

2020 年度 卒業論文

電荷を区別しないミュー粒子と
負電荷の μ^- 粒子の寿命測定

奈良女子大学 理学部
物理科学科 高エネルギー物理学研究室

犬飼愛美 安西郁美

2021 年 3 月 1 日

目次

第 1 章	宇宙線・ μ 粒子	2
1.1	宇宙線とは	2
1.2	μ 粒子とは	3
1.3	弱い相互作用と真空中の μ 粒子の崩壊	4
1.4	μ 粒子の崩壊と寿命	6
第 2 章	測定原理とセットアップ	10
2.1	全体の流れ	10
2.2	シンチレーションカウンター	11
2.3	NIM	15
2.4	CAMAC	17
2.5	エレクトロニクスのセットアップ	22
第 3 章	μ 粒子の寿命測定	28
3.1	測定原理	28
3.2	データ収集	30
3.3	イベントの選定	32
3.4	TDC の時間較正	34
3.5	μ 粒子の寿命の結果	37
3.6	考察	49
第 4 章	まとめ	56
参考文献		58
付録 A	ファインマン図からの崩壊幅	59
A.1	3 体崩壊の $S^{\mu\nu}$ の計算	65
付録 B	宇宙線が同時に入射した場合の検討	69
B.1	69
付録 C	解析に用いたソースコード	73

C.1	ele_mag.c	73
C.2	ele_mag.csh	81
C.3	watch_daq.csh	83
C.4	rootfileOutput-mag.cc	85
C.5	tdc_2020.cc	88
C.6	tdc-calibration2020	90
C.7	histgramOutput2020	92
C.8	histlifetime2020.cc	94
C.9	histlifetime2020.cc	96

図目次

1.1	宇宙線の模式図	2
1.2	左図：光子が放出される場合、右図：W ボソンによる影響	5
1.3	μ^+ 粒子の崩壊のファインマン図	6
1.4	μ^- 粒子の崩壊のファインマン図	6
1.5	ミューオン原子の模式図	7
1.6	μ 粒子の原子核捕縛	7
2.1	セットアップの概略図	10
2.2	左図：シンチレーションカウンター 右図：シンチレーションカウンター 模式図	11
2.3	実際のシンチレーターの配置	12
2.4	シンチレーションカウンターの模式図	13
2.5	シンチレーター内の様子と入射 μ 粒子の信号	15
2.6	シンチレーター内の様子と入射 μ 粒子と e^- の信号	15
2.7	左図：本実験で使用している NIM モジュール 右図：主な NIM モジュール	16
2.8	LAM をチェックする行程	18
2.9	LAM をチェックするフローチャート	18
2.10	本実験で使用している CAMAC モジュール	19
2.11	Gate 信号と入力パルス	20
2.12	TDC の仕組み	20
2.13	データ収集プログラムのフローチャート	21
2.14	エレクトロニクスの全体図	22
2.15	ADC (μ) の回路図	23
2.16	ADC (μ) のロジック図	23
2.17	ADC (e) の回路図	24
2.18	ADC(e) のロジック図	24
2.19	TDC の回路図	25
2.20	TDC のロジック図	25
2.21	左図：取り付けたネオジウム磁石、 右図：磁石を取り付けるヨーク	26
2.22	装置の簡略図	26

3.1	放射性崩壊の図	29
3.2	TDC 分布の概形図	29
3.3	mut1,es1 の ADC 分布	32
3.4	μ^- 粒子 (電荷区別あり) の寿命測定の様式図	33
3.5	ADC(μ) に入力した Gate 信号のロジック図	33
3.6	左図:T3 カウンターの ADC(mut3)、右図:T4 カウンターの ADC(mut4) 分布	34
3.7	TDC 時間較正	34
3.8	TDC 時間較正についてのロジック	35
3.9	各 Delay time での TDC の値 (平均)	36
3.10	TDC の時間較正の結果 y 座標の誤差の大きさは、プロットしている点のサイズより小さい。Pull のグラフは順に y 軸の誤差に 1/1、1/2、1/5 かけた値をプロットしたも のである。	37
3.11	オシロスコープ このように 1 マス (約 1 cm 四方) に最小メモリが 5 メモリある	38
3.12	μ 粒子の TDC 分布	39
3.13	μ 粒子が同時に入射する事象のイメージ図	40
3.14	μ 粒子の TDC 分布	41
3.15	ADC のカット値	41
3.16	条件 1 (mut1 > 500、400 > es1 > 100) の TDC 分布	42
3.17	条件 2 (mut1 > 500、600 > es1 > 400) の TDC 分布	43
3.18	条件 3 (mut1 > 500、es1 > 600) の TDC 分布	43
3.19	条件 4 (800 > mut1 > 500、es1 > 100) の TDC 分布	44
3.20	条件 5 (1200 > mut1 > 800、es1 > 100) の TDC 分布	44
3.21	条件 6 (mut1 > 1200、es1 > 100) の TDC 分布	45
3.22	各サブサンプルから求めた μ 粒子の平均寿命	45
3.23	μ^- 粒子の TDC 分布	47
3.24	μ^- 粒子の TDC 分布 (横軸 4000、定数項あり、loglikelihood でのフィッ ト)	48
3.25	μ^- 粒子の TDC 分布 (2018 年度)	50
3.26	μ^- 粒子の TDC 分布 (2019 年度)	50
3.27	μ^- 粒子の TDC 分布 (2020 年度)	51
3.28	μ^- 粒子の TDC 分布 (2018+2019+2020 年度)	51
3.29	μ 粒子の TDC 分布 (2018 年度)	53
3.30	μ 粒子の TDC 分布 (2019 年度)	53
3.31	μ 粒子の TDC 分布 (2020 年度)	54
3.32	μ 粒子の TDC 分布 (2018+2019+2020 年度)	54

A.1	μ^- 粒子のファインマン図	60
B.1	μ 粒子が同時に入射する事象のイメージ図	69
B.2	TDC が $TDC > 2500$ の事象に対する T1 カウンターの ADC (e) の分布	70
B.3	TDC 分布 ($et1 < 100$)	70
B.4	TDC 分布 ($et1 < 80$)	71
B.5	TDC 分布 ($et1 < 50$)	71
B.6	TDC 分布 ($et1 < 30$)	71
B.7	T1 の ADC(e)(et1) の値が 150 以上の時の時間 (TDC) 分布 分布は一定値ではなく、 μ 粒子の崩壊による直線に下がる成分が存在し ている。	72

