

**TECHNISCHE  
UNIVERSITÄT  
DRESDEN**

---

# **Entwicklung einer FPGA-Firmware zur Messung der Myon-Lebensdauer innerhalb des Physikpraktikums**

Bachelor-Arbeit  
zur Erlangung des Hochschulgrades  
Bachelor of Science  
im Bachelor-Studiengang Physik

vorgelegt von

Arne Heimbold  
geboren am 01.09.1990 in Dresden

Institut für Kern- und Teilchenphysik  
Fachrichtung Physik  
Fakultät Mathematik und Naturwissenschaften  
Technische Universität Dresden

2014

Eingereicht am 17. November 2014

1. Gutachter: Prof. Dr. Arno Straessner

2. Gutachter: PD. Dr. Jürgen Henniger

## Kurzbeschreibung

Vorbereitend zur Digitalisierung der Ausleseelektronik von Szintillatoren am Institut für Kern- und Teilchenphysik der TU Dresden wird die Zeitmessung für den Versuch zur Lebensdauerbestimmung von Myonen mit einem Spartan-3E FPGA<sup>1</sup> realisiert. Die innerhalb dieser Arbeit in VHDL<sup>2</sup> geschriebene Firmware umfasst die Zeitmessung sowie die Datenübertragung, welche mit Hilfe eines UART<sup>3</sup> über die serielle Schnittstelle erfolgt. Die Funktionsweise des FPGA und der Software auf der Empfängerseite wird mit zwei Kontrollmessungen überprüft. Abschließend wird das Lebenszeitspektrum von Myonen aus der Höhenstrahlung aufgenommen und mit Ergebnissen alter Messungen verglichen.

---

<sup>1</sup>Field Programmable Gate Array

<sup>2</sup>Very High Speed Integrated Circuit Hardware Description Language

<sup>3</sup>Universal Asynchronous Receiver Transmitter

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>2</b>
<b>2</b>	<b>Physikalische Grundlagen</b>	<b>3</b>
2.1	Myonen aus der Höhenstrahlung . . . . .	3
2.2	Der $\mu^-$ -Einfang . . . . .	5
<b>3</b>	<b>Aufbau des Praktikumsversuches: Lebensdauer der Myonen</b>	<b>6</b>
3.1	Detektoren . . . . .	6
3.2	Versuchsaufbau und Messprinzip . . . . .	8
<b>4</b>	<b>Umsetzung mit dem FPGA</b>	<b>9</b>
4.1	Spartan-3E Starterboard . . . . .	9
4.2	Funktionsweise eines Zustandsautomaten . . . . .	9
4.3	Zeitmessung mit dem FPGA . . . . .	11
4.4	Datenübertragung über die serielle Schnittstelle . . . . .	13
4.4.1	Funktionsweise eines UART . . . . .	13
4.4.2	Umsetzung des Transmitters auf dem FPGA . . . . .	14
4.4.3	Übertragung eines Messwertes . . . . .	15
4.4.4	Verwendung und Funktionsweise eines FIFO-Buffers . . . . .	17
<b>5</b>	<b>Messungen mit dem FPGA</b>	<b>19</b>
5.1	Kontrollmessungen zur Verlässlichkeit der Datenübertragung und Zeitmessung	19
5.2	Messung der Lebensdauer von Myonen mit dem FPGA . . . . .	22
5.2.1	Auswertung durch Fit-Funktionen . . . . .	24
5.2.2	Maximum Likelihood Methode . . . . .	26
5.2.3	Zusammenfassung und Ausblick . . . . .	28
<b>A</b>	<b>Literaturverzeichnis</b>	<b>30</b>

# 1 Einleitung

Im Laborpraktikum zur Bestimmung der Lebensdauer von Myonen, am Institut für Kern- und Teilchenphysik (IKTP) der TU Dresden, werden Myonen aus der Höhenstrahlung mit Detektoren, welche aus einem Szintillator und einem Photomultiplier bestehen, nachgewiesen. Die von den Myonen oder ihren Zerfallsprodukten ausgelösten Signale bilden die Grundlage der Zeitmessung.

Bisher beruht die Auswertung dieser Signale auf analoger Trigger- und **Ausleseelektronik** **welche** im Rahmen einer Modernisierung durch einen FPGA ersetzt werden soll. Innerhalb dieser Bachelorarbeit wurde **dazu** die Firmware für den FPGA in VHDL geschrieben, welche die Zeitmessung zwischen Koinzidenzereignissen und die Datenübertragung über eine serielle Schnittstelle ermöglicht. Zeitmessung und Datenübertragung **wurde in Test** auf Fehler untersucht. Abschließend wurde ein Lebenszeitspektrum von Myonen im Laborversuch mit dem FPGA aufgenommen und mit alten Messungen verglichen.

Im zweiten Kapitel werden kurz die zum Verständnis des Praktikumsversuches benötigten physikalischen Grundlagen zusammengefasst. Anschließend wird in Kapitel drei der Versuchsaufbau und das Messprinzip vorgestellt. Die Umsetzung der Zeitmessung und der Datenübertragung mit dem FPGA wird im vierten Kapitel dargestellt. Dabei geht es nicht um die exakte Implementierung im Code **sondern** um die prinzipielle Funktionsweise einiger wichtiger Module innerhalb der Firmware. Im fünften Kapitel werden zwei Kontrollmessungen vorgestellt, mit denen die Funktionsweise des FPGA überprüft wurde. **Außerdem** wird ein mit dem FPGA aufgenommenes Lebenszeitspektrum dem einer älteren Messung mit der analogen Auslesetechnik gegenübergestellt.

## 2 Physikalische Grundlagen

### 2.1 Myonen aus der Höhenstrahlung

Kosmische Höhenstrahlung ist eine hochenergetische Teilchenstrahlung aus dem Weltall, die in Primär- und Sekundärstrahlung unterteilt wird. Bei der Primärstrahlung handelt es sich um die Teilchen, die direkt von einer Quelle (z.B. einem Stern) stammen. Sie besteht zu ca. 79 % aus freien Protonen (Wasserstoffkern), 15 % Alpha-Teilchen (Heliumkern) und etwa 6 % schwereren Atomkernen [4]. Als Sekundärstrahlung werden diejenigen Teilchen bezeichnet, die durch Wechselwirkungen der Primärstrahlung mit anderen Atomen (z.B. aus der Erdatmosphäre) entstehen. Auf der Erdoberfläche sind Myonen die häufigsten geladenen Sekundärteilchen der kosmischen Höhenstrahlung. Ihr Ursprung liegt in etwa 15 km Höhe, wo in Proton-Proton- oder Neutron-Proton-Reaktionen kurzlebige Pionen ( $\tau_\pi = (2,6033 \pm 0,0005) \cdot 10^{-8} \text{ s}$  [2]) und Kaonen ( $\tau_K = 1,2380 \pm 0,0021) \cdot 10^{-8} \text{ s}$  [3]) entstehen, die wiederum über die schwache Wechselwirkung in ein Myon und ein Neutrino zerfallen können. Aufgrund der jeweils sehr kurzen Lebensdauer finden diese Zerfälle noch in der Erdatmosphäre statt.



Die mittlere Lebensdauer von  $\mu^+$  und  $\mu^-$  ist fast identisch und beträgt  $\tau_\mu^\pm = (2,1969811 \pm 0,0000022) \cdot 10^{-6}$  s [1]. Ihre Geschwindigkeit ist so hoch, dass sie aufgrund der Zeitdilatation die Erdoberfläche erreichen, bevor sie in zwei Neutrinos und ein Elektron bzw. Positron zerfallen (Abbildung 2.1.1). Auf dem Weg zur Erdoberfläche verlieren die Myonen etwa 2 GeV durch Stoßionisation. Auf Meereshöhe besitzen sie im Durchschnitt noch eine Energie von 4 GeV [4].

$$\mu^+ \rightarrow e^+ + \nu_e + \bar{\nu}_\mu \quad (2.7)$$

$$\mu^- \rightarrow e^- + \bar{\nu}_e + \nu_\mu \quad (2.8)$$

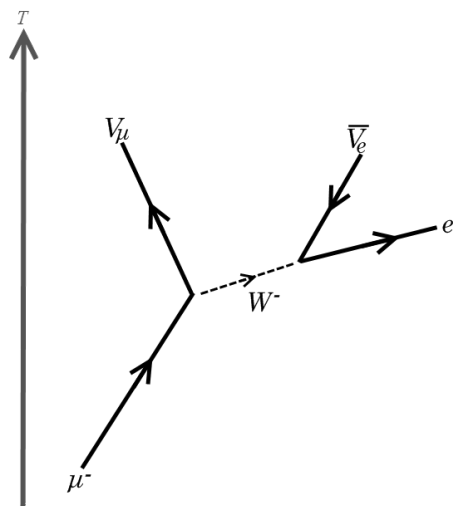


Abbildung 2.1.1: Feynman-Diagramm des  $\mu^-$ -Zerfalls [7]



## 2.2 Der $\mu^-$ -Einfang

Bei dem  $\mu^-$ -Einfang handelt es sich um einen Prozess, der in Konkurrenz zum freien Zerfall des  $\mu^-$  steht. Er kann auftreten, wenn ein negativ geladenes Myon in Materie zur Ruhe kommt. Dort wird es von dem elektromagnetischen Feld eines Atoms eingefangen und erreicht in weniger als  $10^{-12}$  s den Grundzustand des Atoms, wo es vom Kern absorbiert wird. Der Prozess wird durch folgende Reaktionsgleichung beschrieben:



Durch den Einfang wird die effektive Lebenszeit der negativ geladenen Myonen verkürzt. Sie berechnet sich wie folgt:

$$\frac{1}{\tau_{\text{eff}}} = \frac{1}{\tau_{\text{frei}}} + \frac{1}{\tau_{\text{capture}}} \quad (2.10)$$

Wie stark sich dieser Effekt auf die effektive Lebenszeit des  $\mu^-$  auswirkt, hängt vom Material ab (siehe Tabelle 2.2.1).

