien zur Energiedispersiven RFA von Blei im menschlichen Knochen*. Diplomarbeit, Technische Universität Wien.
- DEMTRÖDER, W. 2009. *Experimentalphysik 3*, Berlin Heidelberg New York, Springer.
- DYSON, N. A. 1990. *X-Rays in Atomic and Nuclear Physics*, Cambridge, Cambridge University Press.
- EGGERT, T. 2004. Der Ketek Silizium Drift-detektor: Grundlagen Anwendungen Funktionsprinzipien. Available: [www.ketek.net](http://www.ketek.net).
- HAKEN, H. & WOLF, H. 2004. *Atom- und Quantenphysik : Einführung in die experimentellen und theoretischen Grundlagen*, Springer Verlag.
- HÜBL, W. & MILOSAVLJEVIC, D. 2011. *Elektrophorese - eine Einführung* [Online]. med4you. Available: [http://www.med4you.at/laborbefunde/techniken/elektrophorese/lbef\\_elektrophorese.htm](http://www.med4you.at/laborbefunde/techniken/elektrophorese/lbef_elektrophorese.htm) 2011].
- JENKINS, R., GAULD, R. W. & GEDCKE, D. 1981. *Quantitative X-Ray Spectrometry*, New York, Marcel Dekker INC.
- JENKINS, R., GAULD, R. W. & GEDCKE, D. 1991. *Quantitative X-Ray Spectrometry*, New York, Macel Dekker.
- KRAMERS, H. A. 1923. On the theory of x-ray absorption and of the continuous x-ray spectrum. *Philosophical Magazine*.
- KREUZTISCH, X. 2007. 18507 Koordinatentisch In: DEMA (ed.). Deutschland.
- KRIEGER, H. & PETZOLD, W. 1988. *Strahlenphysik, Dosimetrie und Strahlenschutz*. Stuttgart, B.G Teubner.
- KÜCK, H. 2002. Grundlagen der Mikrotechnik: Dotieren
- KULENKAMPFF, H. 1923. The continuous x-ray spectrum. *Annalen der Physik*.
- MAROSI, N. 2002. *Aufbau eines EDXRS-Mikroanalyse-Meßplatzes mit Polykapillaro-optik*. Technischen Universität Wien.
- NANOTECH 2010. Steping Motor PD-04118L18042. Nanotec.
- NANOTECH 2011. 120 W Single Output Industrial DIN RAIL Power Supply. In: NANOTECH (ed.).
- PAHLKE, A. 2004. Dissertation, Technische Universität München.
- PEMMER, B. 2007. *Studien zur Quantifizierung: In vivo energiedispersive Röntgenfluoreszenzanalyse von Blei im menschlichen Knochen*. Diplomarbeit, Technische Universität Wien
- REHM, H. & LETZEL, T. 2010. *Der Experimentator* Heidelberg, Spektrum Akademischer Verlag.
- REINEKER, P. & SCHULZ, M. 2007. *Theoretische Physik III Quantenmechanik1*, Deutschland, Wiley-VCH Verlag GmbH&Co.
- ROTHHARDT, R. 2011. *roro-seiten* [Online]. Available: [http://www.roro-seiten.de/physik/lk12/emwellen/elektromagnetisches\\_spektrum.html](http://www.roro-seiten.de/physik/lk12/emwellen/elektromagnetisches_spektrum.html) 2011].
- TECHNIK, W. U. 2012. *Kreuztisch 140x310mm* [Online]. Available: [http://www.werkzeug-und-technik.de/15728\\_Kreuztisch\\_140x310mm/8f6ae28285f37d8b75fd5be6b7c507fb/](http://www.werkzeug-und-technik.de/15728_Kreuztisch_140x310mm/8f6ae28285f37d8b75fd5be6b7c507fb/).
- TUBE, X. G. 2007. Instruction manual. In: PANALYTICAL (ed.).
- WOBRAUSCHEK, P. Spezialverfahren der RFA. In: VORLESUNGSSCRIPTUM (ed.). Technische Universität Wien.
- WOBRAUSCHEK, P. 2012. Methods of quantitative X-Ray Fluorescence. Atominstitut der Österreichischen Universitäten.

# 11 Anhang

## 11.1 Steuerprogramm

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;
using CommandsPD4I;
using System.Threading;

namespace μXRF.XSpect
{
    public partial class formMainXSpect : Form
    {

        public enum xAxisModes
        {
            Channel,
            Energy
        };

        public enum yAxisModes
        {
            Linear,
            Logarithmic
        };

        public enum zoomModes
        {
            auto,
            manual,
            setStart,
            setEnd
        };

        private enum channelEnterByCursorModes
        {
            none,
            ROIstart,
            ROIend,
            calibration
        };

        public class CountedEventArgs : EventArgs
        {
            int rep = 0;
            string[] ROInames;
```

```

uint[] start, end;

ulong[] gross, net;

public CountedEventArgs(string[] Names, uint[] StartChannels, uint[] EndChannels, ulong[] GrossCounts, ulong[] NetCounts)
{
    ROInames = Names;
    start = StartChannels;
    end = EndChannels;
    gross = GrossCounts;
    net = NetCounts;
}

public CountedEventArgs(ListOfROI ROIs, ulong[] Spectrum)
{
    int length = ROIs.Count;
    ulong back;

    ROInames = new string[length];
    start = new uint[length];
    end = new uint[length];
    gross = new ulong[length];
    net = new ulong[length];

    for (int i = 0; i < length; i++)
    {
        ROInames[i] = ROIs[i].Name;
        start[i] = ROIs[i].Start;
        end[i] = ROIs[i].End;
        ROIs[i].GetData(Spectrum, out gross[i], out net[i], out back);
    }
}

public string[] ROInames
{
    get
    {
        return ROInames;
    }
}

public uint[] StartChannels
{
    get
    {
        return start;
    }
}

public uint[] EndChannels
{
    get
    {
        return end;
    }
}

public ulong[] GrossCounts
{
    get
    {

```

```

        return gross;
    }
}

public ulong[] NetCounts
{
    get
    {
        return net;
    }
}
}

public ComMotorCommands motor1;
public ComMotorCommands motor2;

private delegate void changeSepctrumDataDelegate(ulong[] data, float realtime, float livetime, uint percentageComplete);
private delegate void displayReportDelegate();
private delegate void acquisitionStopDelegate();
private delegate void gostepandwaitDelegate();
public delegate void countedEventHandler(object sender, CountedEventArgs e);
public delegate void mcaErrorEventHandler(object sender, EventArgs e);

private changeSepctrumDataDelegate changeSpectrumDataDel;
private displayReportDelegate displayReportDel;
private acquisitionStopDelegate acquisitionStopDel;
private acquisitionStopDelegate acquisitionDel;
private gostepandwaitDelegate gostepandwaitDel;

public event countedEventHandler Counted;
public event mcaErrorEventHandler MCAError;
public bool busy = true;
public bool busymca = false;
public bool end = false;
public int rep = 0;
public int repy = 0;

const int xAxisBoxHeight = 40;
const int yAxisBoxWidth = 100;
const int boxPadding = 2;

private Color backgroundColor = Color.White;
private Pen axisPen = Pens.Black;
private Pen spectrumPen = Pens.Navy;
private Brush ROIsBrush = new SolidBrush(Color.LemonChiffon);
private Brush spectrumROIsBrush = new SolidBrush(Color.GreenYellow);
    private Brush spectrumROIsBackgroundBrush = new SolidBrush(Color.OrangeRed);
private Pen peaksPen = new Pen(new SolidBrush(Color.FromArgb(127, Color.CadetBlue)));

private Rectangle xAxisBox = new Rectangle(0, 0, 1, 1);
private Rectangle yAxisBox = new Rectangle(0, 0, 1, 1);
private Rectangle spectrumBox = new Rectangle(0, 0, 1, 1);

private Bitmap xAxisBmp;
private Bitmap yAxisBmp;
private Bitmap spectrumBmp;

private xAxisModes xAxisMode = xAxisModes.Channel;
private yAxisModes yAxisMode = yAxisModes.Linear;

```