表目次

1.1	Compendium of total muon capture results for light nuclei.[4]	9
2.1	シンチレーターのサイズ	13
2.2	各シンチレーターの仕様	14
3.1	本実験のデータ収集の詳細	31
3.2	Delay time と TDC カウント数	36
3.3	各条件での μ 粒子の平均寿命 (横軸 4000、bin 数 800、定数項あり)	42
3.4	各選定条件での μ^- 粒子の平均寿命 この値は比較のみ。以降用いない	49
3.5	本実験の条件での μ^- 粒子の平均寿命 以降、この値を用いる	49
3.6	各選定条件での μ 粒子の平均寿命 比較のみ、この値は以降用いない	52
3.7	本実験の選定条件での μ 粒子の平均寿命 まとめでは τ_μ とする	52
3.8	μ 粒子 (電荷の区別あり、なし) の平均寿命	55
4.1	本実験の結果	56
B.1	et1 のカット値を入れた μ 粒子の寿命	72

第1章

宇宙線・ μ 粒子

1.1 宇宙線とは

宇宙線とは、宇宙から地球へ絶え間なく降り注ぐ高エネルギーの放射線のことである。大気圏外から地球に向かってくる放射線を「一次宇宙線」といい、主に陽子からなる。一次宇宙線が大気圏内に入り、大気中の原子核と相互作用してできる放射線を「二次宇宙線」という。二次宇宙線は主に μ 粒子、ニュートリノ、電子、陽電子、光子などからなる。

生じた二次宇宙線の粒子はさらに崩壊・生成を繰り返し、大量の二次粒子が発生する現象を「空気シャワー」という。このうち、地上に降り注ぐのはミュー粒子とニュートリノがほとんどである。

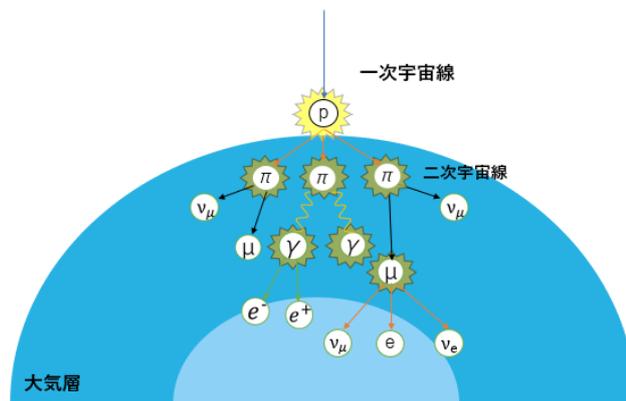
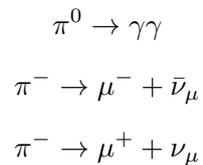


図 1.1 宇宙線の模式図

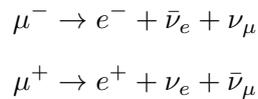
1.2 μ 粒子とは

μ 粒子は、 π 中間子の崩壊によって生成される。 μ 粒子は物質と相互作用するが、核力のような強い力では相互作用しないので電子と似ている。質量は 105.66MeV と電子 (0.511MeV) の約 207 倍である。 μ 粒子には正電荷と負電荷の μ^+ 、 μ^- の 2 種類がある。 μ 粒子のスピンは $1/2$ で μ^+ と μ^- は互いに粒子と反粒子の関係にあり、質量は等しい。

一次宇宙線の陽子は、大気中の原子核と衝突して二次宇宙線の一種である π 中間子を生成する。この π 中間子は π^0, π^+, π^- の 3 種類からなり、以下のように崩壊する。



このようにして μ 粒子が生成される。生成された μ 粒子は以下のようにほとんどの確率で崩壊する。 μ 粒子の平均寿命は $2.2\mu\text{s}$ であるが相対論的効果のために、多くの μ 粒子が崩壊前に地上まで到達する。



本実験ではこの μ 粒子の寿命を測定していく。

1.3 弱い相互作用と真空中の μ 粒子の崩壊

1.3.1 弱い相互作用

弱い相互作用は、 β 崩壊のような粒子の崩壊に関係する力であり、素粒子レベルの非常に近い範囲にしか働かない作用である。弱い相互作用を媒介する粒子は、 W^\pm ボソンと Z^0 ボソンである。 β 崩壊とは、原子核内で陽子が中性子に、中性子が陽子に変化する現象である。中性子過剰の原子核内では、核内の中性子が陽子と電子及び反ニュートリノに崩壊する。これが β^- 崩壊である。また、陽子過剰の原子核内では、核内の陽子が中性子、陽電子及びニュートリノに崩壊する。これが β^+ 崩壊である。

$$n \rightarrow p + e^- + \bar{\nu}$$

$$p \rightarrow n + e^+ + \nu$$

π 粒子の崩壊や、 μ 粒子の崩壊についても弱い相互作用が関係している。

1.3.2 フェルミ理論

フェルミ理論とは、弱い相互作用を記述する有効理論で、4つのフェルミ粒子が1点で直接的に、他の粒子を媒介せずに、相互作用すると仮定するものである。フェルミ理論によれば μ 粒子の崩壊の振幅 M は以下の式で与えられる。これは $V - A$ 理論とも呼ばれている。[1]

$$M = \frac{G_F}{\sqrt{2}} (\bar{\nu}_\mu \gamma^\mu (1 - \gamma^5) \mu) (\bar{e} \gamma_\mu (1 - \gamma^5) \nu_e) \quad (1.1)$$

ここで G_F はフェルミ結合定数 G_F である。ここで、 γ^μ はディラックの γ 行列で、 γ^5 は $\gamma^5 = i\gamma^0\gamma^1\gamma^2\gamma^3$ である。この振幅から μ 粒子の崩壊幅 (Γ) を求める手続きを付録 A にまとめた。

1.3.3 真空中 μ 粒子の崩壊

付録 A で示したように、 μ 粒子の崩壊幅は最低次の近似で次式で与えられる。

$$\Gamma = \frac{1}{\tau_{\mu^-}} = \frac{G_F^2 m_\mu^5}{192\pi^3} F\left(\frac{m_e^2}{m_\mu^2}\right) \quad (1.2)$$

ここで、 $F(x)$ は

$$F(x) = 1 - 8x + 8x^3 - x^4 - 12x^2 \ln(x) = 0.999813$$

である。

放射補正

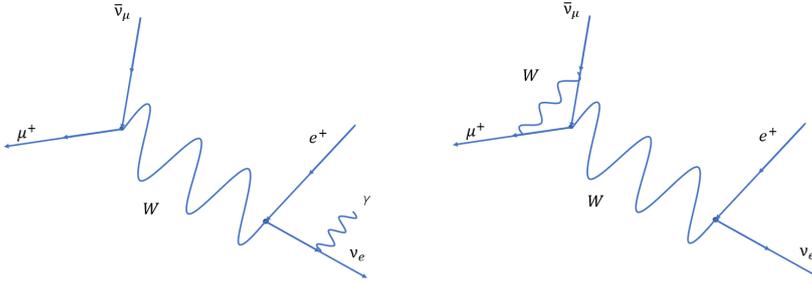


図 1.2 左図：光子が放出される場合、右図：W ボソンによる影響

最低次の崩壊幅 Γ には、一般的な崩壊の途中での光子の放出や、W ボソンによる影響の補正を加える必要がある。(図 1.2)