Kern bzw. Material	mittlere Lebensdauer $\tau_{\mu^-}$ [ $\mu\text{s}$ ]
AL	$0,865 \pm 0,004$
Cu	$0,1636 \pm 0,0008$
Pb	$0,0746 \pm 0,0006$
Szintillator	$\approx 2,1 \pm 0,1$

**Tabelle 2.2.1:** Effektive Lebenszeiten von  $\mu^-$  für ausgewählte Materialien [6]

# 3 Aufbau des Praktikumsversuches: Lebensdauer der Myonen

## 3.1 Detektoren

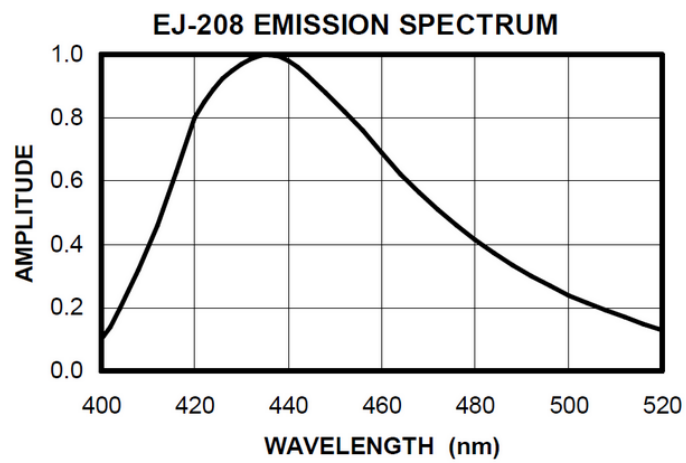
Die Detektoren im Praktikumsversuch bestehen jeweils aus einem Szintillator NE 110, einem Lichtleiter und einem Photomultiplier XP2020 mit der Basis S563/03. Die Szintillatoren und die Lichtleiter sind mit einer 25  $\mu\text{m}$  dicken Aluminiumfolie sowie einer 200  $\mu\text{m}$  dicken, schwarzen Plastikfolie umwickelt. Zusätzlich sind die Kanten mit mehreren Lagen eines schwarzen, 25 mm breiten Tapes abgeklebt. Diese Maßnahmen sollen die Bauteile vor Lichteinstrahlung schützen.

Szintillatoren sind Stoffe, deren Moleküle durch Stöße mit energiereichen Photonen, geladenen Teilchen oder Neutronen angeregt werden. Die Anregungsenergie wird in Form von Licht abgestrahlt. Mit dem Plastiksintillator NE 110 können sowohl die Myonen, als auch ihre Zerfallsprodukte aus dem freien Zerfall und dem  $\mu^-$ -Einfang detektiert werden.

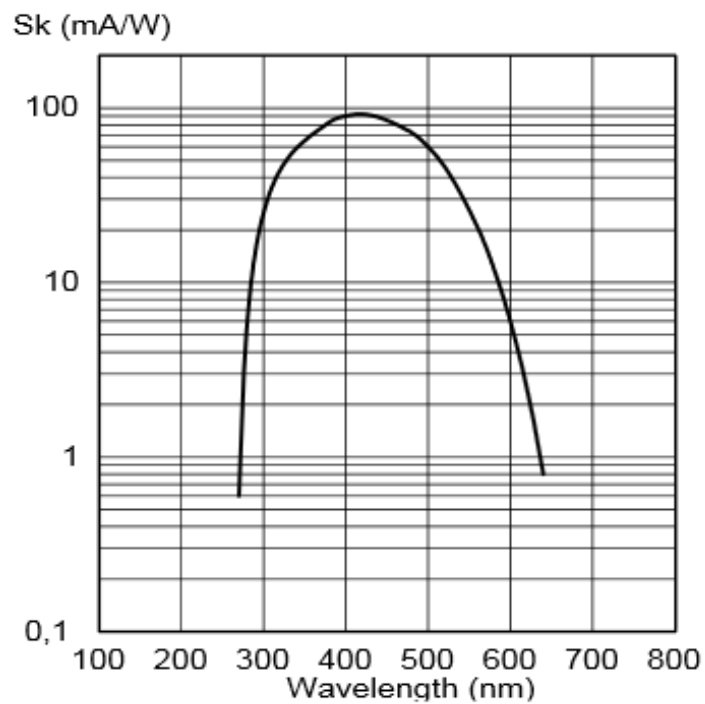
In Abbildung 3.1.1 ist das Spektrum der vom Szintillator emittierten Photonen zu sehen. Wie in der Abbildung **zu sehen ist**, befinden sich die meisten Photonen im sichtbaren Wellenlängenbereich zwischen 400 nm und 500 nm.

Über Lichtleiter werden die emittierten Photonen zu den Photomultipliern geleitet, wo sie aus den Photokathoden durch den äußeren photoelektrischen Effekt Elektronen lösen. Diese werden wiederum durch eine Hochspannung beschleunigt, bis sie auf eine Dynode treffen und dort eine Vielzahl von Sekundärelektronen herausschlagen. Die Sekundärelektronenemission wird bei den folgenden Dynoden mehrfach wiederholt, wodurch sich die Anzahl der Elektronen lawinenartig erhöht. Durch diese Verstärkung des Ausgangssignals können mit Photomultipliern selbst einzelne Photonen nachgewiesen werden.

Betrachtet man das Absorptionsspektrum des Photomultipliers (Abbildung 3.1.2), sieht man, dass dieser gut dazu geeignet ist, die vom Szintillator emittierten Photonen nachzuweisen.



**Abbildung 3.1.1:** Wellenlängenabhängigkeit des Emissionsfaktors von NE 110. EJ-208 ist die von Eljen Technology benutzte Bezeichnung für das Szintillatormaterial. [8]



**Abbildung 3.1.2:** Absorptionsspektrum eines Photomultipliers des Typs XP2020 [9]

## 3.2 Versuchsaufbau und Messprinzip

Der Versuchsaufbau besteht aus zwei 1 cm dicken Kupferplatten sowie zwei Detektoren über und einem Detektor unter den Kupferplatten (siehe Abbildung 3.2.1). An den Detektoren sind Diskriminatoren angeschlossen. Diese formen eingehende Signale in einheitliche Rechtecksignale, mit einer Breite von 40 ns um. Signale unterhalb einer einstellbaren Diskriminatorschwelle werden unterdrückt, um Rauscheffekte zu vermeiden.

Wird ein Myon aus der Höhenstrahlung im Kupfer-Target gestoppt, wurde es zuvor in Detektor 1 und 2 erfasst und hat Detektor 3 nicht erreicht. Das dadurch resultierende Signal wird kurz  $12\bar{3}$  genannt.

Beim Zerfall oder dem Einfang des Myons entstehen wiederum Teilchen, die in den Detektoren ein Signal auslösen (Positronen, Neutronen oder Elektronen). Wenn das Teilchen nach oben emittiert wird, erhält man das Signal  $2\bar{3}$ . Wird es nach unten emittiert, erhält man das Signal  $\bar{2}3$ . Somit kann  $12\bar{3}$  als Startsignal,  $2\bar{3}$  und  $\bar{2}3$  als Stoppsignal für eine Zeitmessung verwendet werden.

Bei schrägem Einfall von Teilchen können die Signale der Detektoren einen zeitlichen Abstand von etwa 3 ns besitzen, obwohl sie durch das selbe Teilchen ausgelöst wurden sind.

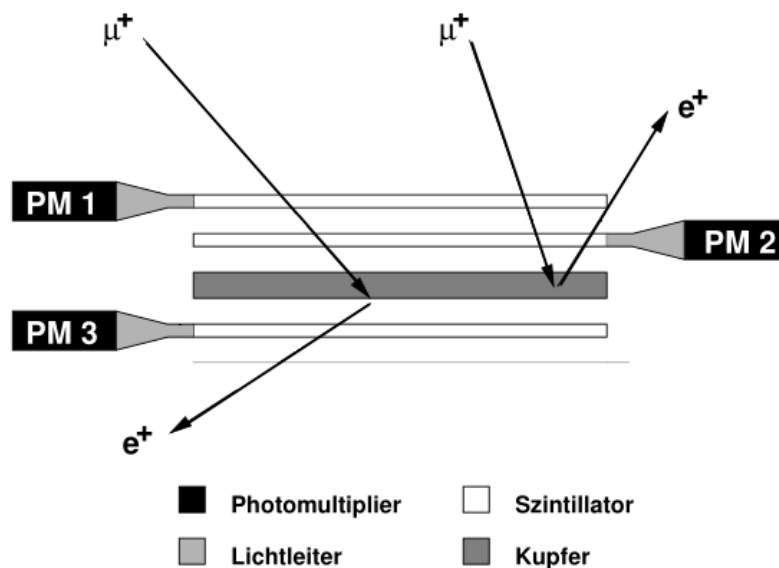


Abbildung 3.2.1: Skizze des Versuchsaufbaues [6]

# 4 Umsetzung mit dem FPGA

## 4.1 Spartan-3E Starterboard

Die Auswertung der Detektorsignale wird auf einem XC3S500E Spartan-3E FPGA<sup>1</sup> durchgeführt, welcher sich auf dem Spartan-3E Starterboard befindet. Die dazu benötigte Firmware wird in VHDL<sup>2</sup> geschrieben. Auf dem Board befindet sich ein 50 MHz Quarzoszillator, der als Quelle für ein schnelleres Taktsignal genutzt wird. Dazu wird mit dem Digital Clock Manager (DCM), welcher ein fester Bestandteil des FPGAs ist, aus dem 50 MHz Eingangssignal eine 200 MHz clock abgeleitet. Diese wird als Taktgeber für alle Module auf dem FPGA benutzt, das bedeutet ein Takt ist 5 ns lang.

Der FPGA ist über einen IC2-Spannungswandler an einen RS-232 Serial Port angeschlossen. Außerdem verfügt das Board über einen Serial Flash Speicher, von welchem die Firmware, beim Einschalten des Boards, auf den FPGA geschrieben werden kann.

## 4.2 Funktionsweise eines Zustandsautomaten

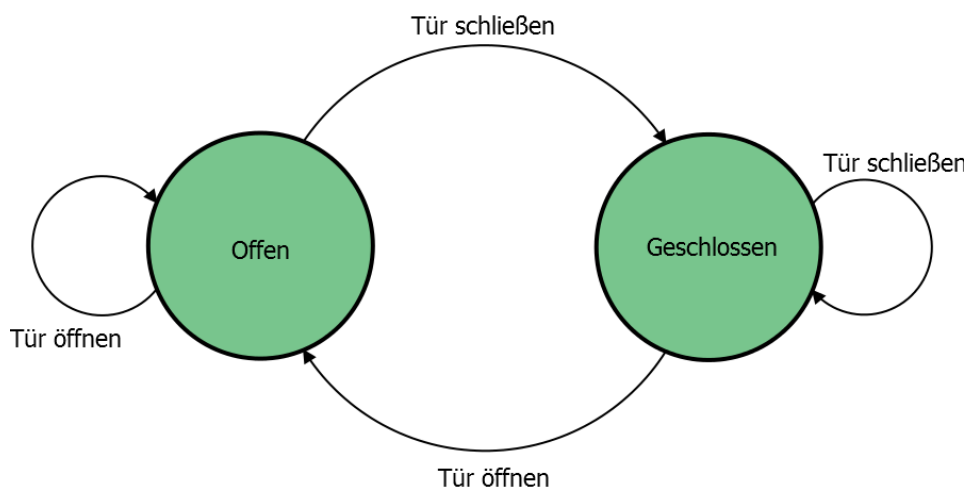
Zustandsautomaten (Finite State Machines, FSMs) bilden eine wesentliche Grundlage für die Beschreibung von digitalen Systemen. Bei ihnen handelt es sich um sequentielle Logikschaltungen die, von einem Taktsignal gesteuert, eine Reihe von Zuständen durchlaufen. Sie werden in der Hardwarebeschreibung vielseitig eingesetzt, um z.B. zyklische Prozesse zu beschreiben, andere Logikelemente zu steuern oder um die Synchronisation zwischen mehreren Modulen zu gewährleisten. **Nachfolgend** soll die grundlegende Funktionsweise eines Zustandsautomaten an Hand des Beispiels in Abbildung 4.2.1 aufgezeigt werden.