```

private zoomModes zoomMode = zoomModes.auto, previousZoomMode = zoomModes.auto;
private uint zoomMinChannel = 0, zoomMaxChannel = 1023;
private ulong zoomMinCounts = 0, zoomMaxCounts = 10;
private uint zoomStartChannel = 0, zoomEndChannel = 0;
private ulong zoomStartCounts = 0, zoomEndCounts = 0;

private uint cursorPosition = 0;

private IMCA mca;

private bool remoteConnected = false;           //connected to the remote server
private bool remoteOvertaken = false, remoteOvertakenMCA = false; //no user interaction, MCA available
private bool remoteCountAndSave = false;        //count and save commanded
private bool remoteClosed = false;
private bool closed = false;

private string scanDirectory = "", scanName = "", scanFileNumber = ""; //base scan dir, current scan name and file number

private bool connected = false;                 //controller is connected

private ClassSpectrum spectrum;
private ListOfROI ROIs;
private Calibration calibration;

private uint acquireTime = 0, acquireTimeNew = 0;
private float realtime = 0, livetime = 0, deadtimePercent = 0;
private  $\mu$ XRF.Enums.AcquisitionTimers acquireTimer =  $\mu$ XRF.Enums.AcquisitionTimers.Realtime;

bool counting = false, fileLoaded = false;

private List<string> header = new List<string>();
private DateTime spectrumDateTime = new DateTime();

private bool saveSecondary = true, saveShort = false, saveSecondaryShort = true, saveSumSpectrum = true;
private SpectrumFiles.writeFormats autoWriteFormat = SpectrumFiles.writeFormats.txt, secondaryWriteFormat =
SpectrumFiles.writeFormats.spe;

private channelEnterByCursorModes channelEnterByCursor = channelEnterByCursorModes.none; //select channel with the spectrum
display

private bool textEditROI = false;               //editing ROI texts
private bool textEditCalibration = false;       //editing Calibration texts
private bool textEditHeader = false;           //editing Header text

private Rectangle formLocationAndSize;

private void StartPostitionBtn_Click(object sender, EventArgs e)
{
    movemotor(1, 0, 1, 4);
    movemotor(2, 0, 1, 4);
}

```

```

private void gotopositionBtn_Click(object sender, EventArgs e)
{
    int yposition = Convert.ToInt32(Ypositon.Text);
    int xposition = Convert.ToInt32(Xpositon.Text);
    movemotor(2, yposition, 0, 2);
    movemotor(1, xposition, 0, 2);
}

private void movemotor(int motor, int step, int direct, int mode)
{
    if (motor == 1)
    {
        // Set comm settings for motor 2

        motor1.SelectedPort = ComPortBox1.Text;
        motor1.Baudrate = Convert.ToInt32(BaudrateBox1.Text);
        // Set motor address
        motor1.MotorAdresse = Convert.ToInt32(Motor1ID.Value);
        motor1.SetSteps(step);
        motor1.SetPositionType(mode);
        motor1.SetDirection(direct);
        motor1.SetRampType(1);
        motor1.StartTravelProfile();
    }
    else if (motor == 2)
    {
        motor2.SelectedPort = ComPortBox1.Text;

        //motor2.SelectedPort = COM4;// ComPortBox1.Text;
        motor2.Baudrate = Convert.ToInt32(BaudrateBox1.Text);

        motor2.MotorAdresse = Convert.ToInt32(Motor2ID.Value);
        motor2.SetSteps(step);
        motor2.SetPositionType(mode);
        motor2.SetDirection(direct);
        motor2.SetRampType(1);
        motor2.StartTravelProfile();
    }
}

private void gostepandwait()
{
    movemotor(1, 500, 0, 1);
    while (!motor1.IsMotorReady());

    /*

    for (int i = 0; i <= 3; i++)
    {

        if (end == true)
        {
            goto Ende;
        }
        else
        {

            if (!busymca)

```

```

    {

        movemotor(1, 500, 1, 1);
        busy = true;
    }

    while (busy)
    {
        if (motor1.IsMotorReady())
        {
            busy = false;
        }
    }

    RemoteCountAndSave("C:\\Documents and Settings\\XPMUser\\Desktop\\data~", "filone", "test1",
μXRF.Enums.AcquisitionTimers.Realtime, 10, new List<string>());

    }
    Ende:
    {
        end = false;
    }

    }
    */

}

public void linearemessung(object sender, EventArgs e)
{
    //  gostepandwait();
    RemoteCountAndSave("C:\\Documents and Settings\\XPMUser\\Desktop\\data~", "filone", "test1",
μXRF.Enums.AcquisitionTimers.Realtime, 10, new List<string>());

}
private void XtopBtn_Click(object sender, EventArgs e)
{
    movemotor(1, 500, 1, 1);
}
private void XbotBtn_Click(object sender, EventArgs e)
{
    movemotor(1, 500, 0, 1);

}
private void YtopBtn_Click(object sender, EventArgs e)
{
    movemotor(2, 500, 0, 1);
}

```

```

    }
    private void YbotBtn_Click(object sender, EventArgs e)
    {

        movemotor(2, 500, 1, 1);

    }
    public void StartBtn_Click(object sender, EventArgs e)
    {

        {

            // int yschritte = Convert.ToInt32(yschritt.Text);

            // int yschrittegrosesse = Convert.ToInt32(yschrittgro.Text);

            // for (int y = 0; y < 2; y++)
            {

                /* if (this.backlash.Checked)
                {
                    movemotor(2, 500, 0, 1);
                    movemotor(2, 500, 1, 1);
                }

                */

                RemoteCountAndSave("C:\\Documents and Settings\\XPMUser\\Desktop\\data", "filone", "test1",
                μXRF.Enums.AcquisitionTimers.Realtime, 10, new List<string>());

            }

            // movemotor(2, 0, 0, 2);

            /*
            if (busy == true)
            {

                Statusready.Text = "true";
            }
            else
            {
                Statusready.Text = "false";
            }

            if (motor1.ErrorFlag)
            {
                StatusLabel1.Text = "Status 1: " + motor1.ErrorMessageString;
            }
            else
            {
                StatusLabel1.Text = "Status 1: OK";
            }
            if (motor2.ErrorFlag)
            {

```



```

        StatusLabel2.Text = "Status 2: " + motor2.ErrorMessageString;
    }
    else
    {
        StatusLabel2.Text = "Status 2: OK";
    }