一般的な崩壊の途中での光子の放出や、W ボソンによる影響を補正した崩壊幅 Γ は以下のようなのである。[1]

$$\Gamma = \frac{1}{\tau_{\mu^-}} = \frac{G_F^2 m^5}{192\pi^3} F(x) \left(1 + \frac{3m_\mu^2}{5M_W^2} \right) \left[1 + H_1(x) \frac{\hat{\alpha}(m_\mu)}{\pi} + H_2(x) \frac{\hat{\alpha}^2(m_\mu)}{\pi^2} \right] \quad (1.3)$$

ここで、

$$x = \frac{m_e^2}{m_\mu^2}$$

$$H_1(x) = \frac{25}{8} - \frac{\pi^2}{2} - (9 + 4\pi + 12\ln(x))x + 16\pi^2 x^{\frac{3}{2}} + o(x^2) = -1.8079$$

$$H_2(x) = \frac{156815}{5184} - \frac{518}{81}\pi^2 - \frac{895}{36}\zeta(3) + \frac{67}{720}\pi^4 + \frac{53}{6}\pi^2 \ln 2 - \frac{5}{4}\pi^2 \sqrt{x} + o(x) = 6.7$$

$$\hat{\alpha}(m_\mu)^{-1} = \alpha^{-1} - \frac{2}{3\pi} \ln \left(\frac{m_\mu}{m_e} \right) + \frac{1}{6\pi} = 135.9$$

$$G_F = 1.16637 \times 10^{-5} (\text{GeV}^{-2})$$

である。また、 m_e は電子の質量、 m_μ は μ 粒子の質量、 M_W はボソンの質量である。既知の値を代入すると、

$$\tau_\mu = 2.197 \times 10^{-6} (\text{sec}) \quad (1.4)$$

である。これは、(1.3) 式に代入した結果である。後に示す測定値（文献値）の精度は計算結果よりも高い。

今回の本実験の μ 粒子の寿命は、フェルミ結合定数 G_F を実験的に求める最も精密な方法であることが知られている。フェルミ結合定数は最も基本的な物理パラメーターの 1 つであり、素粒子の標準理論の精密検証において重要な役割を果たしてきた。

1.4 μ 粒子の崩壊と寿命

1.4.1 μ^+ 粒子の崩壊

μ^+ は以下のように崩壊する。

$$\mu^+ \rightarrow e^- + \nu_e + \bar{\nu}_\mu$$

μ^+ は物質中で原子核に束縛されないので、物質中での μ^+ 粒子の寿命は、真空中での μ 粒子の寿命と同じである。

1.4.2 μ^- 粒子の崩壊

μ^- は以下のように崩壊する。

$$\mu^- \rightarrow e^- + \bar{\nu}_e + \nu_\mu$$

真空中の μ^- の振る舞いや平均寿命は μ^+ と等しいが、物質中では μ^- は原子核に束縛されてミュオニック原子を形成する過程が存在するので、物質中の μ^- の平均寿命は μ^+ と異なることが予想される。

入射した μ^- 粒子は物質中でミュオン原子を形成する。ミュオン原子は異種原子の一種で、原子中の e^- が μ^- に置き換わったものである。 μ^- (105.66MeV) は電子 (0.51MeV) よりも重いので電子よりも内側の独自の軌道を周回する。

原子核に束縛された μ^- は、以下のどちらかの過程をたどる。

1. 束縛軌道上での崩壊 (Decay In Orbit(DIO))

真空中と同様の過程で崩壊する。平均寿命も真空中と同様である。

$$\mu^- \rightarrow e^- + \bar{\nu}_e + \nu_\mu$$

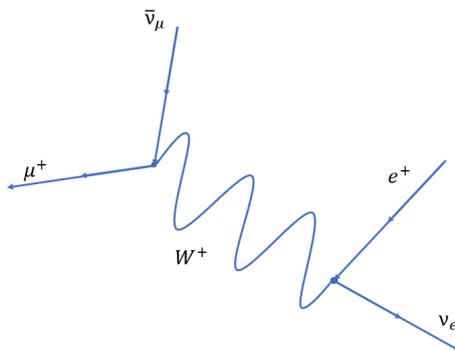


図 1.3 μ^+ 粒子の崩壊のファインマン図

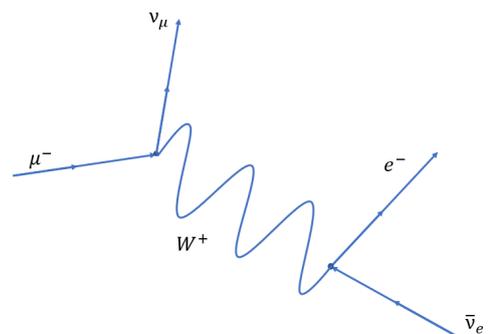


図 1.4 μ^- 粒子の崩壊のファインマン図

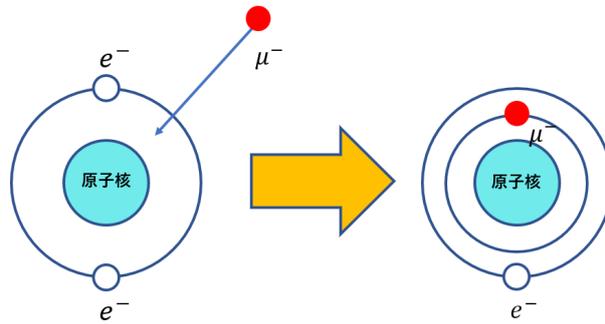


図 1.5 ミューオン原子の模式図

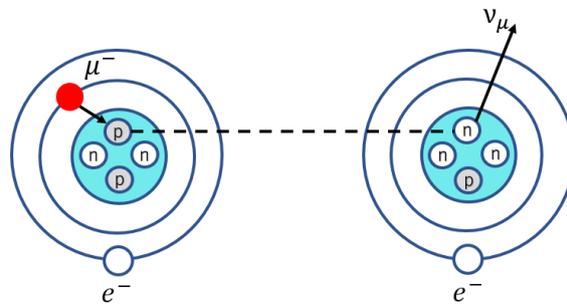
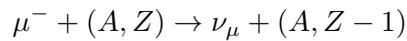


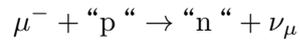
図 1.6 μ 粒子の原子核捕縛

2. 原子核捕縛

原子核との間の弱い相互作用により、原子核内の陽子が μ^- を吸収し中性子に変じ、ニュートリノを放出する。この結果、 μ^- の寿命が見かけ上縮まったことになる。この様子は以下の式で表される。