---

<sup>1</sup>Field-Programmable Gate Array

<sup>2</sup>Universal Asynchronous Receiver Transmitter

Der Zustandsautomat „Tür“ besitzt zwei Zustände: Offen und Geschlossen. Zu jedem Zeitpunkt befindet sich der Automat in genau einem dieser Zustände. In jedem Takt wird geprüft, ob sich der Zustand ändert. Dies geschieht mittels der Steuersignale Tür schließen und Tür öffnen. Wenn sich der Automat im Zustand Offen befindet und das Signal Tür schließen erhält, wechselt er zum nächsten Takt in den Zustand Geschlossen. Erhält er hingegen kein Signal oder Tür öffnen, ändert sich der Zustand nicht. Der Wechsel von Geschlossen zu Offen funktioniert dazu analog. Dieser Zustandsautomat kann zur Steuerung weiterer Signale genutzt werden, die wiederum die Funktionsweise anderer Automaten beeinflussen.



**Abbildung 4.2.1:** Skizze eines einfachen Zustandsautomaten mit den beiden Zuständen Offen und Geschlossen sowie den Signalen Tür schließen und Tür öffnen.

## 4.3 Zeitmessung mit dem FPGA

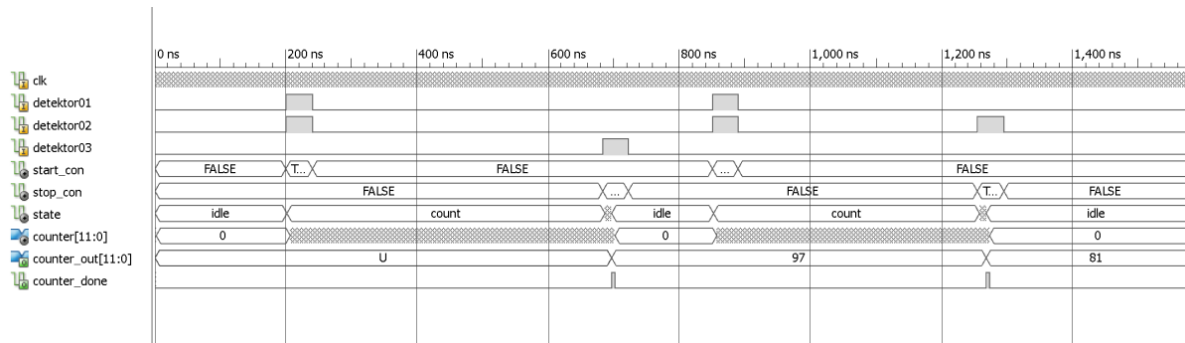
Die Zeitmessung wird im Modul Detektor in Form einer Zustandsmaschine mit den Zuständen `idle`, `count`, `test` und `stop` realisiert. Ein weiteres Modul überprüft, ob die eingehenden Signale der drei Detektoren die Start- oder Stoppbedingungen erfüllen. Dazu übergibt es die booleschen Variablen `start_con` bzw. `stop_con` an das Zeitmessungsmodul. Mit diesen beiden Signalen wird die Zustandsmaschine gesteuert.

Wenn die Startbedingung erfüllt ist und sich der Automat im Zustand `idle` befindet, wechselt er im nächsten Takt in `count` und die Zeitmessung beginnt. Dazu werden die Takte gezählt, in denen sich der Automat im Zustand `count` befindet. Die Zeitauflösung hängt direkt von der Frequenz der `clock` ab. Bei der hier verwendeten 200 MHz-`clock` erreicht man eine Zeitauflösung von 5 ns.

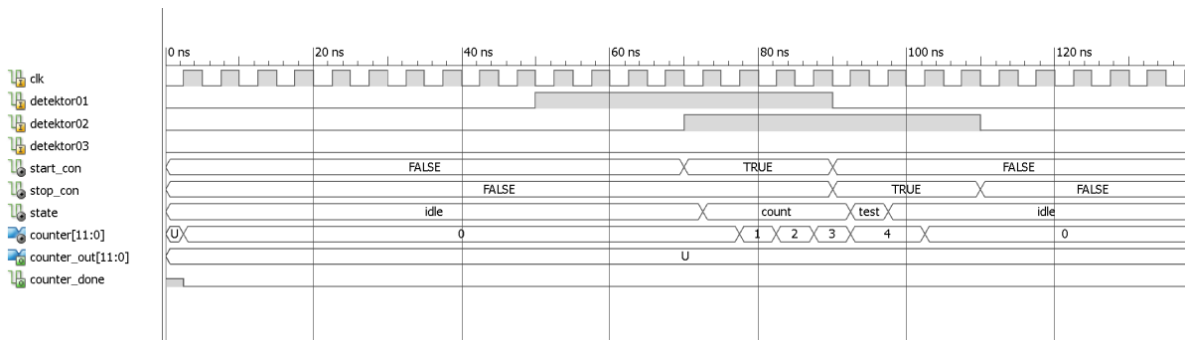
Es gibt zwei Möglichkeiten, den Zustand `count` zu verlassen. Die erste Möglichkeit besteht darin, dass innerhalb von  $10\ \mu\text{s}$  keine der Stoppbedingungen erfüllt wird. In diesem Fall wird die Messung abgebrochen und der Automat wechselt in den `idle` Zustand.

Wenn innerhalb des  $10\ \mu\text{s}$ -Zeitfensters die Stoppbedingung erfüllt wird, wechselt die Maschine in den `test` Zustand. Dort wird geprüft, ob die gemessene Zeit kleiner oder gleich 40 ns ist. Dies ist notwendig, da der Diskriminator die eingehenden Detektorsignale in Rechteckimpulse mit 40 ns Länge umwandelt. Das hat zur Folge, dass je ein Signal von Detektor 1 und 2 mit einer zeitlichen Differenz von weniger als 40 ns, sowohl die Start- als auch die Stoppbedingung erfüllt. Dadurch werden diese fehlerhaften Zeiten aussortiert (siehe Abbildung 4.3.2).

Wenn die gemessene Zeit länger als 40 ns ist, wechselt der Automat in den Zustand `stop`. Hier wird die gemessene Zeit an die Ausgabe `counter_out` übergeben und das interne Signal `counter_done` ausgelöst. Eine vollständige Zeitmessung ist in Abbildung 4.3.1 zu sehen.



**Abbildung 4.3.1:** Simulierte Zeitmessung von zwei Myonen-Zerfällen mit beiden möglichen Stoppbedingungen  $\bar{2}3$  und  $2\bar{3}$ . Das Signal counter\_out ist die **gemessene Zeit in** Vielfachen der clock-Periode (5 ns)



**Abbildung 4.3.2:** Simulierte Zeitmessung mit um 20 ns verschobenen Signalen von den Detektoren 1 und 2. Obwohl zuerst die Startbedingung und danach die Stoppbedingung erfüllt ist, werden keine Zeiten gemessen: counter\_out bleibt nicht definiert (U, undefiniert) und counter\_done '0'.



## 4.4 Datenübertragung über die serielle Schnittstelle

### 4.4.1 Funktionsweise eines UART

Ein Universal Asynchronous Receiver and Transmitter (UART) ist eine Schaltung, mit der von paralleler zu serieller Datenübertragung gewechselt wird. Das heißt ein Signal, das mehrere Datenpfade benutzt (parallel) wird zu einem Signal mit lediglich einem Datenpfad (seriell) konvertiert.

Ein solches Signal kann über eine RS-232 serielle Schnittstelle übertragen werden. Ein UART beinhaltet normalerweise einen Transmitter zum Senden und einen Receiver zum Empfangen von Daten. Da der FPGA innerhalb **unseres** Versuches nur Daten sendet, wurde lediglich der Transmitter umgesetzt. Grundsätzlich besteht der Transmitter aus einem Schieberegister, welches die parallelen Daten aufnimmt und in einer festgelegten Rate Bit für Bit seriell ausgibt. Die Ausgabegeschwindigkeit wird Baudrate genannt.

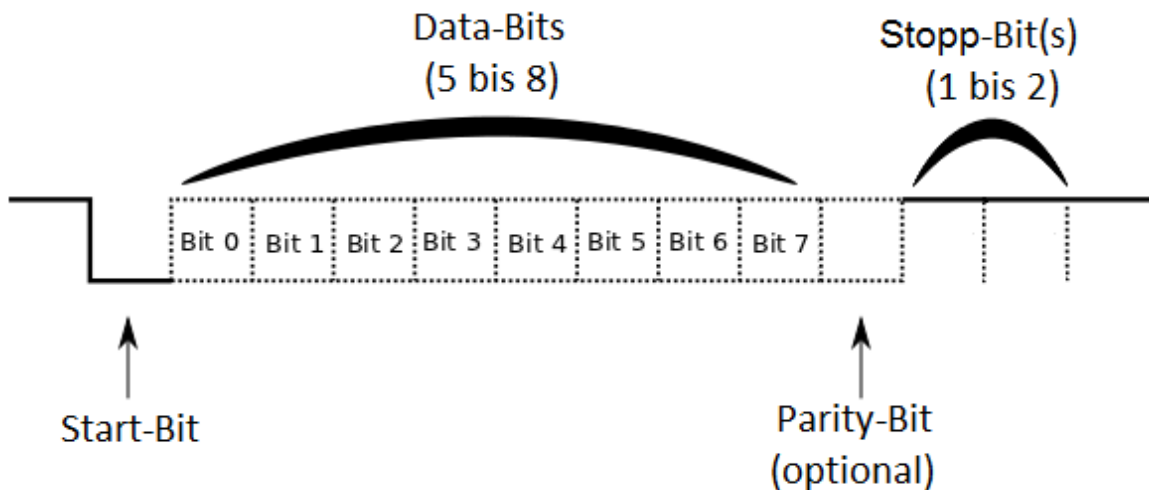
Ein Baud bedeutet, dass ein Bit pro Sekunde gesendet wird. Sender und Empfänger müssen mit der selben Baudrate arbeiten, damit die Daten korrekt übertragen werden können. Das serielle Signal ist '1', wenn keine Daten übertragen werden. Durch diese Konvention kann z.B. ein defektes Kabel bereits vor der eigentlichen Datenübertragung bemerkt werden. Ein eingehendes Datenpaket wird durch ein Start-Bit '0' angekündigt. Nach dem Start-Bit werden die Daten-Bits aus dem Schieberegister übertragen. Die Anzahl der Daten-Bits muss dabei in jedem Datenpaket identisch sein und zwischen fünf und acht liegen.