    */

}
}
private void StopBtn_Click(object sender, EventArgs e)
{
    // Stop travel profile
    motor1.StopTravelProfile();
    motor2.StopTravelProfile();
}

private void calculateBoxSizes()
{
    yAxisBox = new Rectangle(boxPadding, boxPadding, yAxisBoxWidth, pictureBoxSpectrum.ClientSize.Height - xAxisBoxHeight - 4 *
boxPadding);
    xAxisBox = new Rectangle(yAxisBox.Right + boxPadding, yAxisBox.Bottom + boxPadding, pictureBoxSpectrum.ClientSize.Width -
yAxisBoxWidth - 3 * boxPadding, xAxisBoxHeight);
    spectrumBox = new Rectangle(yAxisBox.Right + boxPadding, boxPadding, pictureBoxSpectrum.ClientSize.Width - yAxisBoxWidth - 3 *
boxPadding, pictureBoxSpectrum.ClientSize.Height - xAxisBoxHeight - 4 * boxPadding);

    xAxisPaint();
    yAxisPaint();
    spectrumPaint();
}

private string autoChannelOrEnergyString(uint channel)
{
    if (xAxisMode == xAxisModes.Channel)
    {
        return channel.ToString();
    }
    else
    {
        return Math.Round(energyFromChannel(channel), 2).ToString("F2");
    }
}

private void xAxisPaint()
{
    xAxisBmp = new Bitmap(xAxisBox.Width, xAxisBox.Height);
    Graphics gr = Graphics.FromImage(xAxisBmp);
    gr.SmoothingMode = System.Drawing.Drawing2D.SmoothingMode.AntiAlias;

    gr.Clear(backgroundColor);
    gr.DrawLine(axisPen, new Point(0, 0), new Point(xAxisBox.Width - 1, 0));

    Font xFont = new Font("Arial", 10);
    Brush xBrush = Brushes.Black;
    StringFormat xStringFormat = new StringFormat();

```

```

xStringFormat.Alignment = StringAlignment.Center;
xStringFormat.LineAlignment = StringAlignment.Far;
gr.DrawString(xAxisMode.ToString(), xFont, xBrush, new PointF(xAxisBox.Width / 2, xAxisBox.Height), xStringFormat);

xStringFormat.LineAlignment = StringAlignment.Near;

xStringFormat.Alignment = StringAlignment.Near;
gr.DrawString(autoChannelOrEnergyString(zoomMinChannel), xFont, xBrush, new PointF(0, 5), xStringFormat);
textBoxEditMinChannel.Location = new Point(pictureBoxSpectrum.Location.X + xAxisBox.Left + 4, pictureBoxSpectrum.Location.Y +
xAxisBox.Top + 7);
gr.DrawLine(axisPen, new Point(0, 0), new Point(0, 5));

xStringFormat.Alignment = StringAlignment.Far;
gr.DrawString(autoChannelOrEnergyString(zoomMaxChannel), xFont, xBrush, new PointF(xAxisBox.Width - 1, 5), xStringFormat);
textBoxEditMaxChannel.Location = new Point(pictureBoxSpectrum.Location.X + xAxisBox.Right - textBoxEditMaxChannel.Width,
pictureBoxSpectrum.Location.Y + xAxisBox.Top + 7);
gr.DrawLine(axisPen, new Point(xAxisBox.Width - 1, 0), new Point(xAxisBox.Width - 1, 5));

xStringFormat.Alignment = StringAlignment.Center;

int parter = 6;

if (zoomMaxChannel - zoomMinChannel < parter)
{
    parter = (int)(zoomMaxChannel - zoomMinChannel);
}

for (int i = 1; i <= parter - 1; i++)
{
    uint posX = (uint)((i * (zoomMaxChannel - zoomMinChannel) / parter) + zoomMinChannel);
    gr.DrawLine(axisPen, new Point(xFromChannel(posX), 0), new Point(xFromChannel(posX), 5));
    gr.DrawString(autoChannelOrEnergyString((uint)(posX)), xFont, xBrush, new PointF(xFromChannel(posX), 5), xStringFormat);
}
}

private void yAxisPaint()
{
    yAxisBmp = new Bitmap(yAxisBox.Width, yAxisBox.Height);
    Graphics gr = Graphics.FromImage(yAxisBmp);
    gr.SmoothingMode = System.Drawing.Drawing2D.SmoothingMode.AntiAlias;

    gr.Clear(background.Color);
    gr.DrawLine(axisPen, new Point(yAxisBox.Width - 1, 0), new Point(yAxisBox.Width - 1, yAxisBox.Height - 1));

    Font yFont = new Font("Arial", 10);
    Brush yBrush = Brushes.Black;
    StringFormat yStringFormat = new StringFormat();

    yStringFormat.Alignment = StringAlignment.Center;
    yStringFormat.LineAlignment = StringAlignment.Center;
    gr.DrawString("C\\no\\nu\\nn\\nt\\ns", yFont, yBrush, new PointF(10, yAxisBox.Height / 2), yStringFormat);

    yStringFormat.Alignment = StringAlignment.Far;

    yStringFormat.LineAlignment = StringAlignment.Near;
    gr.DrawString(zoomMaxCounts.ToString(), yFont, yBrush, new PointF(yAxisBox.Width - 6, 0), yStringFormat);
    textBoxEditMaxCounts.Location = new Point(pictureBoxSpectrum.Location.X + yAxisBox.Right - textBoxEditMaxCounts.Width - 5,
pictureBoxSpectrum.Location.Y + spectrumBox.Top + boxPadding);
    gr.DrawLine(axisPen, new Point(yAxisBox.Width - 1, 0), new Point(yAxisBox.Width - 6, 0));

```

```

yStringFormat.LineAlignment = StringAlignment.Far;
gr.DrawString(zoomMinCounts.ToString(), yFont, yBrush, new PointF(yAxisBox.Width - 6, yAxisBox.Height - 1), yStringFormat);
textBoxEditMinCounts.Location = new Point(pictureBoxSpectrum.Location.X + yAxisBox.Right - textBoxEditMaxCounts.Width - 5,
pictureBoxSpectrum.Location.Y + spectrumBox.Top + boxPadding + yAxisBox.Height - textBoxEditMinCounts.Size.Height - 2);
gr.DrawLine(axisPen, new Point(yAxisBox.Width - 1, yAxisBox.Height - 1), new Point(yAxisBox.Width - 6, yAxisBox.Height - 1));

yStringFormat.LineAlignment = StringAlignment.Center;

switch (yAxisMode)
{
    case yAxisModes.Linear:
        uint linParter = 10;

        if (zoomMaxCounts - zoomMinCounts < linParter)
        {
            linParter = (uint)(zoomMaxCounts - zoomMinCounts);
        }

        for (int i = 1; i <= linParter - 1; i++)
        {
            uint posY = (uint)(i * (int)((zoomMaxCounts - zoomMinCounts) / linParter) + (int)zoomMinCounts);
            gr.DrawLine(axisPen, new Point(yAxisBox.Width - 1, yFromCounts(posY)), new Point(yAxisBox.Width - 10,
yFromCounts(posY)));
            gr.DrawString(posY.ToString(), yFont, yBrush, new PointF(yAxisBox.Width - 6, yFromCounts(posY)), yStringFormat);
        }

        break;
    case yAxisModes.Logarithmic:
        uint logParter = (uint)(Math.Log10(zoomMaxCounts) - Math.Log10(zoomMinCounts));

        if (logParter > 1)
        {
            for (int i = 1; i <= logParter - 1; i++)
            {
                float posY = (float)(Math.Log10(zoomMaxCounts) - Math.Log10(zoomMinCounts)) * i / logParter +
(float)Math.Log10(zoomMinCounts);
                posY = (float)Math.Pow(10, posY);
                gr.DrawLine(axisPen, new Point(yAxisBox.Width - 1, yFromCounts((uint)posY)), new Point(yAxisBox.Width - 10,
yFromCounts((uint)posY)));
                gr.DrawString(posY.ToString("F0"), yFont, yBrush, new PointF(yAxisBox.Width - 6, yFromCounts((uint)posY)),
yStringFormat);
            }
        }

        break;
}

private void spectrumPaint()
{
    if (this.WindowState != FormWindowState.Minimized)
    {
        spectrumBmp = new Bitmap(spectrumBox.Width, spectrumBox.Height);
        Graphics gr = Graphics.FromImage(spectrumBmp);
        gr.SmoothingMode = System.Drawing.Drawing2D.SmoothingMode.AntiAlias;

        gr.Clear(background-color);

        foreach (ROI element in ROIs)
        {
            if (element.Check(spectrum.Size))

```