核子レベルでは、



この原子核捕縛が起こる割合 (Capture rate) は原子番号 Z に比例して大きくなるため、金属などの Z 番号の大きな物質中に入射した μ^- は原子核捕縛を起こしやすい。物質中の μ^- は、この 2 通りの崩壊過程のどちらかをたどる。したがって、次のセクションで示すように物質中の μ^- の寿命はこれらの過程が競合した結果として決定される。

1.4.3 物質中の μ^- 粒子の寿命

原子に束縛された軌道上での μ^- の崩壊の確率 P_{decay} と、 μ^- の原子核捕縛の確率 P_{capture} は以下のように表される。

$$P_{\mu^- \text{ total}} = P_{\text{capture}} + P_{\text{decay}} \quad (1.5)$$

Capture rate を Λ_{cap} とすると、上の式はこのように書き換えられる。

$$\begin{aligned} \exp\left(-\frac{t}{\tau_{\mu^-}}\right) &= \Lambda_{\text{cap}} t \cdot \exp\left(-\frac{t}{\tau_{\mu^+}}\right) \\ &= \exp\left(-(\Lambda_{\text{cap}}) + \frac{1}{\tau_{\mu^+}}\right) t \end{aligned} \quad (1.6)$$

すなわち、

$$\frac{1}{\tau_{\mu^-}} = \Lambda_{\text{cap}} + \frac{1}{\tau_{\mu^+}} \quad (1.7)$$

ここで、 τ_{μ^-} は全ての μ^- の平均寿命を表しており、 τ_{μ^+} は真空中の μ 粒子の平均寿命と同値である。式 (1.7) から分かるように、Capture rate が大きいほど μ^- の寿命は短くなる。

本実験で用いるプラスチックシンチレーターはポリスチレン ($[C_6H_5CHCH_2]_n$) 製で、H よりも C の電荷が 6 倍大きい。この表から、シンチレーターで観測できる μ^- の寿命は真空中での寿命より 10 % 程度短い $2020 \pm 20\text{ns}$ (測定値) をとると考えられる。この予想を、電荷を区別しない μ 粒子と、電荷を区別した μ^- 粒子の寿命を別々に測定することにより確認するのが本研究の目的である。

表 1.1 Compendium of total muon capture results for light nuclei.[4]

$Z(Z_{\text{eff}})$	Element	Mean life (ns)	Refs.
Positive muon		2197.03 ± 0.04	4.5
1(1.0)	$^1\text{H}^b$	$2194.0.3 \pm 0.066$	34
	$^1\text{H}^b$		35
		2194.53 ± 0.11	36
6(5.72)	C	2020 ± 20	40
		2043 ± 3	41
		2041 ± 5	42
		2040 ± 30	43
		2025 ± 4	1
		2035 ± 8	27
		2060 ± 30	44
		2030.0 ± 1.6	7
		2040 ± 10	73
		2029 ± 3	29
		2026.3 ± 1.5	a
	^{13}C	2045 ± 2	29
		2029.1 ± 3.0	a

第 2 章

測定原理とセットアップ

2.1 全体の流れ

本実験で用いた装置の概略図を図 2.1 に示す。まず上部に設置された磁石にて μ 粒子の電荷の区別を行う。荷電粒子 (今回は μ 粒子) がシンチレーションカウンターに入射すると、シンチレーターがシンチレーション光を発生し、光電子増倍管で電子を発生・増幅させ、アナログ信号として出力する。出力された波形は NIM モジュールを経て波形形成され、CAMAC モジュール内でデジタル信号化される。それを PC を用いて解析する。ほとんどの μ 粒子はシンチレーターを通過するが、一部がシンチレーター内で静止する。静止した μ 粒子は寿命に達すると崩壊して電子 (とニュートリノ) を放出する。放出された電子はシンチレーション光を発生するので、静止した μ 粒子は 2 つのシンチレーション信号を発生する。この 2 つの信号間隔を測定することで、 μ 粒子の寿命を決定できる。

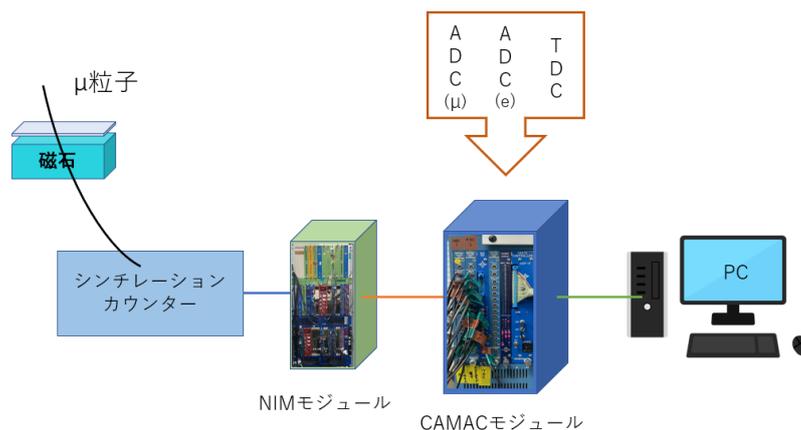


図 2.1 セットアップの概略図

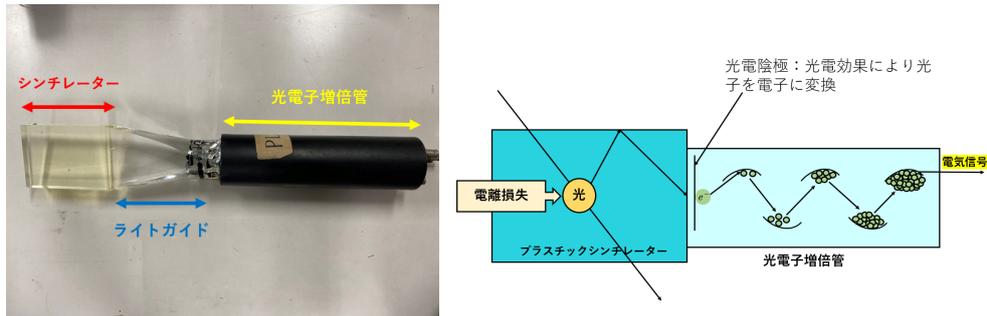


図 2.2 左図：シンチレーションカウンター 右図：シンチレーションカウンター模式図

2.2 シンチレーションカウンター

シンチレーションカウンターはシンチレーター、ライトガイド、光電子増倍管を光学的に結合させた放射線検出器のことである。放射線がシンチレーターを通過すると、シンチレーション光が放出される。この光は極めて微弱なため、光電子増倍管に直接あるいはライトガイドを通じて送られ、そこで電子数を $10^5 - 10^6$ 倍の増幅することで検出可能な電気信号に変換される。