Nach den Datenbits kann ein Parity-Bit versendet werden, um die empfangenen Daten auf Übertragungsfehler zu prüfen. Die Verwendung des Parity-Bits ist optional und wurde für den Transmitter nicht umgesetzt. Den Abschluss des Datenpakets bilden ein bis zwei Stopp-Bits '1'.

Der Empfänger benötigt alle Informationen über die versendeten Datenpakete, um die Bits korrekt **zusammenzusetzen**;

- Mit welcher Baudrate wird gesendet? (Meist steht im Terminal-Programm eine begrenzte Anzahl von üblichen Baudraten zur Auswahl.)
- Wie viele Daten-Bits sind in einem Datenpaket enthalten? (zwischen 5 und 8)
- Wird ein Parity-Bit zur Erkennung von Übertragungsfehlern verwendet? (ja/nein)
- Wie viele Stoppbits bilden den Abschluss eines Datenpakets? (1, 1,5 oder 2)

Bei der Umsetzung des Transmitters auf dem FPGA müssen diese Limitierungen beachtet werden.



**Abbildung 4.4.1:** Aufbau eines Datenpakets mit Idle-Phase ('1'), Start-Bit '0', Data-Bits ('1' oder '0'), Parity-Bit ('1' oder '0') und Stopp-Bit(s) ('1').

## 4.4.2 Umsetzung des Transmitters auf dem FPGA

Der Transmitter ist im Modul „UART“ realisiert. In den `generics` werden die wesentlichen Parameter des in 4.4.1 beschriebenen Datenpakets an das Modul übergeben. So kann der Nutzer zwischen einem und zwei Stoppbits wählen, die Anzahl der Data-Bits einstellen und mit dem Baud-Divisor (`Baud_Div`) die Baudrate festlegen. Der Baud-Divisor berechnet sich wie folgt: 
$$\text{Baud\_Div} = \frac{\text{clock\_frequenz}}{\text{Baudrate}} = \frac{200\text{MHz}}{\text{Baudrate}}$$

Die Baudrate wird im Baudgenerator direkt von der verwendeten 200 MHz-clock abgeleitet, d.h. die Signale sind synchron zueinander.

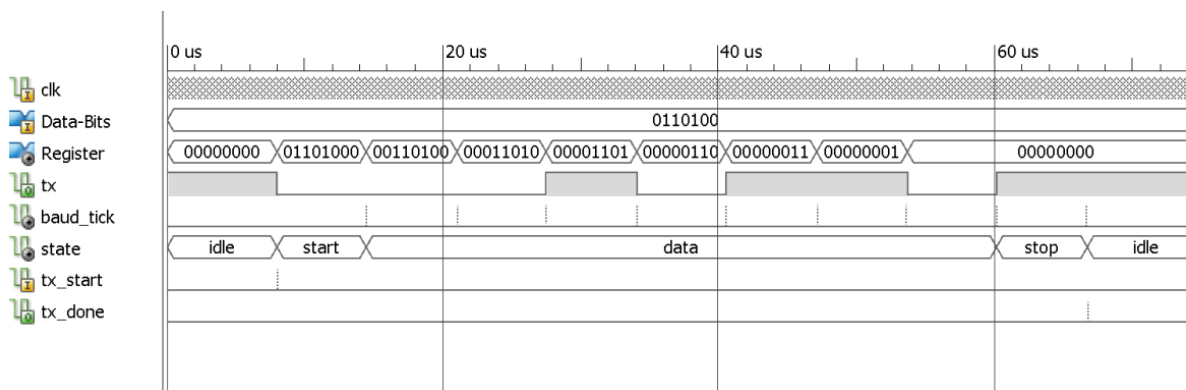
Bei dem Transmitter-Modul handelt es sich um einen Zustandsautomaten mit den Zuständen: `idle`, `start`, `data`, `stop` und `done`. Wenn keine Daten gesendet werden, befindet sich der Automat im `idle` Zustand. Wenn `tx_start` '1' ist, d.h., es gibt ein Byte, das versendet werden soll, wechselt der Automat in den Zustand `start` und der Baudgenerator startet.

Zusätzlich wird das zu versendende Byte in das Schieberegister geladen.

Solange sich der Automat in `start` befindet, ist das serielle Signal `tx` gleich '0'. Damit wird das Start-Bit versendet. Sobald das Signal `baud_tick` aus dem Baudgenerator '1' wird, wechselt der Automat in den Zustand `data`. Der Abstand zwischen den `baud_tick` Signalen entspricht genau der **Länge eines Bit** für die gewählte Baudrate. In `data` werden die Data-Bits in der Frequenz der Baudrate aus dem Register geschoben. Das serielle Signal `tx` entspricht dabei immer dem LSB (least significant Bit) im Schieberegister. Wenn alle Bits aus dem Register geschoben sind, wechselt der Automat in den Zustand `stop`. In diesem Zustand bleibt er, bis die in den `generics` festgelegte Anzahl von Stopp-Bits versendet wurde.

**Danach wechselt der** Automat für einen Takt in den Zustand `done`. Hier wird das interne Signal `tx_done` auf '1' gesetzt. Mit diesem Signal kann anderen Modulen mitgeteilt werden, das ein Datenpaket versendet wurde.

In Abbildung 4.4.2 ist ein Durchlauf des Zustandsautomaten zu sehen.



**Abbildung 4.4.2:** Simulation des Transmitters, in der das Byte "0110100" gesendet wird. Die Übertragung beginnt mit dem LSB (least significant Bit) und endet mit dem MSB (most significant bit). Der Sendevorgang beginnt mit dem Signal `tx_start`. Nachdem das komplette Datenpaket übertragen wurde, wird das Signal `tx_done` ausgegeben. In der Simulation wurde mit einer Geschwindigkeit von 38400 Baud gesendet.

### 4.4.3 Übertragung eines Messwertes

Im Experiment werden Zerfallszeiten in einem Intervall von 40 ns bis 10 μs gemessen. Bei einer Zeitauflösung von 5 ns werden mindestens 11 Bit benötigt, um alle möglichen Messzeiten darzustellen. Die serielle Schnittstelle ermöglicht nicht mehr als 8 Bit pro Datenpaket. Daher ist es notwendig, eine Messzeit auf zwei Byte aufzuteilen.

Da alle Datenpakete **dieselbe** Länge haben müssen, wird ein 12-Bit-Zähler für die Zeitmessung verwendet. Zur Übertragung wird dieser in zwei 6-Bit-Pakete zerteilt.

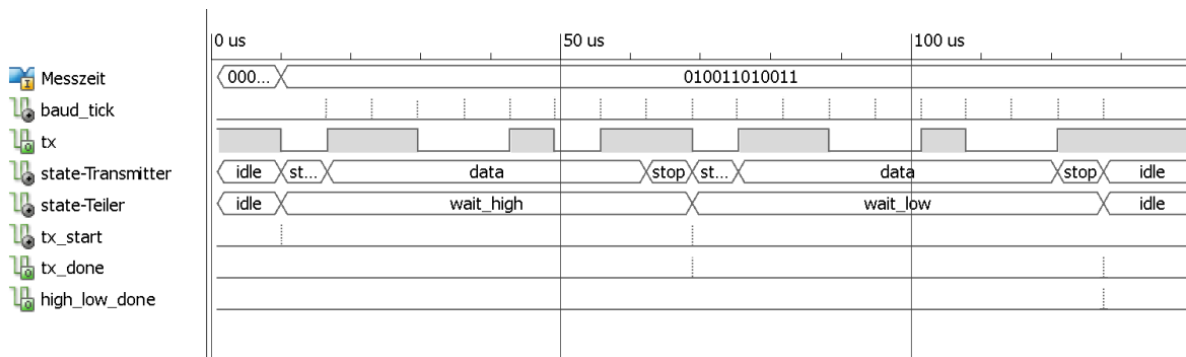
Um die Zeiten am Computer korrekt zusammensetzen zu können, ist es notwendig, die Pakete zu codieren. Dazu wird dem ersten Paket (Highbyte) eine '1' und dem zweiten Paket (Lowbyte) eine '0' als MSB (most significant bit) hinzugefügt. Eine Zeitmessung besteht somit aus zwei Paketen mit je sieben Datenbits.

Das Aufteilen der Messzeit in High- und Lowbyte ist in einem weiteren Zustandsautomaten umgesetzt, welcher den Transmitter über die internen Signale `tx_start` und `tx_done` steuert. In Abbildung 4.4.3 ist ein vollständiger Sendevorgang dargestellt.

Auf den LEDs des Spartan-3E Boards kann man wahlweise das zuletzt verschickte High- oder Lowbyte ausgeben lassen. Diese Funktion wurde zur Fehlersuche eingebaut und in der finalen Version des Codes beibehalten.

Messzeit	Highbyte	Lowbyte
"010011010011"	"1010011"	"0010011"

**Beispiel 4.4.1:** Demonstration der Zerlegung einer Messzeit in High- und Lowbyte.



**Abbildung 4.4.3:** Simulation der Übertragung eines Messwertes mit einer Geschwindigkeit von 38400 Baud.

Bei dem Messwert handelt es sich um **dieselbe** Bit-Folge wie im Beispiel 4.4.1. Zuerst wird das Highbyte an den Transmitter übergeben, der Sendevorgang wird durch das Signal `tx_start` gestartet. Wenn das Highbyte vollständig übertragen wurde, sendet der Transmitter das interne Signal `tx_done`. Im folgenden Schritt wird das Lowbyte übergeben und mit `tx_start` der zweite Teil der Übertragung gestartet.

Die Zeitdifferenz zwischen den beiden Steuersignalen beträgt lediglich einen Takt und ist daher bei der verwendeten Auflösung nicht zu erkennen. Ist das Lowbyte übertragen, signalisiert `high_low_done`, dass der komplette Messwert übermittelt wurde.

#### 4.4.4 Verwendung und Funktionsweise eines FIFO-Buffers

In Abbildung 4.4.3 wird deutlich, dass die Übertragung eines Messwertes eine nicht zu vernachlässigende Zeit in Anspruch nimmt. Daher muss sichergestellt werden, dass neue Daten, die während des Sendevorgangs des letzten Wertes aufgenommen werden, nicht verloren gehen. Dazu wird ein FIFO-Buffer verwendet.