```

{
    int xStart = xFromChannel(element.Start);
    int xEnd = xFromChannel(element.End);
    int width = xEnd - xStart;
    Rectangle ROIrectangle = new Rectangle(new Point(xStart, 0), new Size(width, spectrumBox.Height));
    gr.FillRectangle(ROIsBrush, ROIrectangle);

    Point[] ROIpolygon = new Point[element.End - element.Start + 2];
    for (int i = 0; i <= element.End - element.Start; i++)
    {
        ROIpolygon[i] = new Point(xFromChannel((uint)(i + element.Start)), yFromCounts(spectrum.Data[i + element.Start]));
    }
    ROIpolygon[ROIpolygon.Length - 2] = new Point(xFromChannel(element.End), spectrumBox.Height);
    ROIpolygon[ROIpolygon.Length - 1] = new Point(xFromChannel(element.Start), spectrumBox.Height);
    gr.FillPolygon(spectrumROIsBrush, ROIpolygon);

    Point[] backPolygon = new Point[4];
    backPolygon[0] = new Point(xFromChannel(element.Start), spectrumBox.Height);
    backPolygon[1] = new Point(xFromChannel(element.Start), yFromCounts(spectrum.Data[element.Start]));
    backPolygon[2] = new Point(xFromChannel(element.End), yFromCounts(spectrum.Data[element.End]));
    backPolygon[3] = new Point(xFromChannel(element.End), spectrumBox.Height);
    gr.FillPolygon(spectrumROIsBackgroundBrush, backPolygon);

    StringFormat nameStringFormat = new StringFormat();
    nameStringFormat.Alignment = StringAlignment.Center;
    nameStringFormat.LineAlignment = StringAlignment.Far;
    gr.DrawString(element.Name, new Font("Arial", 8), Brushes.Black, new PointF(xStart + width / 2, spectrumBox.Top + 10),
nameStringFormat);
}
}

Point[] spectrumLinePoints = new Point[spectrum.Size];

for (uint i = 0; i < spectrum.Size; i++)
{
    spectrumLinePoints[i] = new Point(xFromChannel(i), yFromCounts(spectrum.Data[i]));
}
gr.DrawLines(spectrumPen, spectrumLinePoints);
}
}

private void autoZoom()
{
    zoomMinChannel = 0;
    zoomMaxChannel = spectrum.Size - 1;

    if (yAxisMode == yAxisModes.Linear)
    {
        zoomMinCounts = 0;

        long maxCountsInSpectrum = 9;
        foreach (long element in spectrum.Data)
        {
            if (element > maxCountsInSpectrum) maxCountsInSpectrum = element;
        }

        int numberOfDigits = maxCountsInSpectrum.ToString().Length;
        uint multiplier = (uint)Math.Pow(10, numberOfDigits - 1);

```

```

        zoomMaxCounts = (uint.Parse(maxCountsInSpectrum.ToString().Substring(0, 1)) + 1) * multiplier;
    }
    else if (yAxisMode == yAxisModes.Logarithmic)
    {
        zoomMinCounts = 1;

        long maxCountsInSpectrum = 9;
        foreach (long element in spectrum.Data)
        {
            if (element > maxCountsInSpectrum) maxCountsInSpectrum = element;
        }

        int numberOfDigits = maxCountsInSpectrum.ToString().Length;

        zoomMaxCounts = (uint)Math.Pow(10, numberOfDigits);
    }
}

private float energyFromChannel(uint channel)
{
    return channel * calibration.Gain + calibration.Offset;
}

private uint channelFromEnergy(float energy)
{
    return (uint)((energy - calibration.Offset) / calibration.Gain);
}

private int xFromChannel(uint channel)
{
    float channelStepWidth = (float)(spectrumBox.Width - 1) / (float)(zoomMaxChannel - zoomMinChannel);
    float returner = ((float)channel - (float)zoomMinChannel) * channelStepWidth;
    return (int)returner;
}

private uint channelFromX(int x)
{
    float channelStepWidth = (float)(spectrumBox.Width - 1) / (float)(zoomMaxChannel - zoomMinChannel);
    float returner = x / channelStepWidth + zoomMinChannel;
    return (uint)Math.Round(returner);
}

private int yFromCounts(ulong counts)
{
    float returner = 0;
    float yDisplayRange = spectrumBox.Height - 1;

    switch (yAxisMode)
    {
        case yAxisModes.Linear:
            ulong countsDisplayRangeLin = zoomMaxCounts - zoomMinCounts;
            returner = yDisplayRange - ((float)counts - (float)zoomMinCounts) * (yDisplayRange / (float)countsDisplayRangeLin);
            break;
        case yAxisModes.Logarithmic:
            if (counts == 0)
            {
                returner = yDisplayRange;
            }
    }
}

```

```

    }
    else
    {
        float countsDisplayRangeLog = (float)(Math.Log10(zoomMaxCounts) - Math.Log10(zoomMinCounts));
        returner = yDisplayRange - ((float)Math.Log10(counts) - (float)Math.Log10(zoomMinCounts)) * (yDisplayRange /
(float)countsDisplayRangeLog);
    }
    break;
}
return (int)returner;
}

private ulong countsFromY(int y)
{
    int yDisplayRange = spectrumBox.Height - 1;
    ulong countsDisplayRangeLin = zoomMaxCounts - zoomMinCounts;
    ulong returner = 0;

    switch (yAxisMode)
    {
        case yAxisModes.Linear:
            returner = countsDisplayRangeLin - (ulong)Math.Round((float)countsDisplayRangeLin / yDisplayRange * (float)y);
            returner += zoomMinCounts;
            break;
        case yAxisModes.Logarithmic:
            float countsDisplayRangeLog = (float)(Math.Log10(zoomMaxCounts) - Math.Log10(zoomMinCounts));
            int yInInv = yDisplayRange - y;
            float toPow = (float)yInInv / (float)yDisplayRange * countsDisplayRangeLog + zoomMinCounts;
            returner = (ulong)Math.Pow(10.0, toPow - 1);
            break;
    }
    return returner;
}

private void checkMinMax(ref int makeMin, ref int makeMax)
{
    int temp = 0;

    if (makeMin > makeMax)
    {
        temp = makeMax;
        makeMax = makeMin;
        makeMin = temp;
    }
}

private void checkMinMax(ref uint makeMin, ref uint makeMax)
{
    uint temp = 0;

    if (makeMin > makeMax)
    {
        temp = makeMax;
        makeMax = makeMin;
        makeMin = temp;
    }
}

private void checkMinMax(ref ulong makeMin, ref ulong makeMax)
{

```

```

        ulong temp = 0;

        if (makeMin > makeMax)
        {
            temp = makeMax;
            makeMax = makeMin;
            makeMin = temp;
        }
    }