2.2.1 シンチレーター

荷電粒子がシンチレーターを通過するとき、シンチレーターを構成する原子と分子が荷電粒子のエネルギーを吸収し、励起される。この吸収したエネルギーを光エネルギーとして放出する物質で、粒子のエネルギーに応じてシンチレーション光を放出する。シンチレーターには大きく分けて有機シンチレーターと無機シンチレーターの2種類がある。本実験では有機シンチレーターの一種であるプラスチックシンチレーターを使用する。

- 有機シンチレーター

有機シンチレーターは結合または縮合したベンゼン環構造を含む芳香族炭化水素化合物。 n 秒程度の速い減衰時間を持つので、高速計数、高計数率測定に使われる

- 無機シンチレーター

無機シンチレーターは、主に少量の活性剤不純物を含むハロゲン化アルカリの結晶（例えば NaI、CsI）である。発光時間が μ 秒程度と長く反応は遅いが、発光量は有機シンチレーターよりも大きいため低エネルギーでの測定や精度の高いエネルギーの測定に適している。

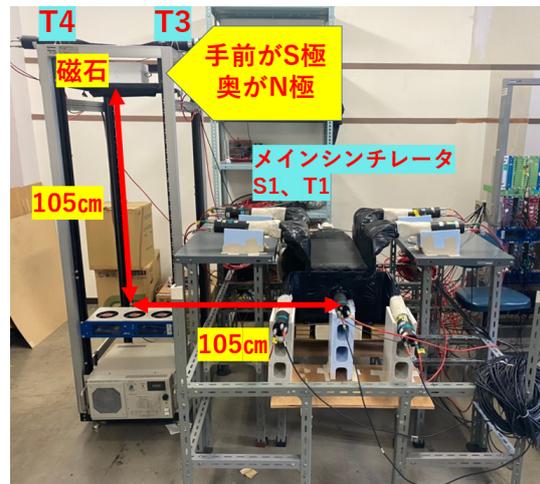


図 2.3 実際のシンチレーターの配置

2.2.2 ライトガイド

シンチレーターで発生した光を効率よく光電子増倍管に送るためのガイドで、着色のない優れた分光透過性を持ったアクリル棒やガラス光ファイバーを多数束ねたものが使用される。

2.2.3 光電子増倍管

光電子増倍管 (PMT : Photo Multiplier Tube) は、光電陰極、複数のダイノード、陽極 (アノード) で構成される。シンチレーターで発生した光子は、光電陰極に衝突すると、光電効果を起こして電子を発生させる。電子は電場によって最初のダイノードに向かって加速され、ダイノードに衝突すると、複数の二次電子を放出する。発生した二次電子は次のダイノードへと向かい再び多くの電子を放出する。これを順次繰り返すことで電子の数を増幅させていく。増幅された電子は陽極に収集され、分析可能な電気信号が得られる。

2.2.4 シンチレーションカウンターの配置

本実験では、中央に位置するメインシンチレーターと、その上部に設置した 1 個のシンチレーションカウンター (T1)、磁石の上部横並びに T3 と T4 のシンチレーションカウンター、合計 4 つを用いた。メインシンチレーターには直接、光電子増倍管 (S1) を取り付けている。表 2.1 に各シンチレーションカウンターの配置と仕様、サイズを示す。

表 2.2 は、本実験で用いた各シンチレーションカウンターの光電子増倍管 (PMT) の型番、それぞれにかけた電圧 (HV : High Voltage) とそのチャンネル番号、Discriminator の閾値 (Threshold)、各シンチレーターのペDESTアル値を表にしたものである。

表 2.1 シンチレーターのサイズ

	名称	横幅 (mm)	縦幅 (mm)	奥行 (mm)	個数 (個)
	(S1)	580	260	250	1
プラスチックシンチレーター	T3,T4	100	10	100	2
	T1	582	10	250	1
ライトガイド					3
光電子増倍管					4

2.2.5 トリガーカウンター

トリガーとは、いくつかのカウンターを用いて信号が来たと判断することであり、トリガーに用いるシンチレーションカウンターのことをトリガーカウンターと呼ぶ。

宇宙線はあらゆる方向から、絶え間なく降り注いでいる。本実験で用いる μ 粒子も同様に、あらゆる方向からシンチレーションカウンターに絶え間なく入射している。本実験ではシンチレーションカウンターに上から入射し、メインシンチレーター内で静止、その後崩壊し電子を放出した μ 粒子の寿命のみを測定したい。シンチレーションカウンターの側から入射した粒子のみを選定するために、複数のシンチレーションカウンターが同時に反応しているイベントを採用する。本実験ではメインシンチレーター上部に設置して

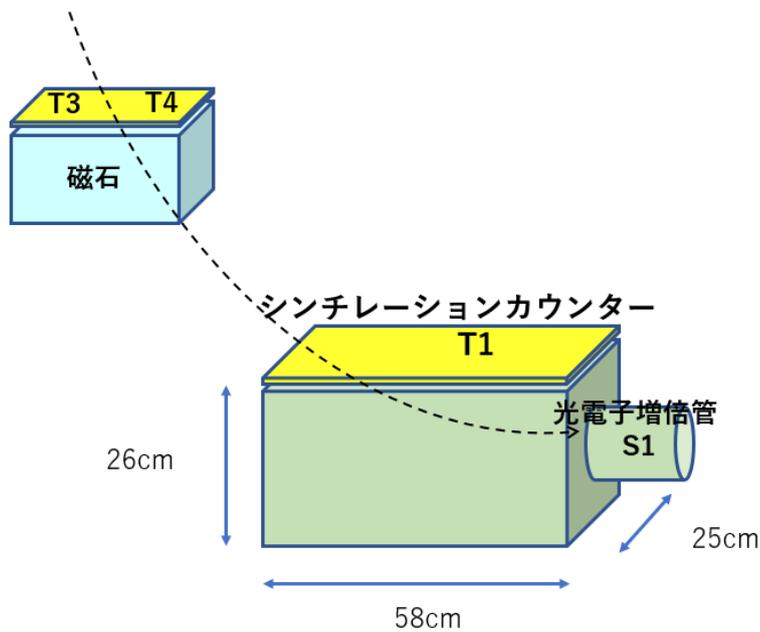


図 2.4 シンチレーションカウンターの模式図

表 2.2 各シンチレーターの仕様

カウンター	PMT 型番	HV (V)	閾値 (mV)	ペDESTAL (ADC(μ))	ペDESTAL (ADC (e))
S1	H161	2050	160	124	60
T1	H161	2250	100	89	56
T3	H7195	1600	200	20	
T4	H3983	1950	100	97	