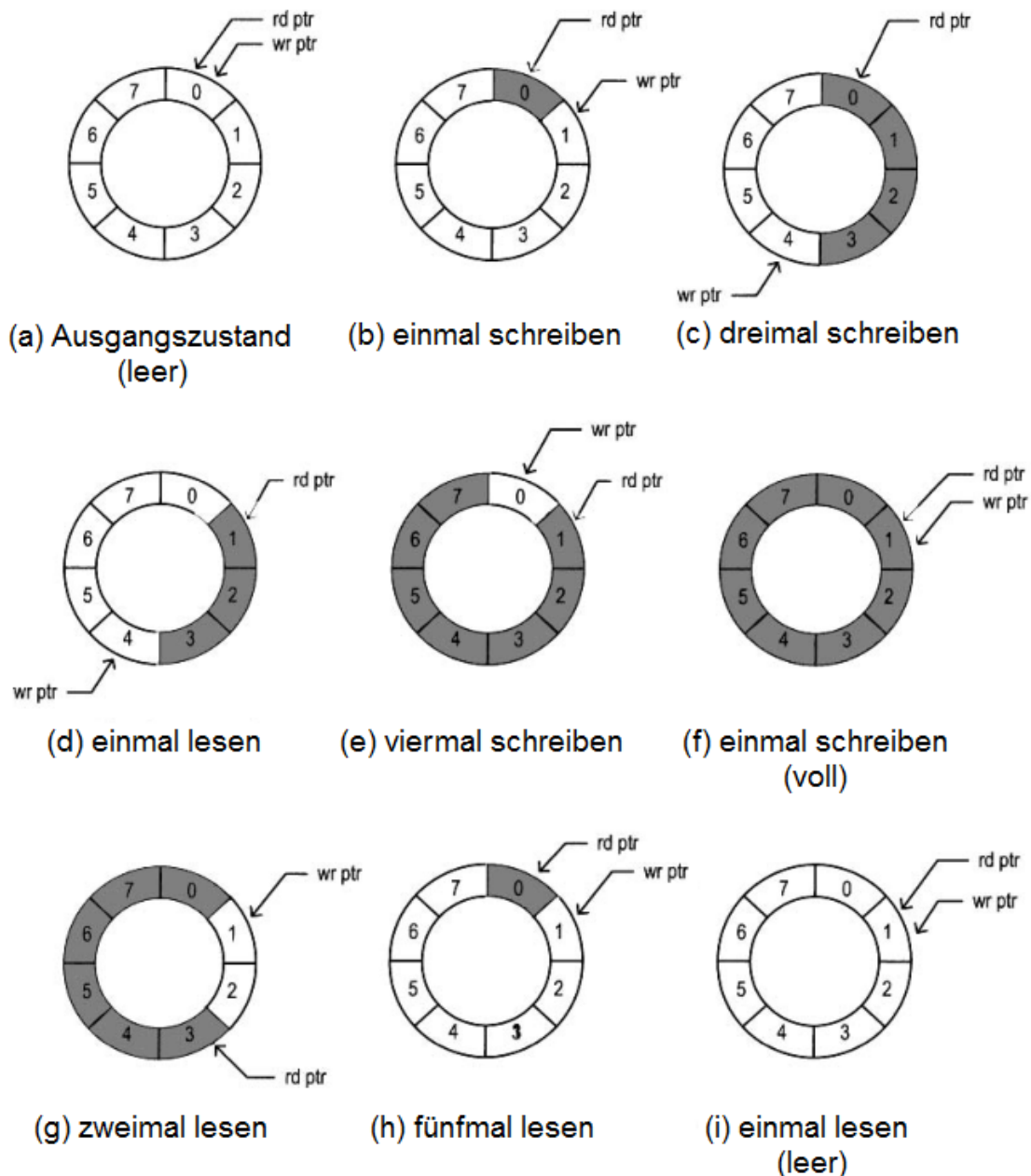
FIFO steht für „First In First Out“ und bezeichnet ein Speichersystem, in dem die Daten, die als Erstes in den Speicher geschrieben, auch zuerst ausgelesen werden.

Die Funktionsweise eines solchen Buffers ist am Beispiel einer einfachen Umsetzung in Abbildung 4.4.4 erklärt. In (a), (f) und (i) wird deutlich, dass  $wr\_ptr$  gleich  $rd\_ptr$  sowohl für einen vollen, als auch für einen leeren Speicher zutrifft. Daher muss unterschieden werden, welcher Zeiger den anderen einholt.

Wechselt der Lese-Zeiger auf die Position des Schreib-Zeigers, ist der FIFO leer. Holt der Schreib-Zeiger hingegen den Lese-Zeiger ein, ist der FIFO voll. Zur Umsetzung des FIFOs wurde der LogiCORE™IP FIFO (IP Core) Generator der Firma Xilinx benutzt. Ein IP Core ist ein vorgefertigter Funktionsblock eines Herstellers, der meist über Lizenzgebühren erworben werden kann.

Mit dem Generator kann ein FIFO mit den für die spezielle Anwendung benötigten Eigenschaften erstellt werden. Dieser FIFO IP Core wird als Blackbox in den restlichen Code eingebettet. Der im Code verwendete FIFO nutzt nur einen Bruchteil der im Generator angebotenen Funktionen.

Wenn der FIFO voll ist, leuchtet eine Warn-LED auf dem Spartan-3E Board und Messdaten gehen verloren. Die 16 Speicherplätze des FIFO und die voreingestellte Baudrate von 38400 Baud sind mehr als ausreichend, um die eingehenden Messwerte verlustfrei an den Computer zu übertragen.



**Abbildung 4.4.4:** FIFO-Buffer, aufgebaut auf einem 8 Bit Ringspeicher.

Die Abkürzung **wr ptr** steht für write pointer und bezeichnet einen Zeiger, der auf das Speicherelement zeigt, welches als nächstes beschrieben werden soll.

rd ptr bezeichnet analog dazu den read pointer, der auf das Element zeigt, welches als nächstes ausgelesen wird. Die Bildfolge stellt die Zeigerpositionen und die beschriebenen Speicherelemente im Verlauf mehrere Schreib- und Lesebefehle dar. Die Bildunterschriften geben jeweils an, durch welche Aktionen das Teilbild aus seinem Vorgänger entstanden ist. [5]

# 5 Messungen mit dem FPGA

## 5.1 Kontrollmessungen zur Verlässlichkeit der Datenübertragung und Zeitmessung

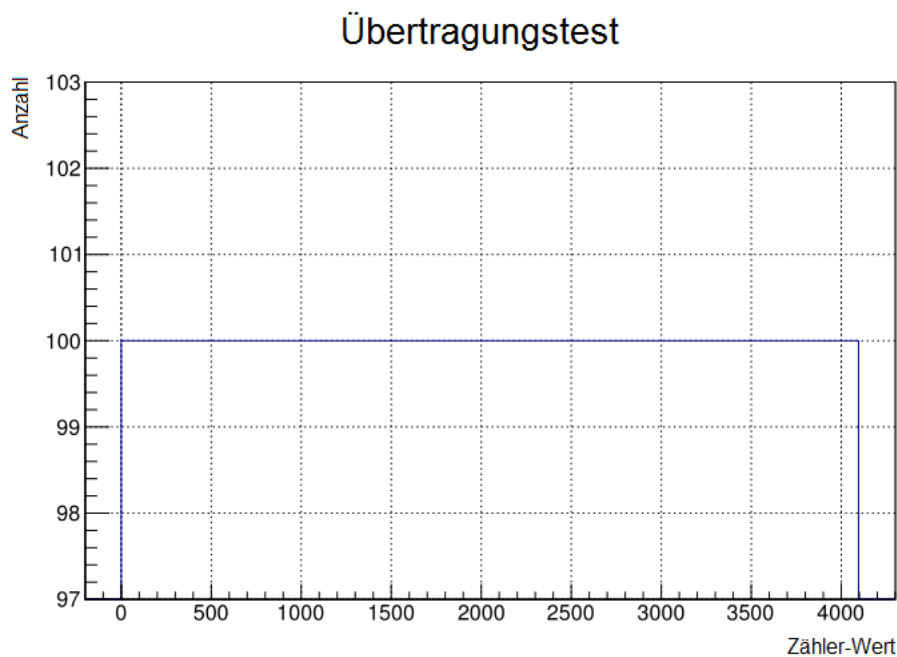
Im Folgenden werden zwei Messungen vorgestellt, mit denen die finale Version des Codes getestet wurde. Diese Tests waren nötig, da die Simulation nur für den Zeitbereich einiger Millisekunden geeignet ist. Fehler, die erst nach einer häufigen Wiederholung der Zustandszyklen auftreten, können daher nur mit einer echten Messung gefunden werden. Zusätzlich muss sichergestellt werden, dass die Empfängerseite ebenfalls funktioniert.

Für den ersten Test wurde das Detektor-Modul zur Zeitmessung durch einen 12-Bit-Zähler ersetzt, der in festen Zeitabständen zählt und jeden Wert an die FIFO übergibt. Dabei wurde der 12-Bit-Zähler genau einhundert mal durchgezählt.

In Abbildung 5.1.1 ist das Histogramm der vom Terminalprogramm empfangenen Werte zu sehen. Jeder Wert hat dabei einen eigenen Balken. Die Y-Achse wurde so skaliert, dass eine Abweichung von den 100 Messwerten pro Balken deutlich zu sehen wäre. Die Werte reichen von **null bis 4095**, was dem mit 12 Bit darstellbaren Zahlenbereich entspricht. Insgesamt wurden 409600 Werte übertragen. **Die Messung legt nahe**, dass die Daten die in den FIFO geschrieben werden **Fehlerfrei** über den UART an den Computer übertragen werden. Nachdem der erste Test bestätigt hat, dass die Datenübertragung funktioniert, wurde im zweiten Test die Zeitmessung untersucht. Hierfür wurde mit dem Funktionsgenerator Tektronix AFG3252 zwei Rechteck-Funktionen erzeugt, die zueinander um eine halbe Periode verschoben waren. Eines der Signale wurde als Detektor 1 und 2 verwendet, das andere als Detektor 3. Dadurch wird periodisch immer erst die Startbedingung  $12\bar{3}$  für eine halbe Periode erfüllt und danach die Stoppbedingung  $\bar{1}2\bar{3}$ . Somit kann man mit der Periode der Rechteck-Funktionen gezielt einstellen, welche Zeiten gemessen werden sollen (siehe Abbildung 5.1.2). Mit dieser Methode wurden für verschiedene Periodendauern Daten aufgenommen und in einem Histogramm dargestellt (Abbildung 5.1.3a).