    private float calcDeadTimePercent(float rt, float lt)
    {
        if (rt > 0)
        {
            return (rt - lt) / rt * 100;
        }
        else
        {
            return 0;
        }
    }

    private void newSpectrum(bool clearHeader)
    {
        ulong[] data = new ulong[1024];
        data.Initialize();

        if (clearHeader)
        {
            header = new List<string>();
            listBoxHeader.Items.Clear();
            textBoxHeader.Text = "";
        }

        changeSpectrumData(data, 0, 0, 0);

        fileLoaded = false;

        spectrumDateTime = DateTime.Now;
        dateTimePickerSpectrumDateTime.Value = DateTime.Now;

        this.Text = "X-Spect";
    }

    private void openSpectrum(SpectrumFiles.readFormats format, string file)
    {
        try
        {
            ulong[] data = SpectrumFiles.read(format, file, out header, out realtime, out livetime, out spectrumDateTime);

            if (data == null) throw new Exception();

            fileLoaded = true;

            dateTimePickerSpectrumDateTime.Value = spectrumDateTime;

            listBoxHeader.Items.Clear();
            textBoxHeader.Text = "";
            foreach (string element in header)
            {

```

```

        listBoxHeader.Items.Add(element);
    }

    changeSpectrumData(data, realtime, livetime, 0);

    this.Text = "X-Spect [" + file + "]";
}
catch
{
    newSpectrum(true);

    throw new Exception("Error loading file!");
}
}

private void changeSpectrumData(ulong[] data, float realtime, float livetime, uint percentageComplete)
{
    spectrum.Data = data;

    this.realtime = realtime;
    this.livetime = livetime;

    deadtimePercent = calcDeadTimePercent(realtime, livetime);

    if (cursorPosition >= spectrum.Size)
    {
        cursorPosition = spectrum.Size - 1;
    }

    if (zoomMaxChannel >= spectrum.Size)
    {
        zoomMaxChannel = spectrum.Size - 1;
    }

    if (zoomMode == zoomModes.auto)
    {
        autoZoom();
        xAxisPaint();
        yAxisPaint();
    }

    progressBarAcquire.Value = (int)percentageComplete;

    displayReport();
    spectrumPaint();

    pictureBoxSpectrum.Invalidate();

/*    if ((int)percentageComplete == 100)
    {
        //end = true;
        busymca = false;

    }
    else
    {
        busymca = true;
    }
*/
}

```



```

}

private void cursorPositionChanged()
{
    displayReport();
    pictureBoxSpectrum.Invalidate();

    if (channelEnterByCursor == channelEnterByCursorModes.calibration)
    {
        textBoxCalibrationChannel.Text = cursorPosition.ToString();
    }
    else if (channelEnterByCursor == channelEnterByCursorModes.ROIstart)
    {
        textBoxROIStartChannel.Text = cursorPosition.ToString();
    }
    else if (channelEnterByCursor == channelEnterByCursorModes.ROIend)
    {
        textBoxROIEndChannel.Text = cursorPosition.ToString();
    }
}

private void displayReport()
{
    float countrate = 0;
    if (livetime > 0)
    {
        countrate = spectrum.SumCounts / livetime;
    }

    string newReport = "";
    newReport += "--- Cursor Position ---" + Environment.NewLine;
    newReport += "Channel: " + cursorPosition.ToString() + Environment.NewLine;
    newReport += "Energy: " + energyFromChannel(cursorPosition).ToString("F2") + "keV" + Environment.NewLine;
    newReport += "Counts: " + spectrum.Data[cursorPosition].ToString() + Environment.NewLine + Environment.NewLine;

    newReport += "--- Acquisition ---" + Environment.NewLine;
    newReport += "Channels: " + spectrum.Size.ToString() + Environment.NewLine;
    newReport += "Sum Counts: " + spectrum.SumCounts.ToString() + Environment.NewLine;
    newReport += "Count Rate: " + countrate.ToString("F2") + "cps" + Environment.NewLine;
    newReport += "Real Time: " + realtime.ToString("F1") + "sec." + Environment.NewLine;
    newReport += "Live Time: " + livetime.ToString("F1") + "sec." + Environment.NewLine;
    newReport += "Dead Time: " + deadtimePercent.ToString("F1") + "%" + Environment.NewLine + Environment.NewLine;

    newReport += "--- ROIs ---" + Environment.NewLine;
    newReport += "Number of ROIs: " + ROIs.Count.ToString() + Environment.NewLine + Environment.NewLine;

    newReport += ROIs.GetReport(spectrum.Data, calibration.Gain);

    textBoxReport.Text = newReport;
}

private void displayCalibration(bool correct)
{
    if (!correct)
    {
        textBoxCalibrationGain.BackColor = Color.Red;
        textBoxCalibrationGain.Text = "Gain=1";
        textBoxCalibrationOffset.BackColor = Color.Red;
    }
}

```

```

        textBoxCalibrationOffset.Text = "Offset=0";
    }
    else
    {
        textBoxCalibrationGain.BackColor = Color.Empty;
        textBoxCalibrationGain.Text = "Gain=" + calibration.Gain.ToString("F4");
        textBoxCalibrationOffset.BackColor = Color.Empty;
        textBoxCalibrationOffset.Text = "Offset=" + calibration.Offset.ToString("F4");
    }
}

private void saveReport(string file)
{
    using (System.IO.StreamWriter sw = new System.IO.StreamWriter(file))
    {
        string reportText = textBoxReport.Text;
        reportText = reportText.Substring(reportText.IndexOf("--- Acquisition ---"));
        sw.Write(reportText);
    }
}

private void acquisitionStart()
{
    if (fileLoaded) newSpectrum(true);

    // busy = true;
    buttonAcquireStartStop.Text = "stop";

    spectrumDateTime = DateTime.Now;
    dateTimePickerSpectrumDateTime.Value = DateTime.Now;

    acquisitionControlsDisable();

    counting = true;

    mca.mcaStartAcquire();
}

private void acquisitionStop()
{
    //busy = false;
    buttonAcquireStartStop.Text = "start";
    labelAcquisitionStatus.Text = "Ready.";
    counting = false;

    if (!remoteOvertaken)
    {
        acquisitionControlsEnable();
    }
    else
    if (remoteCountAndSave)
    {
        string scanCurrentDirectory = System.IO.Path.Combine(scanDirectory, scanName);
        string reportFile = System.IO.Path.Combine(scanCurrentDirectory, "reports");
        if (!System.IO.Directory.Exists(reportFile)) System.IO.Directory.CreateDirectory(reportFile);
        reportFile = System.IO.Path.Combine(reportFile, scanName + scanFilenameNumber + "-report.txt");

        using (System.IO.StreamWriter sw = new System.IO.StreamWriter(reportFile))

```

```

{
    string reportText = textBoxReport.Text;
    reportText = reportText.Substring(reportText.IndexOf("--- Acquisition ---"));
    sw.Write(reportText);
}

string primarySpectrumFile = System.IO.Path.Combine(scanCurrentDirectory, "spectra");
primarySpectrumFile = System.IO.Path.Combine(primarySpectrumFile, autoWriteFormat.ToString());
if (!System.IO.Directory.Exists(primarySpectrumFile)) System.IO.Directory.CreateDirectory(primarySpectrumFile);
if (saveShort)
{
    primarySpectrumFile = System.IO.Path.Combine(primarySpectrumFile, scanFilename + "." + autoWriteFormat.ToString());
}
else
{
    primarySpectrumFile = System.IO.Path.Combine(primarySpectrumFile, scanName + scanFilename + "." +
autoWriteFormat.ToString());
}