シンチレーションカウンター T1 とメインシンチレーターに直接取り付けられた光電子増倍管 S1 をトリガーとしている。この T1 と S1 が同時に信号を発しているとき (T1 と S1 のコインシデンスが ON のとき) に μ 粒子が上部からメインシンチレーター内に入射した。すなわち、T1 と S1 のコインシデンスを取った信号がデータ収集のトリガーとなる。また、 μ^- 粒子を区別して測定する場合には、磁石上部に取り付けた T3, T4 のシンチレーションカウンターもトリガーカウンターとしている。T3 または T4 が信号を発し、さらに T1、S1 のトリガーが ON であるときに μ^- 粒子が入射した。なお、先行実験では使用したが、本実験ではカット値による選定を行っているため使用していない。

2.2.6 データ選別

シンチレーターに μ 粒子が入射した際、 μ 粒子がシンチレーターをそのまま通過する場合がほとんどで、 μ 粒子がシンチレーター内で静止するのは、通過する場合の数千分の 1 である。これを区別するには、通過した場合と静止した場合の信号の違いに注目する必要がある。

- μ 粒子がシンチレーターを通過する場合
観測される信号は図 2.5 のように、1 つのアナログ信号が見られる。
- μ 粒子がシンチレーター内で静止した後、崩壊する場合
観測される信号は図 2.6 のように、2 つのアナログ信号が見られる。1 つ目が入射した μ 粒子によるもので、2 つ目が μ 粒子の崩壊により放出された電子によるものである。以下、前者を μ 粒子の波形、後者を電子の波形と呼ぶ。

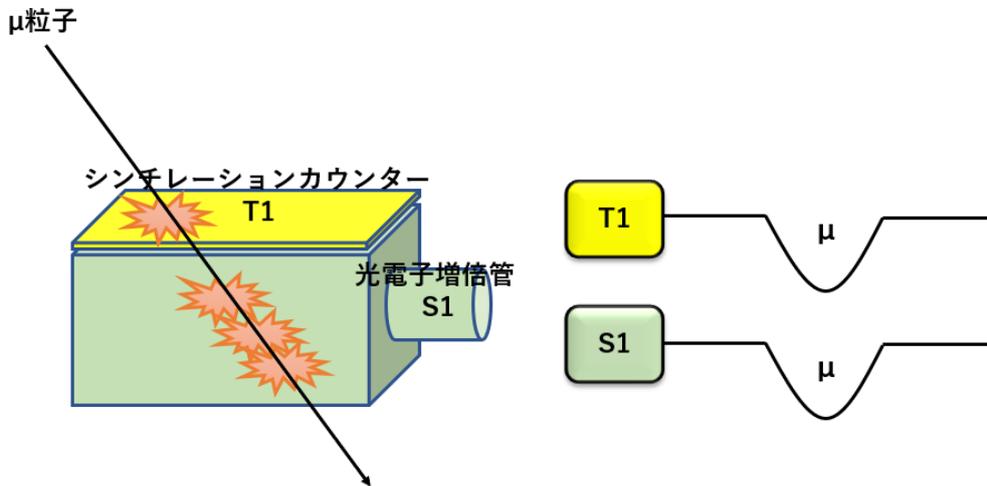


図 2.5 シンチレーター内の様子と入射 μ 粒子の信号

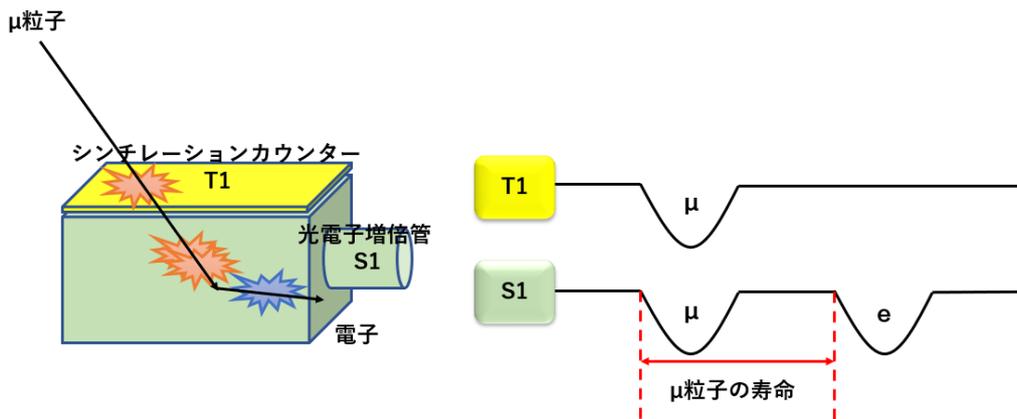


図 2.6 シンチレーター内の様子と入射 μ 粒子と e^- の信号

2.3 NIM

2.3.1 NIMとは

NIM(Nuclear Instrument Modules)とは、米国原子力委員会(AEC)(米国エネルギー省の全身)において1966年に制定された「放射線測定モジュール標準規格TID-20893」に準拠した原子核・素粒子実験用の電子機器の標準規格である。この規格を適用したモジュールがあれば、国内外、またはメーカーを問わず物理的及び電氣的に完全な互換性を有する。

2.3.2 NIMモジュール

NIM モジュールとは、NIM 規格に準拠した電子回路である。NIM モジュールはある条件の信号が入力されたときにロジック信号を出すようなモジュールのことで、NIM 信号のパルスの大きさは 1 が 0.7V で 0 が 0V である。以下本実験で用いた各モジュールを説明する。

- ディスクリミネーター：Discriminator(波高識別器)
ある特定の電圧（しきい値：threshold）を超えるパルス波高の入力信号の場合にのみ、ロジック信号 1 を出力するデバイス。
- デイレイ：Delay（遅延器）
入力された信号を任意の時間遅らせて出力する装置。
- コインシデンス：Coincidence（同時判別器）
2つ以上のロジック信号が時間的に一致するかどうかを決定し、一致する場合はロジック信号 1 を出力し、そうでないロジック信号 0 を出力する。出力パルスの時間幅を任意に調整できる。（コインシデンスは AND 回路である。）
- ゲートジェネレータ：G.G(Gate and delay Generator)
信号が入力されたとき、ロジック信号を出力するモジュール。出力するパルスの時間幅と Delay time を任意に調整・設定できる。G.G. は ADC（アナログ電気信号からデジタル電気信号に変換するモジュール）のゲート信号の生成によく用いられ、VETO 信号の生成にも用いられ、数ナノ秒から数秒までの範囲で幅を変えることができる。



図 2.7 左図：本実験で使用している NIM モジュール 右図：主な NIM モジュール

- ファンイン・ファンアウト：Fan-in Fan-out
いくつかの入力信号を受け入れ、足し合わせたものを出力に送る。(=OR 回路)
本実験では、入力信号を複数の同一信号に分割することにより、複数のチャンネルに1つの信号を分配できるようにするために使用している。
- スケーラー：Scaler
入力端子に入力されるロジック”1”信号の回数を数える装置。スイッチでオンオフが切り替えられる。