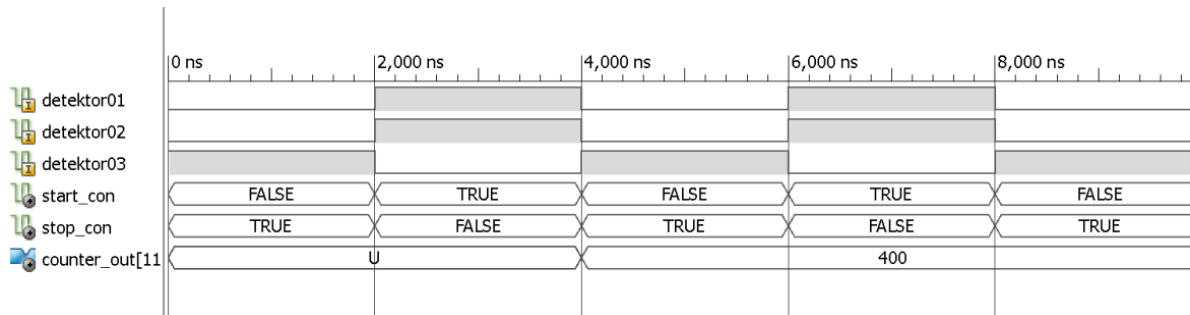
Für verschiedene Zeiten wurden unterschiedlich viele Daten aufgenommen. Grund dafür ist, dass kein extra Zähler in den Code implementiert wurde, der die Übertragung nach einer festen Anzahl von Messwerten abbricht. In den Detailansichten von Abbildung 5.1.3 wurde die Auflösung des Histogramms so angepasst, dass Messwerte zu einer ausgewählten Zeit genauer erkennbar sind. Die Balkenbreite entspricht der durch die 200-MHz-clock vorgegebenen Zeitauflösung von 5 ns. Die Balkenmitte wurde auf die möglichen Messwerte gesetzt.

In den Detailansichten sieht man, dass es für jede Zeit zwei Messwerte gibt. Dies entspricht der erwarteten Messgenauigkeit von einem Digit. Damit legt dieser zweite Test nahe, dass die Zeitmessung mit dem FPGA im Rahmen der technisch möglichen Messgenauigkeit korrekt funktioniert.

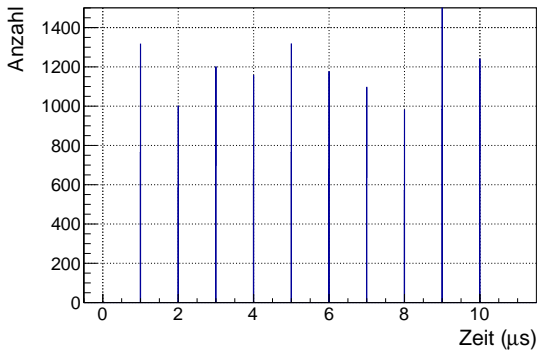


**Abbildung 5.1.1:** Kontrollmessung zur Datenübertragung

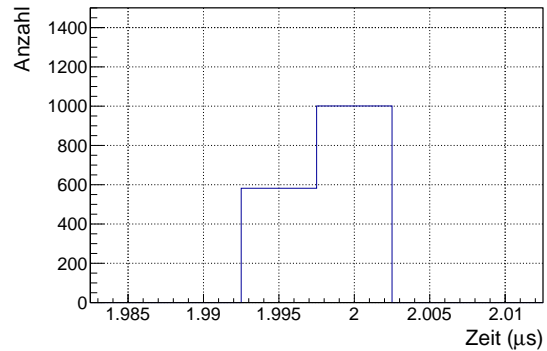




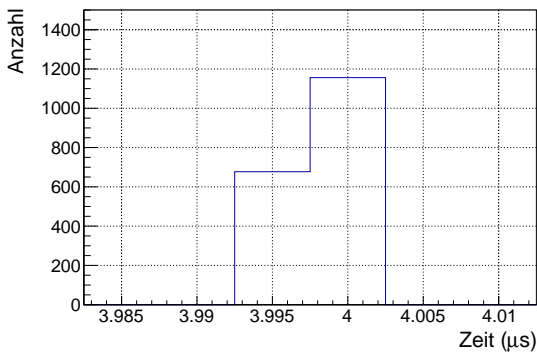
**Abbildung 5.1.2:** Simulation der als Detektorsignale verwendeten Rechteck-Funktionen. Die gemessene Zeit ( $400 * 5 \text{ ns} = 2 \mu\text{s}$ ) entspricht der halben Periode der Rechteck-Funktionen.



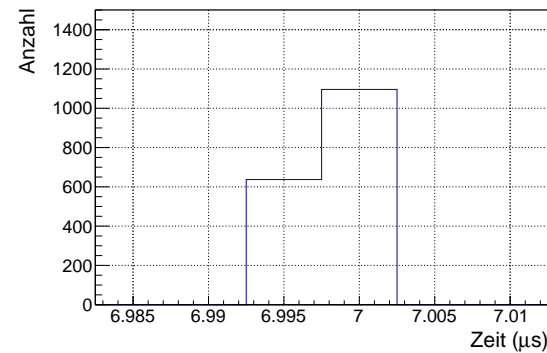
**(a)** Übersicht über alle gemessenen Zeiten



**(b)** Detailansicht für alle zur Zeit  $2 \mu\text{s}$  gehörigen Messwerte



**(c)** Detailansicht für alle zur Zeit  $4 \mu\text{s}$  gehörigen Messwerte



**(d)** Detailansicht für alle zur Zeit  $7 \mu\text{s}$  gehörigen Messwerte

**Abbildung 5.1.3:** Histogramm der Zeitkalibration (a) mit drei verschiedenen Detailansichten (b), (c) und (d)

## 5.2 Messung der Lebensdauer von Myonen mit dem FPGA

In einer fünftägigen Messung wurde mit dem FPGA das Lebenszeitspektrum von Myonen aufgenommen. Dies geschah parallel zu einer Messung mit der analogen Ausleseelektronik (TDC<sup>1</sup>), die im Rahmen des Fortgeschrittenenpraktikums durchgeführt wurde.

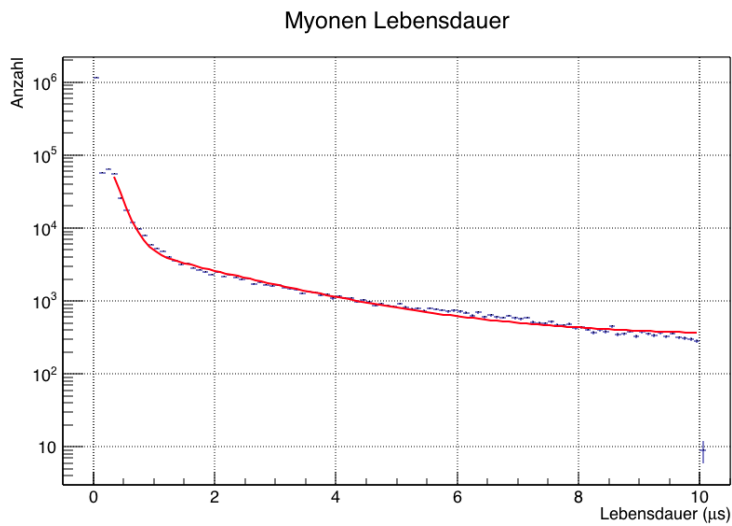
Die Betriebsspannung an den Photomultipliern betrug für Detektor 1 und 2 2350 V und für Detektor 3 2150 V. Das mit dem FPGA aufgenommene Spektrum ist in Abbildung 5.2.1 zu sehen. Im stark abfallenden ersten Teil des Histogramms (bis etwa  $1,5 \mu\text{s}$ ) dominiert der Anteil der negativ geladenen Myonen, die durch den  $\mu^-$ -Einfang eine stark verkürzte effektive Lebensdauer in Kupfer aufweisen. Der zweite Teil fällt wesentlich flacher ab als der erste, da der Großteil der negativ geladenen Myonen bereits zerfallen ist. Ebenso fällt auf, dass es im vordersten Teil des Spektrums ( $t < 0,5 \mu\text{s}$ ) einen deutlichen Einbruch bei der Anzahl der erfassten Messwerte gibt. Im Normalfall sollte die Zeitmessung nach  $10 \mu\text{s}$  abbrechen, sodass die Werte mit  $t > 10 \mu\text{s}$  theoretisch gar nicht messbar wären. Der drastische Unterschied in den Ereigniszahlen legt nahe, dass es sich hierbei um einen äußerst selten auftretenden Fehler handelt.

Da die Messdaten aus dem Praktikumsversuch wegen eines technischen Problems nicht ordnungsgemäß auswertbar sind, wird der Messung mit dem FPGA stattdessen eine ältere Messung gegenübergestellt (Abbildung 5.2.2).

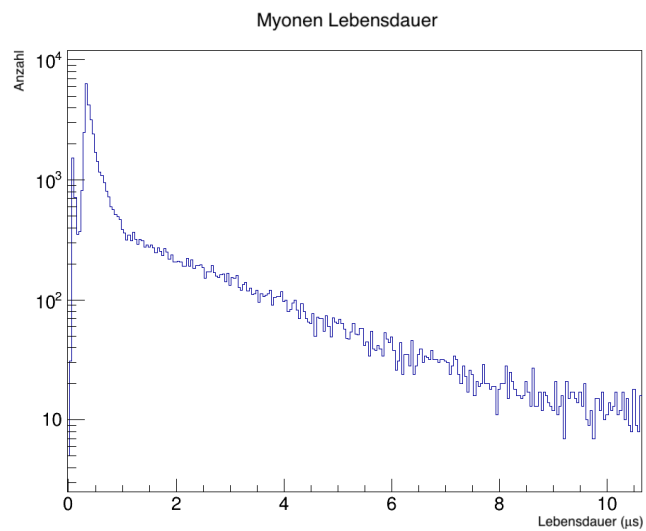
Hier gibt es ebenfalls einen stark abfallenden ersten Teil des Histogramms und einen zweiten, wesentlich flacheren. Der Einbruch bei der Anzahl der Messwerte für sehr kleine Zeiten ist ebenfalls deutlich zu sehen. Daher kann davon ausgegangen werden, dass dieser Einbruch durch die Detektoren oder die Diskriminatoren verursacht wird. Für die Bestimmung der freien Lebenszeit von positiv geladenen Myonen spielt dieser Einbruch keine Rolle, weshalb er die Durchführung des Experiments nicht behindert. Im Folgenden soll das Lebenszeitspektrum mit verschiedenen Methoden ausgewertet werden.

---

<sup>1</sup>Time to Digital Converter



**Abbildung 5.2.1:** Lebenszeitspektrum, das mit dem FPGA aufgenommen wurde.



**Abbildung 5.2.2:** Lebenszeitspektrum, das mit dem TDC aufgenommen wurde.

## 5.2.1 Auswertung durch Fit-Funktionen

Eine einfache Methode, ein exponentielles Zerfallsgesetz auszuwerten, besteht in einem Fit mit einer Exponentialfunktion. Dies sollte zumindest für den Teil des Zerfallsspektrums ein gutes Ergebnis liefern, in dem der Einfluss der negativen Myonen zu vernachlässigen ist. Mit **Root<sup>2</sup>** wurde folgende Fit-Funktion berechnet:

$$f(t) = A + B \cdot e^{-\frac{t}{\tau_0}} \quad (5.1)$$

A: Offset

B: Skalierungsfaktor

$\tau_0$ : mittlere Lebensdauer

Der erhaltene Wert für die mittlere Lebensdauer hängt vom Start- und Endwert des Fit-Intervalls ab. Daher wurde der Fit für eine Vielzahl von verschiedenen Fit-Intervallen durchgeführt (siehe Abbildung 5.2.3). In beiden Fällen gibt es eine starke Abhängigkeit vom gewählten Startwert. Dies ist logisch, da die gewählte einfache Exponentialfunktion nicht die Überlagerung von zwei Zerfallsprozessen mit unterschiedlichen Lebensdauern beschreiben kann. Der Startwert des Fit-Intervalls, für den ein  $\tau$  im Bereich  $1,2 - 3,2 \mu\text{s}$  gefunden wird, ist kleiner je größer der Endwert **ist**. Fit-Bereiche mit hohen Start- und Endwerten führen bei den Messdaten vom FPGA zu sehr großen Werten für  $\tau$ . Dieser Effekt lässt sich bei den alten Messwerten nicht beobachten. Ebenso der negative Einfluss auf die  $\tau$ -Werte durch kleine Endwerte ist bei den Messdaten vom FPGA stärker ausgeprägt.