SpectrumFiles.write(autoWriteFormat, primarySpectrumFile, header, spectrum.Data, realtime, livetime, DateTime.Now);

if (saveSecondary)
{
    string secondarySpectrumFile = System.IO.Path.Combine(scanCurrentDirectory, "spectra");
    secondarySpectrumFile = System.IO.Path.Combine(secondarySpectrumFile, secondaryWriteFormat.ToString());
    if (!System.IO.Directory.Exists(secondarySpectrumFile)) System.IO.Directory.CreateDirectory(secondarySpectrumFile);
    if (saveSecondaryShort)
    {
        secondarySpectrumFile = System.IO.Path.Combine(secondarySpectrumFile, scanFilename + "." +
secondaryWriteFormat.ToString());
    }
    else
    {
        secondarySpectrumFile = System.IO.Path.Combine(secondarySpectrumFile, scanName + scanFilename + "." +
secondaryWriteFormat.ToString());
    }

    SpectrumFiles.write(secondaryWriteFormat, secondarySpectrumFile, header, spectrum.Data, realtime, livetime, DateTime.Now);
}

if (saveSumSpectrum)
{
    ulong[] sumSpectrum;
    List<string> sumHeader;
    float sumRealtime, sumLivetime;
    DateTime sumDateTime;

    string sumSpectrumPrimaryFile = System.IO.Path.Combine(scanCurrentDirectory, "spectra");
    sumSpectrumPrimaryFile = System.IO.Path.Combine(sumSpectrumPrimaryFile, "sum." + autoWriteFormat.ToString());
    if (System.IO.File.Exists(sumSpectrumPrimaryFile))
    {
        SpectrumFiles.readFormats sumReadFormat = (SpectrumFiles.readFormats)Enum.Parse(typeof(SpectrumFiles.readFormats),
autoWriteFormat.ToString());

        sumSpectrum = SpectrumFiles.read(sumReadFormat, sumSpectrumPrimaryFile, out sumHeader, out sumRealtime, out
sumLivetime, out sumDateTime);
    }
    else
    {
        sumSpectrum = new ulong[spectrum.Size];
    }
}

```

```

        sumHeader = new List<string>();
        sumHeader.Add("Sum Spectrum");
        sumHeader.Add("Scanname: " + scanName);
        sumRealtime = 0;
        sumLivetime = 0;
        sumDateTime = new DateTime();
        sumDateTime = DateTime.Now;
    }

    sumRealtime += realtime;
    sumLivetime += livetime;
    for (int i = 0; i < sumSpectrum.Length; i++)
    {
        if (i < spectrum.Size) sumSpectrum[i] += spectrum.Data[i];
    }

    SpectrumFiles.write(autoWriteFormat, sumSpectrumPrimaryFile, sumHeader, sumSpectrum, sumRealtime, sumLivetime,
sumDateTime);

    if (saveSecondary)
    {
        string sumSpectrumSecondaryFile = System.IO.Path.Combine(scanCurrentDirectory, "spectra");
        sumSpectrumSecondaryFile = System.IO.Path.Combine(sumSpectrumSecondaryFile, "sum." + secondaryWriteFormat.ToString());
        SpectrumFiles.write(secondaryWriteFormat, sumSpectrumSecondaryFile, sumHeader, sumSpectrum, sumRealtime, sumLivetime,
sumDateTime);
    }

    }
    onCounted();

}

}

}

private void acquisitionControlsDisable()
{
    buttonAcquireClear.Enabled = false;
    groupBoxAcquisitionTime.Enabled = false;
    groupBoxAcquisitionTimer.Enabled = false;

    menuFile.Enabled = false;
    menuMCA.Enabled = false;

    tabControlSide.Controls.Remove(tabPageROIs);
    tabControlSide.Controls.Remove(tabPageCalibration);
    tabControlSide.Controls.Remove(tabPageHeader);
    tabControlSide.Controls.Remove(tabPageOptions);
}

private void acquisitionControlsEnable()
{
    buttonAcquireClear.Enabled = true;
    groupBoxAcquisitionTime.Enabled = true;
    groupBoxAcquisitionTimer.Enabled = true;

    menuFile.Enabled = true;

```

```

menuMCA.Enabled = true;

tabControlSide.Controls.Add(tabPageROIs);
tabControlSide.Controls.Add(tabPageCalibration);
tabControlSide.Controls.Add(tabPageHeader);
tabControlSide.Controls.Add(tabPageOptions);

rep++;
if (rep < Convert.ToInt32(xschrift.Text))
{
    gostepandwait();
    linear.PerformClick();
}
else
{
    repy++;
    if (repy < Convert.ToInt32(yschrift.Text))
    {
        rep = 0;
        movemotor(2, 500, 0, 1);
        movemotor(1, 0, 0, 2);
        linear.PerformClick();

    }
    else
    {
        repy = 0;
        rep = 0;
        movemotor(1, 0, 0, 2);
        movemotor(2, 0, 0, 2);
    }

}

}

private void mca_mcaAcquisitionEnd(object sender, mcaDataUpdateEventArgs e)
{
    this.Invoke(changeSpectrumDataDel, e.Data, e.Realtime, e.Livetime, e.PercentComplete);
    this.Invoke(displayReportDel);
    this.Invoke(acquisitionStopDel);

}

private void mca_mcaDataUpdate(object sender, mcaDataUpdateEventArgs e)
{
    try
    {
        this.Invoke(changeSpectrumDataDel, e.Data, e.Realtime, e.Livetime, e.PercentComplete);
        this.Invoke(displayReportDel);
    }
    catch
    {
        //da passt was ned!! division by zero!!
    }
}

```

```
protected virtual void onCounted()
{
    if (Counted != null)
    {
        Counted(this, new CountedEventArgs(ROIs, spectrum.Data));
    }
}
```

```
protected virtual void onMCAError()
{
    if (MCAError != null)
    {
        MCAError(this, new EventArgs());
    }
}
```

```
public bool WasClosed
{
    get
    {
        return closed;
    }
}
```

```
public bool RemoteConnected
{
    get
    {
        return remoteConnected;
    }
    set
    {
        remoteConnected = value;
    }
}
```

```
public bool RemoteClosed
{
    get
    {
        return remoteClosed;
    }
    set
    {
        remoteClosed = value;
    }
}
```

```
public bool Connected
{
    get
    {
        return connected;
    }
}
```

```
public bool RemoteOvertaken
{
    get
    {
```

```

        return remoteOvertaken && ROIs.CheckNamesValidity();
    }
}

public bool RemoteOvertakenMCA
{
    get
    {
        return remoteOvertakenMCA && ROIs.CheckNamesValidity();
    }
}

public string ScanDirectory
{
    get
    {
        return scanDirectory;
    }
    set
    {
        scanDirectory = value;
    }
}

public string ScanName
{
    get
    {
        return scanName;
    }
    set
    {
        scanName = value;
    }
}

public string ScanFileumber
{
    get
    {
        return scanFileumber;
    }
    set
    {
        scanFileumber = value;
    }
}

public bool Takeover(bool MCA)
{
    if (MCA && !connected)
    {
        return false;
    }
    else if (!counting)
    {
        remoteOvertaken = true;
        remoteOvertakenMCA = MCA;

        menuFile.Enabled = false;
        menuMCA.Enabled = false;
        menuSaveBitmap.Enabled = false;
    }
}

```