2.4 CAMAC

2.4.1 CAMAC 規格

CAMAC(Computer Aided Measurement And Control) とは、ハーウェル原子力研究所を中心にして使用実績のあった規格等が拡張発展され、ヨーロッパ統一規格として誕生したエレクトロニクス規格である。計算機器周辺でのデジタル化された情報の処理を各機能毎にモジュール化して行うことができる。実験装置等からの情報はモジュールの中で処理されると、クレートと呼ばれるプラグインを収納する箱の裏側の配線と結線される。これらの作業は全てクレートの正面から向かって右端に挿入される CC (Crete Contoller) が制御する。この CC は PC からの指示により動作する。PC から CC への命令には以下のアドレス (住所) とファンクション (動作) で指定する。

- アドレス
C,N,A,F の4つの数字で指定する。
 - C：クレート番号 (本実験で用いるクレートは1台。C = 1)
 - N：ステーション番号 (クレートの向かって左端から順に数える)
 - A：モジュール内のサブアドレス
 - F：動作 (Function)
- ファンクション
今回、主に使用したファンクションを記載する。
 - F(0)：Read Data
 - F(2)：Read and Clear Data
 - F(8)：Test LAM
 - F(9)：Clear Data
 - F(10)：Clear LAM
 - F(24)：Disable LAM
 - F(26)：Enable LAM

ここで、LAM (Look At Me) とは、モジュールからデータが読み出し可能になったことを PC 側に知らせる信号のことである。図 2.8 は LAM チェックをする行程、図 2.9 は

プログラム側から見たデータ収集のフローチャートを示している。

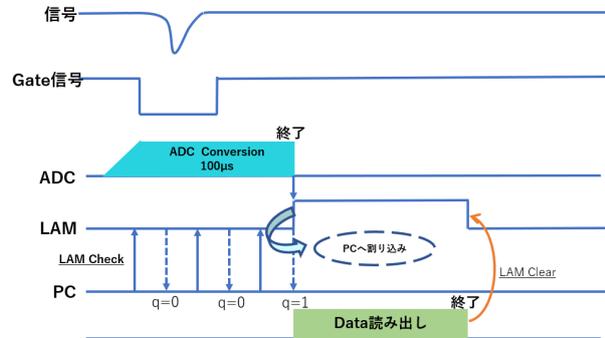


図 2.8 LAM をチェックする行程

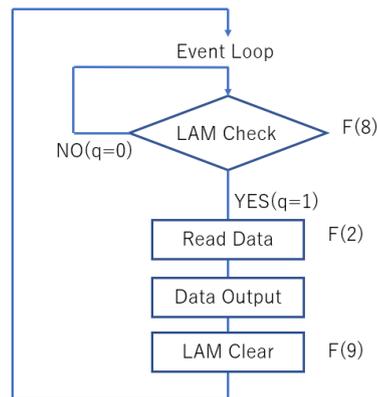


図 2.9 LAM をチェックするフローチャート

2.4.2 CAMAC モジュール

本実験で使用した CAMAC モジュールの各装置は以下の通りである。

- クレートコントローラ：CC(Crate Contoroller)
ADC や TC など制御して、データ収集を PC などで行うためのモジュール。
- ADC (Analoge to Digital Converter)
アナログ電気信号をデジタル電気信号に変えるモジュール。
 - － V モード (ピークホール型 ADC)
Gate パルスが開いている間に入ってきた信号のピーク電圧を数値化するもの。



図 2.10 本実験で使用している CAMAC モジュール

– Q モード（荷電積分型 ADC）

Gate パルスが開いている間に入ってきた信号を時間積分して数値化するもの。

今回実験で扱うプラスチックシンチレータは時間応答が速く、ピークの電圧値を取ることが難しいので Q モードを使用している。無機シンチレータは遅いので V モードでもピークの高さをとらえることができる。Q モードで測定される電荷量 Q は、信号の電流を i とすると、

$$Q = \int idt$$

と表せ、図の色部分の面積に相当する。一つのパルスに一つの値（これをチャンネルと呼ぶ）を対応させる。Gate 内に波形が入るようにするために信号側に Delay を入れる。本実験では ADC を 2 台使用する。以降 μ 粒子のエネルギー測定に用いる ADC (μ)、 e^- のエネルギー測定に用いる ADC (e) と呼称する。ADC は 11 ビット（最大 2046 カウント）である。

- TDC(Time to Digital Converter) 時間というアナログ量をデジタル信号に変換する装置のこと。Start 信号が入力されてから、Stop 信号が入力されるまで、一定の時間間隔でカウントを刻む。また、そのカウント数で時間間隔を表す。本実験で用いた TDC は 12 ビット（最大 4096 カウント）である。

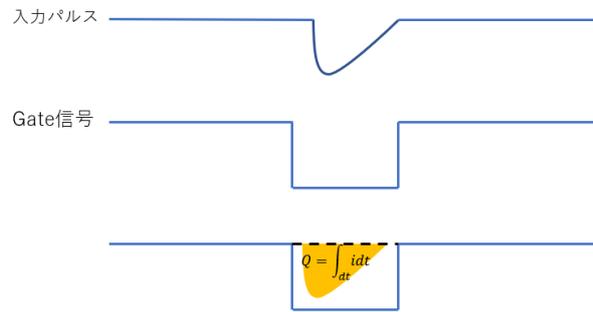


図 2.11 Gate 信号と入力パルス

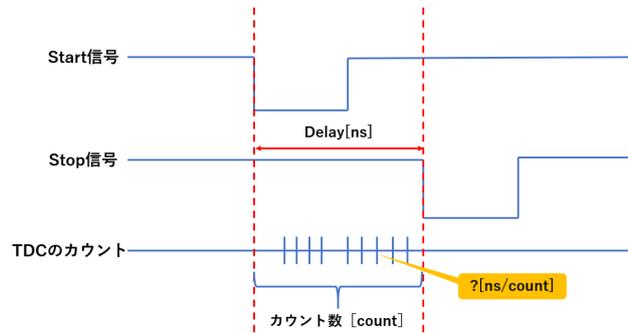


図 2.12 TDC の仕組み

2.4.3 データ収集プログラム

本実験では、CAMAC のデータ収集プログラムを C 言語を用い手作成した。まず、PC から ADC にテスト LAM の信号を送り、LAM OK の Q 信号 ($q \neq 0$) がかえってきたら ADC・TDC のデータを読み込む。データをファイルに書き込んだあと、データと LAM のクリアを行い再び ADC にテスト LAM の信号を送り、これをイベント数のだけ繰り返し行う。次のページ図 2.13 に、用いたデータ収集プログラムのフローチャートを示す。また、実際のプログラムを巻末の付録 C-1 に掲載した。