Um das gesamte Lebenszeitspektrum mit einer Funktion zu beschreiben, muss diese sowohl den freien Zerfall der positiv geladenen Myonen, als auch die effektive Lebensdauer der negativ geladenen Myonen beinhalten. Dies soll mit folgendem Fit erreicht werden:

$$f(t) = A + B \cdot e^{-\frac{t}{\tau_0}} \cdot e^{-\frac{t}{\tau_c}} + C \cdot e^{-\frac{t}{\tau_0}} \quad (5.2)$$

A: Offset

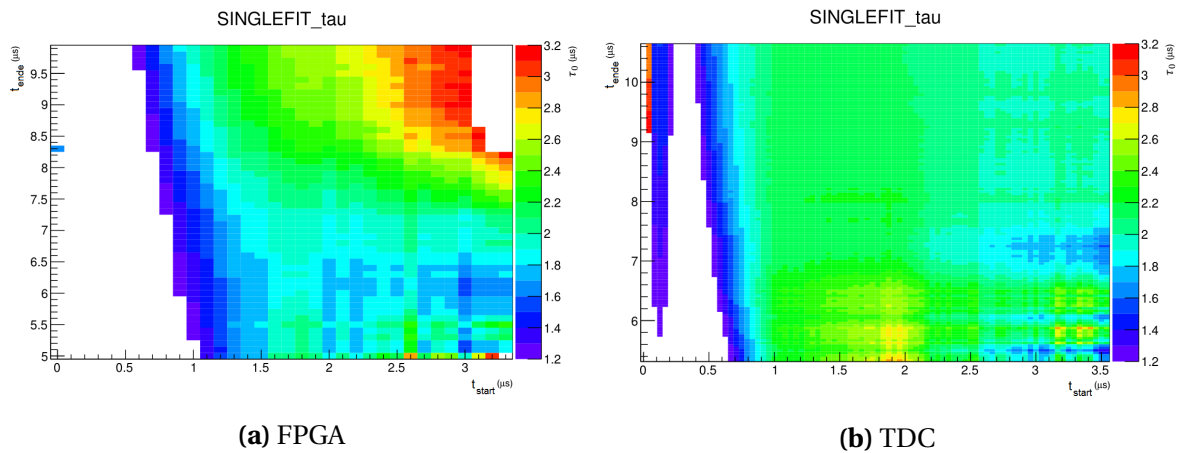
B,C: Skalierungsfaktoren

$\tau_0$ : mittlere Lebensdauer beim freien Zerfall

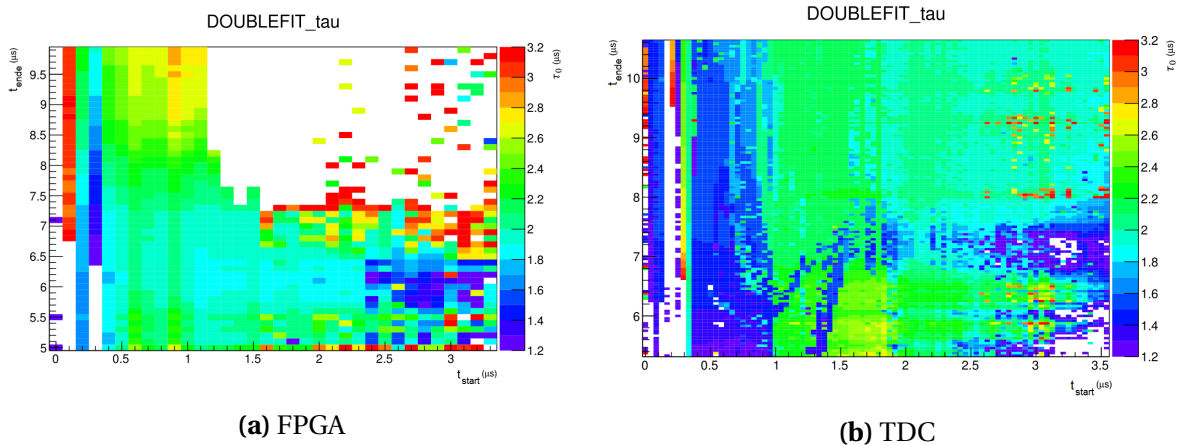
$\tau_c$ : mittlere Lebensdauer beim  $\mu^-$ -Einfang

---

<sup>2</sup>Software zur Datenanalyse



**Abbildung 5.2.3:** Auswertung des Lebenszeitspektrums durch einen Fit mit einer einfachen Exponentialfunktion. Auf der X-Achse ist der Startwert des Fit-Intervalls und auf der Y-Achse der Endwert aufgetragen. Auf der Z-Achse ist der für das jeweilige Fit-Intervall erhaltene Wert für die mittlere Lebenszeit durch eine Farbskala dargestellt.



**Abbildung 5.2.4:** Auswertung des Lebenszeitspektrums durch einen Fit mit einer Summe von zwei Exponentialfunktionen. Auf der X-Achse ist der Startwert des Fit-Intervalls und auf der Y-Achse der Endwert aufgetragen. Auf der Z-Achse ist der für das jeweilige Fit-Intervall erhaltene Wert für die mittlere Lebenszeit durch eine Farbskala dargestellt

Die mit dieser Fit-Funktion ermittelten  $\tau_0$ -Werte sind in der Abbildung 5.2.4 dargestellt. Der Bereich niedriger Startwerte, in denen nur sehr schlechte Werte für  $\tau_0$  ermittelt werden, ist im Vergleich zur Berechnung mit der einfachen Exponentialfunktion deutlich kleiner. Das für Startwerte kleiner  $0,5 \mu\text{s}$  keine verwertbaren Ergebnisse erhalten werden, liegt an dem dortigen Einbruch der gezählten Ereignisse im Lebenszeitspektrum.

Im Bereich von  $t_{start} > 1,2 \mu s$  und  $t_{ende} > 7 \mu s$  liefern die Messdaten vom FPGA sehr schlechte Ergebnisse. Bei den alten Daten vom TDC ist dies nicht der Fall. Das beschriebene Verhalten war bei der Auswertung mit der einfachen Exponentialfunktion bereits zu sehen und tritt hier bei der doppelten Exponentialfunktion sogar noch verstärkt auf.

## 5.2.2 Maximum Likelihood Methode

Bei einer bekannten Wahrscheinlichkeitsverteilung, welche von einem Parameter  $a$  und Messwerten  $\{x_1, \dots, x_N\}$  abhängt, kann mit Hilfe der Maximum-Likelihood-Methode derjenige Wert von  $a$  abgeschätzt werden, der für die Messdaten am wahrscheinlichsten ist. Dazu wird die Likelihood-Funktion  $L(x_1, \dots, x_N, a)$  gebildet, welche das Produkt der Einzelwahrscheinlichkeiten der Messwerte in Abhängigkeit von  $a$  ist:

$$L(x_1, \dots, x_N, a) = \prod_{i=1}^N P(x_1, \dots, x_N, a) \quad (5.3)$$

Die Likelihood-Funktion gibt dabei die Wahrscheinlichkeit an, für einen bestimmten Wert  $a$  exakt diese Messwerte  $\{x_1, \dots, x_N\}$  zu erhalten. Demnach ist genau das  $\hat{a}$  die beste Schätzung, in dem  $L(x_1, \dots, x_N, a)$  maximal ist.

In der Praxis wird häufig der Logarithmus von  $L$  maximiert, da dies bei komplizierten Exponenten in den Wahrscheinlichkeitsverteilungen unkomplizierter ist:

$$\left. \frac{d \ln L}{d a} \right|_{a=\hat{a}} = 0 \quad (5.4)$$

### Maximum-Likelihood für das **Exponentielle Zerfallsgesetz**:

Der freie Zerfall der positiv geladenen Myonen gehorcht dem exponentiellen Zerfallsgesetz. Damit ergibt sich folgende Wahrscheinlichkeitsdichte, in Abhängigkeit von der mittleren Lebensdauer  $\tau$  die Zerfallszeit  $t_i$  zu messen:

$$P(t_i, \tau) = \frac{1}{\tau} e^{-\frac{t_i}{\tau}} \quad (5.5)$$

Da nur Messwerte in einem Intervall  $[t_a, t_b]$  betrachtet werden, muss die Wahrscheinlichkeitsdichte auf dieses Intervall normiert werden:

$$P(t, \tau) = \frac{1}{\tau} \cdot \frac{e^{-\frac{t}{\tau}}}{e^{-\frac{t_A}{\tau}} - e^{-\frac{t_B}{\tau}}} \quad (5.6)$$

Anschließend muss der Logarithmus der Likelihood-Funktion maximiert werden. Dazu wird zuerst die Ableitung gebildet und danach numerisch nach Nullstellen im Intervall  $1,2 - 3,2 \mu\text{s}$  gesucht.

### Maximum-Likelihood für eine Poisson-Verteilung:

Es werden nicht mehr die einzelnen Zeitmessungen betrachtet, sondern die Anzahl der Ereignisse pro Balken im Zeitspektrum. Mit dem exponentiellen Zerfallsgesetz wird die erwartete Anzahl von Ereignissen pro Balken bestimmt. Des Weiteren wird angenommen, dass die Anzahl der tatsächlich gemessenen Ereignisse  $N_i$  gemäß einer Poisson-Verteilung um den jeweiligen Erwartungswert  $f_i(\tau)$  gestreut sind. Damit ist die Wahrscheinlichkeit, im Balken  $i$   $N_i$  Einträge zu messen [6]:

$$P(N_i, \tau) = \frac{f_i^{N_i} \cdot e^{-f_i}}{N_i!} \quad (5.7)$$

Mit der Maximum-Likelihood-Methode und dieser Wahrscheinlichkeitsdichte kann wiederum der wahrscheinlichste Wert für  $\tau$  abgeschätzt werden. Dazu wird allerdings das Minimum von  $-2 \ln L$  gesucht, was äquivalent zum Maximum von  $\ln L$  ist. Die Nullstellensuche nach dem Ableiten erfolgt ebenfalls numerisch im Intervall  $1,2 - 3,2 \mu\text{s}$ .