```

        menuQuestion.Enabled = false;

        tabControlSide.TabPages.Clear();
        if (connected) tabControlSide.TabPages.Add(tabPageAcquire);
        tabControlSide.Invalidate();

        buttonAcquireStartStop.Enabled = false;
        buttonAcquireClear.Enabled = false;
        groupBoxAcquisitionTime.Enabled = false;
        groupBoxAcquisitionTimer.Enabled = false;

        return true;
    }
    else
    {
        return false;
    }
}

public bool Release()
{
    if (remoteOvertaken)
    {
        remoteCountAndSave = false;

        if (counting) mca.mcaStopAcquire();

        remoteOvertaken = false;

        menuFile.Enabled = true;
        menuMCA.Enabled = true;
        menuSaveBitmap.Enabled = true;
        menuQuestion.Enabled = true;

        tabControlSide.Controls.Add(tabPageROIs);
        tabControlSide.Controls.Add(tabPageCalibration);
        tabControlSide.Controls.Add(tabPageHeader);
        tabControlSide.Controls.Add(tabPageOptions);
        tabControlSide.Invalidate();

        buttonAcquireStartStop.Enabled = true;
        buttonAcquireClear.Enabled = true;
        groupBoxAcquisitionTime.Enabled = true;
        groupBoxAcquisitionTimer.Enabled = true;

        return true;
    }
    else
    {
        return false;
    }
}

public void RemoteCountAndSave(string ScanDirectory, string ScanName, string ScanFilenumber,  $\mu$ XRF.Enums.AcquisitionTimers
AcquisitionTimer, uint AcquisitionTime, List<string> HeaderInfo)
{
    if (connected && !counting)
    {
        scanDirectory = ScanDirectory;
        scanName = ScanName;
        scanFilenumber = ScanFilenumber;
    }
}

```



```

        acquireTimer = AcquisitionTimer;
        acquireTime = AcquisitionTime;
        textBoxAcquisitionTime.Text = acquireTime.ToString();
        groupBoxAcquisitionTime.Text = "Acquisition Time: " + acquireTime.ToString() + "sec.";
        textBoxAcquisitionTime.BackColor = Color.Empty;
        if (acquireTimer ==  $\mu$ XRF.Enums.AcquisitionTimers.Realtime)
        {
            radioButtonAcquisitionReal.Checked = true;
            radioButtonAcquisitionLive.Checked = false;
            groupBoxAcquisitionTimer.Text = "Acquisition Timer: Realtime";
        }
        else
        {
            radioButtonAcquisitionReal.Checked = false;
            radioButtonAcquisitionLive.Checked = true;
            groupBoxAcquisitionTimer.Text = "Acquisition Timer: Livetime";
        }

        mca.mcaClear();

        mca.mcaSetPresetTimer(acquireTimer);
        mca.mcaSetPresetTime(acquireTime);

        newSpectrum(false);
        buttonAcquireStartStop.Text = "Stop";
        acquisitionControlsDisable();
        spectrumDateTime = DateTime.Now;
        dateTimePickerSpectrumDateTime.Value = DateTime.Now;

        remoteCountAndSave = true;
        counting = true;

        header = new List<string>(HeaderInfo);

        mca.mcaStartAcquire();
    }
}

public void StopAcquisition()
{
    if (counting)
    {
        remoteCountAndSave = false;
        mca.mcaStopAcquire();
    }
}

```

## 11.2 Datamove

```

clear all;
format shortG
x=0;
y=0;
a=0;
Elementzeile=0;
ZeileElement=0;

files = dir('*.ASR');
firstASR = files(1).name;
firstElement = importdata(firstASR, '', 7);
Elementarray = size(firstElement.data);

PSE = importdata('xray.txt');

```

```

PSElaenge = size(PSE.textdata);

for k = 1:Elementarray(1)
    disp (PSE.textdata(firstElement.data(k)-1));
end

antwort = input('Welches Element soll dargestellt werden \n','s');

for a = 2:PSElaenge(1)
    if (strcmp(antwort,PSE.textdata(a)) == 1)
        Z = a+1;
    end
end

for t = 1:Elementarray(1)
    if (firstElement.data(t) == Z)
        ZeileElement = t;
    end
end

stnumberx(numel(files))= 0;
stnumbery(numel(files))= 0;

for i = 1:numel(files)
    str = files(i).name;
    x = str(1:4);
    stnumberx(i) = str2double(x);

    y = str(5:8);
    stnumbery(i) = str2double(y);
end
xachse = (max(stnumberx)+1);
yachse = (max(stnumbery)+1);

element(yachse,xachse)= 0;

for o = 1:numel(files)
    read = files(o).name;
    text = importdata(read,' ',7);
    wert = (text.data(ZeileElement,4));
    if ((text.data(ZeileElement,4)< 0))
        wert = 0;
    end
    wert = wert -(text.data(ZeileElement,5));
    if ((wert< 0))
        wert = 0;
    end
    wert = wert/PSE.data(Z-2,9);
    element (stnumbery(o)+1,stnumberx(o)+1)= wert;
end

disp (element);
contourf(element)