### Maximum-Likelihood für eine Gaussverteilung:

Im Grenzfall großer Erwartungswerte  $f_i$  geht die Poisson-Verteilung in eine Gauß-Verteilung über. Dies ist in guter Näherung bereits bei  $f_i > 10$  der Fall. Damit erhält man für die Wahrscheinlichkeitsdichte [6]:

$$P(N_i, \tau) = \frac{1}{\sqrt{2\pi\sigma_i^2}} \cdot e^{-\frac{(N_i - f_i)^2}{2\sigma_i^2}} \quad \text{mit } \sigma_i = \sqrt{f_i} \approx \sqrt{N_i} \quad (5.8)$$

Nun kann, analog zur Vorgehensweise bei der Poisson-Verteilung, die Maximum-Likelihood-Methode angewendet werden, in dem  $-2 \ln L$  minimiert wird.

Die Ergebnisse, der verschiedenen Anwendungen, der Maximum-Likelihood-Methode sind in 5.2.5 abgebildet. Der Unterschied zwischen den einzelnen Methoden ist sehr klein, der zwischen den beiden Messproben hingegen ist enorm.

Mit den alten Messdaten, die mit dem **TTDC** aufgenommen wurden, erhält man für fast alle Intervallgrenzen, d.h. für  $t_{start} > 0,8 \mu s$ , Schätzwerte für  $\tau_0$ , die nah am Literaturwert liegen. Das die Werte für  $t_{start} < 0,8 \mu s$  schlecht sind, ist der Tatsache geschuldet, dass der Einfluss der negativ geladenen Myonen nicht berücksichtigt wurde.

Der Farbverlauf ist sehr gleichmäßig und nicht so chaotisch **wie bei der** Bestimmung mit Hilfe der Fit-Funktionen. Bei den Messdaten vom FPGA hingegen ist der Großteil der Schätzwerte weit vom Literaturwert entfernt. Es gibt eine deutliche Tendenz im Farbverlauf: Je größer  $t_{start}$  bzw.  $t_{ende}$  gewählt werden, desto größer wird der Schätzwert für die mittlere Lebensdauer. Diese Tendenz war bereits bei der Auswertung mit den Fit-Funktionen zu sehen. Das deutet darauf hin, dass zu viele lange Zerfallszeiten gemessen wurden. Da bei der Messung allerdings immer nach einem Startsignal das erste eingehende Stoppsignal die Zeitmessung beendet, wäre zu erwarten gewesen, dass zu viele kurze Lebensdauern gemessen werden.

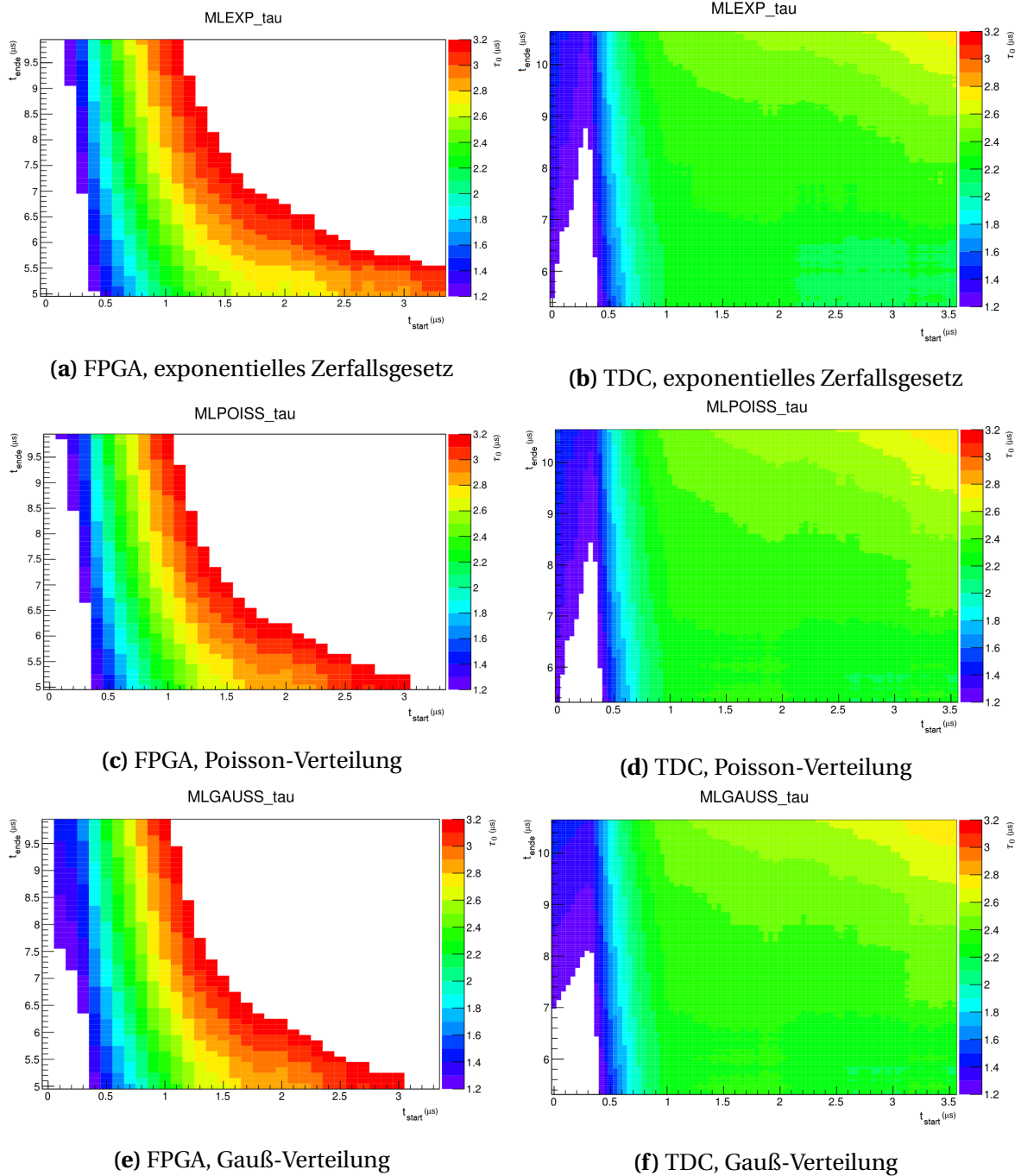
### 5.2.3 Zusammenfassung und Ausblick

Das mit dem FPGA aufgenommene Lebenszeitspektrum der Myonen deutet darauf hin, dass die Messung nicht fehlerfrei stattgefunden hat. Dies wird vor allem bei der Abschätzung der mittleren Lebensdauer, der positiv geladenen Myonen, mit der Maximum-Likelihood Methode deutlich.

Da die parallel durchgeführte Messung mit der analogen Ausleseelektronik ebenfalls ein schlechtes Ergebnis lieferte, liegt der Fehler vermutlich bei den Detektoren. Deren Signale werden zum Zeitpunkt der Fertigstellung dieser Arbeit bereits mit ADCs (Analog to Digital Converter) untersucht. Aufgrund der sehr überzeugenden Kontrollmessungen zur Zeitmessung und Datenübertragung, gehe ich davon aus, dass der FPGA auch brauchbare Ergebnisse erzielen wird, sobald das Problem behoben ist und der TLC ebenfalls wieder gute Daten liefert.

Der Fehler, der für die Messwerte mit  $t > 10 \mu s$  verantwortlich ist, konnte von mir mit Hilfe von Simulationen nicht reproduziert werden. Wenn es sich dabei lediglich um eine Außerkraftsetzung der Abbruchbedingung im Messmodul handelt, beeinträchtigt dieser Fehler die restliche Messung nicht. Korrekturen in der Firmware, um diesen Fehler zu beseitigen oder die Start- und Stoppbedingungen zu verändern, lassen sich **Problemlos** vornehmen. Diese Flexibilität ist eine der großen Stärken der digitalen Datenverarbeitung mit einem FPGA.





**Abbildung 5.2.5:** Auswertung des Lebenszeitspektrums mit der Maximum-Likelihood-Methode. Auf der X bzw Y-Achse sind die betrachteten Intervallgrenzen aufgetragen. Der geschätzte Wert für die mittlere Lebenszeit wird durch die Farbskala auf der z-Achse angegeben

# A Literaturverzeichnis

- [1] Particle Data Group. Leptons:  
<http://pdg.lbl.gov/2014/tables/rpp2014-sum-leptons.pdf> 4
- [2] Particle Data Group. Light unflavored Mesons:  
<http://pdg.lbl.gov/2014/tables/rpp2014-tab-mesons-light.pdf> 3
- [3] Particle Data Group. Strange Mesons:  
<http://pdg.lbl.gov/2014/tables/rpp2014-tab-mesons-strange.pdf> 3
- [4] Particle Data Group. Cosmic Rays:  
<http://pdg.lbl.gov/2014/reviews/rpp2014-rev-cosmic-rays.pdf> 3, 4
- [5] FPGA PROTOTYPING BY VHDL EXAMPLES, Xilinx Spartan<sup>TM</sup>-3 Version, von Pong P. Chu, John Wiley & Sons , Inc, 1. Auflage (2008) 18
- [6] Versuchanleitung zum Praktikumsversuch „Lebensdauer von Myonen“ des IKTP der TU Dresden 5, 8, 27
- [7] Feynman-Diagramme (Creative Commons) der Wikimedia Foundation, Inc.  
[commons.wikimedia.org/wiki/Category:Feynman\\_diagrams](https://commons.wikimedia.org/wiki/Category:Feynman_diagrams) 4
- [8] Datenblatt des EJ-208 Plastic Scintillator  
[http://www.eljentechnology.com/images/stories/Data\\_Sheets/Plastic\\_Scintillators/EJ208%20data%20sheet.pdf](http://www.eljentechnology.com/images/stories/Data_Sheets/Plastic_Scintillators/EJ208%20data%20sheet.pdf) 7
- [9] Datenblatt des XP2020 Photomultipliers  
<http://my.et-enterprises.com/pdf/XP2020.pdf> 7
- [10] Datenblatt der Spartan-3E FPGA Reihe  
[http://www.xilinx.com/support/documentation/data\\_sheets/ds312.pdf](http://www.xilinx.com/support/documentation/data_sheets/ds312.pdf)

[11] Userguide des Spartan-3E Starterboards

[http://www.xilinx.com/support/documentation/boards\\_and\\_kits/ug230.pdf](http://www.xilinx.com/support/documentation/boards_and_kits/ug230.pdf)

## Erklärung

Hiermit erkläre ich, dass ich diese Arbeit am Institut für Kern- und Teilchenphysik ohne unzulässige Hilfe Dritter verfasst und alle Quellen als solche gekennzeichnet habe.

Arne Heimbold

Dresden, November 2014