spefiles = dir('*.*spe');
xspe = num2str(input('X: \n','s'));
yspe = num2str(input('Y: \n','s'));
firstspe = [xspe yspe '.SPE'];
firstSpektrum = importdata(firstspe,' ',7);
spekanaele = size(firstSpektrum.data);
spex = 1:spekanaele(1);
spey = log10(firstSpektrum.data(1:spekanaele(1)));
area(spex,spey);

```

## 11.3 Koordinatentisch

18507 - Kreuztisch

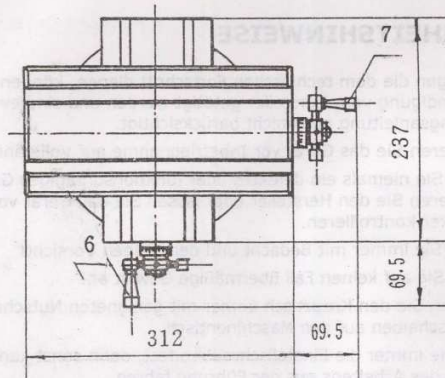


Abbildung 2

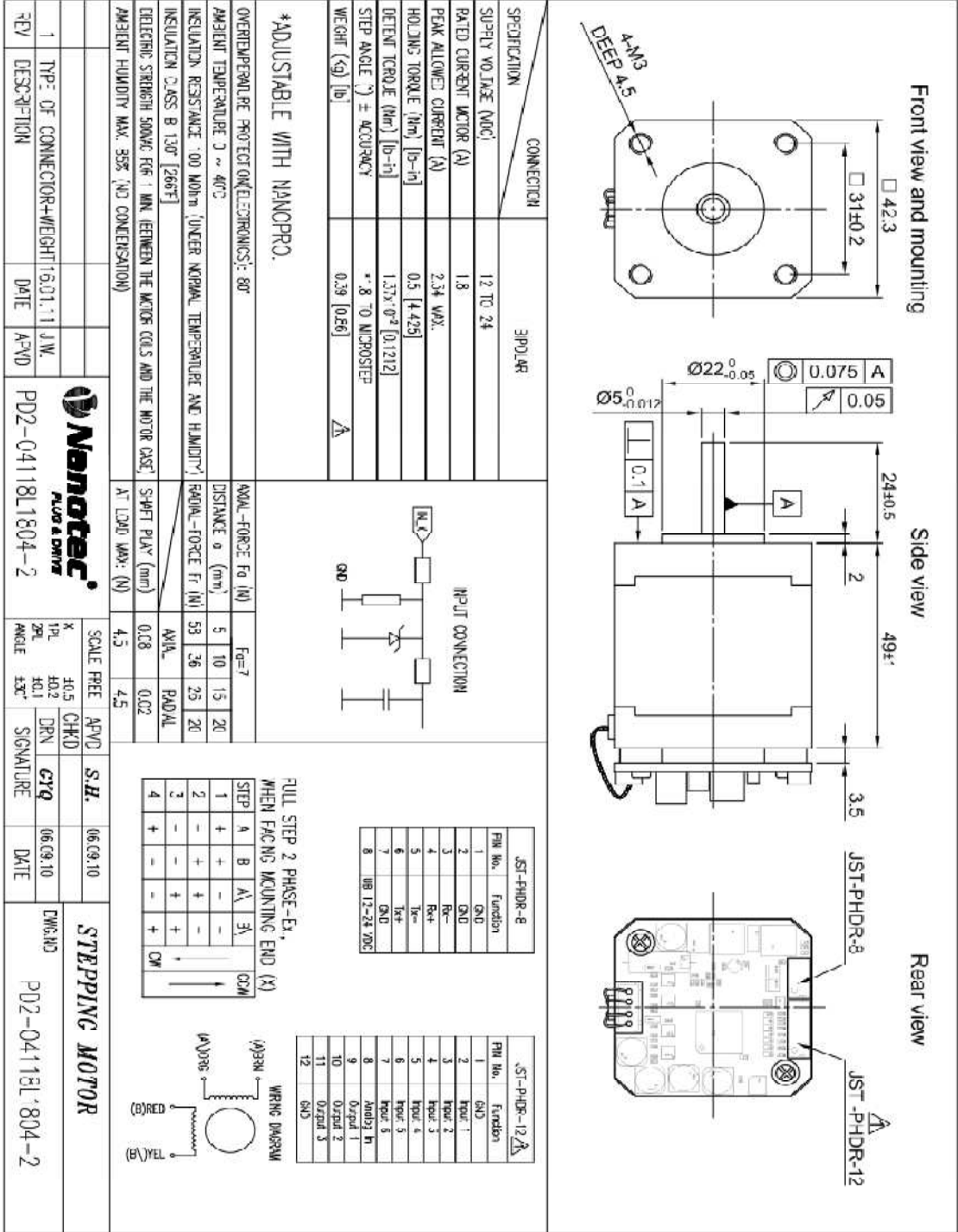
### 3 WARTUNG

- Reinigen Sie den Kreuztisch nach jedem Gebrauch. Späne können sich in den Führungsnuten festsetzen und so die Funktion des Kreuztisches beeinträchtigen.
- Fetten und schmieren Sie die regelmäßig beweglichen Teile und die Oberfläche des Kreuztisches.

### TECHNISCHE DATEN

Bezeichnung	Daten
Rechtwinkligkeit horizontal/vertikal	100: 0,0254 mm
Parallelität des Arbeitstisches	100: 0,0254 mm
Maximaler horizontaler Verfahrweg des Tisches	110 mm
Maximaler vertikaler Verfahrweg des Tisches	165 mm
Abweichung	0,02 mm
Nettogewicht	16 kg

11.4 Schrittmotor



## 11.5 Schaltnetzteil:

### 120W Single Output Industrial DIN RAIL Power Supply



#### ■ Features :

- AC input range selectable by switch
- Protections: Short circuit/Over load/Over voltage/Over temperature
- Cooling by free air convection
- Can be installed on DIN rail TS-35/7.5 or 15
- UL 508(industrial control equipment)approved
- LED indicator for power on
- 100% full load burn-in test
- Fixed switching frequency at 55KHz
- 3 years warranty



#### SPECIFICATION

MODEL		NTS-24V-5A	
OUTPUT	DC VOLTAGE	24V	
	RATED CURRENT	5A	
	CURRENT RANGE	0 ~ 5A	
	RATED POWER	120W	
	RIPPLE & NOISE (max.) Note.2	80mVp-p	
	VOLTAGE ADJ. RANGE	24 ~ 28V	
	VOLTAGE TOLERANCE Note.3	±1.0%	
	LINE REGULATION	±0.5%	
	LOAD REGULATION	±1.0%	
INPUT	SETUP, RISE TIME	500ms, 70ms/230VAC 500ms, 70ms/115VAC at full load	
	HOLD TIME (Typ.)	36ms/230VAC 32ms/115VAC at full load	
	VOLTAGE RANGE	88 ~ 132VAC/176 ~ 264VAC by switch 248 ~ 370VDC	
	FREQUENCY RANGE	47 ~ 63Hz	
	EFFICIENCY (Typ.)	80%	84% 85%
	AC CURRENT (Typ.)	2.6A/115VAC 1.6A/230VAC	
	INRUSH CURRENT (Typ.)	COLD START 20A/115VAC 40A/230VAC	
	LEAKAGE CURRENT	<3.5mA / 240VAC	
PROTECTION	OVER LOAD	105 ~ 150% rated output power Protection type : Constant current limiting, recovers automatically after fault condition is removed	
	OVER VOLTAGE	15 ~ 16.5V 29 ~ 33V 58 ~ 65V	
	OVER TEMPERATURE	85℃±5℃ (TSW1) 90℃±5℃ (TSW1) 90℃±5℃ (TSW1)	
		Protection type : Shut down o/p voltage, recovers automatically after temperature goes down	
ENVIRONMENT	WORKING TEMP.	-10 ~ +60℃ (Refer to output load derating curve)	
	WORKING HUMIDITY	20 ~ 90% RH non-condensing	
	STORAGE TEMP., HUMIDITY	-20 ~ +85℃, 10 ~ 95% RH	
	TEMP. COEFFICIENT	±0.03%/℃ (0 ~ 50℃)	
SAFETY & EMC (Note 4)	VIBRATION	10 ~ 500Hz, 2G 10min./1cycle, 60min. each along X, Y, Z axes	
	SAFETY STANDARDS	UL508, UL60950-1, TUV EN60950-1 Approved	
	WITHSTAND VOLTAGE	I/P-O/P:3KVAC I/P-FG:1.5KVAC O/P-FG:0.5KVAC	
	ISOLATION RESISTANCE	I/P-O/P, I/P-FG, O/P-FG:100M Ohms/500VDC	
OTHERS	EMI CONDUCTION & RADIATION	Compliance to EN55011, EN55022 (CISPR22) Class B	
	HARMONIC CURRENT	Compliance to EN61000-3-2, -3	
	EMS IMMUNITY	Compliance to EN61000-4-2, 3, 4, 5, 6, 8, 11, EN50204, EN55024, EN61000-6-2 (EN50082-2) Heavy industry level, criteria A	
	MTBF	136.8Khrs min. MIL-HDBK-217F (25℃)	
NOTE	DIMENSION	65.5*125.2*100mm (W*H*D)	
	PACKING	0.79Kg; 20pcs/16.5Kg/1.1CUFT	

1. All parameters NOT specially mentioned are measured at 230VAC input, rated load and 25℃ of ambient temperature.  
2. Ripple & noise are measured at 20MHz of bandwidth by using a 12" twisted pair-wire terminated with a 0.1uF & 47uF parallel capacitor.  
3. Tolerance : includes set up tolerance, line regulation and load regulation.  
4. The power supply is considered a component which will be installed into a final equipment. The final equipment must be re-confirmed that it still meets EMC directives.