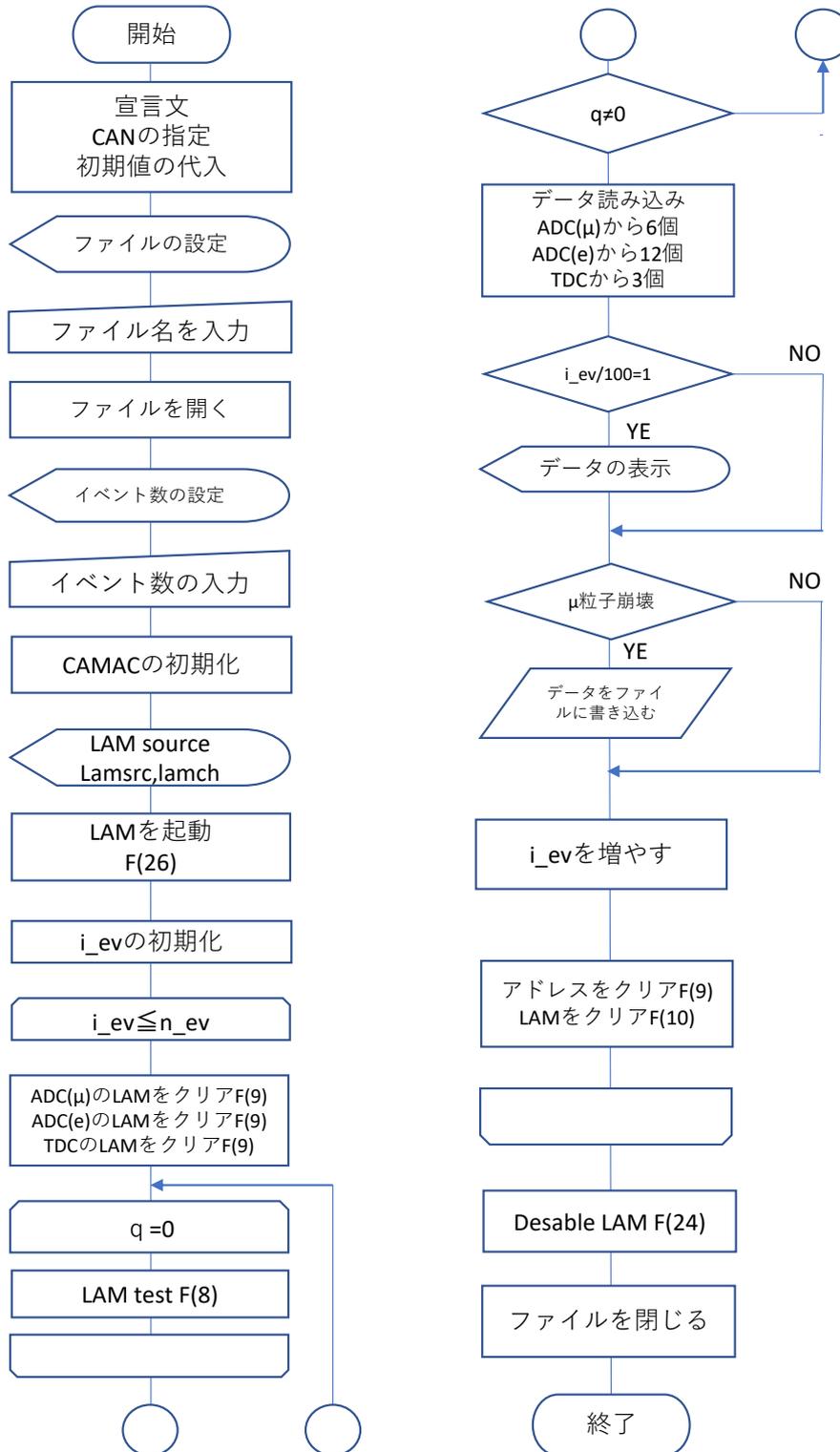


図 2.13 データ収集プログラムのフローチャート

2.5 エレクトロニクスセットアップ

本実験では ADC 2 台、TDC 1 台を使用した。ADC (μ) は μ 粒子のエネルギー測定、ADC (e) は μ 粒子が静止崩壊後に放出する電子のエネルギー測定、TDC は μ 粒子の寿命を測定する目的でセットした。図 2.14 は測定に使用したエレクトロニクスの全体図を示す。

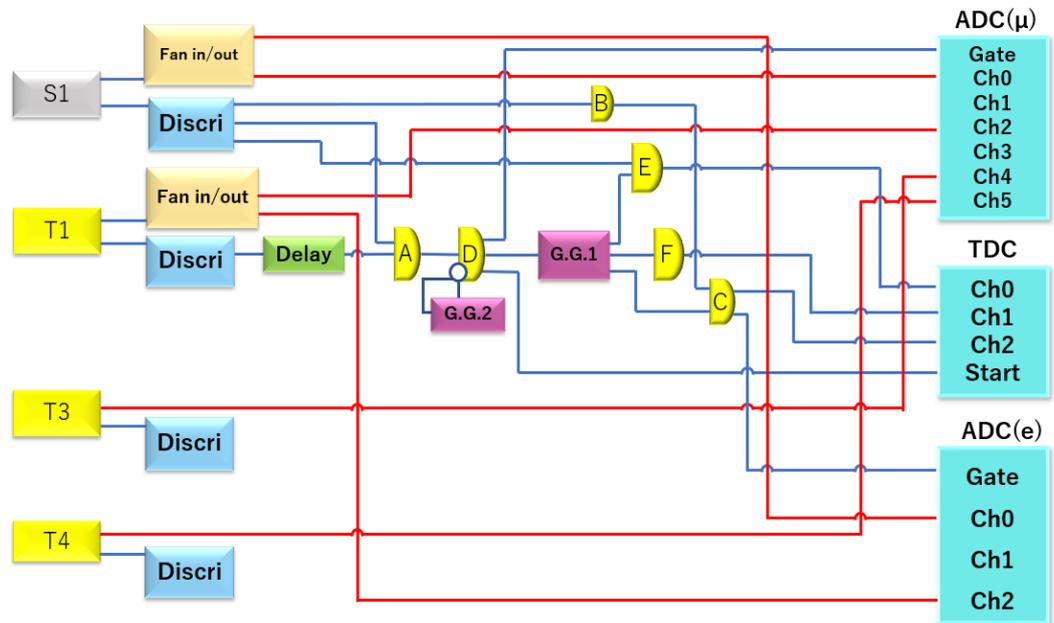


図 2.14 エレクトロニクスの全体図

2.5.1 入射した μ 粒子のエネルギー測定 (ADC (μ) のセットアップ)

μ 粒子がメインシンチレーター内に入射したときの信号を ADC で測定するためには、その信号を測定する ADC モジュール ADC (μ) に入力するデジタルのゲート信号を作成する必要がある。図 2.15 に ADC (μ) のゲートを作成する配線図を、図 2.16 にゲート信号が形成されるまでのタイミング図を示す。

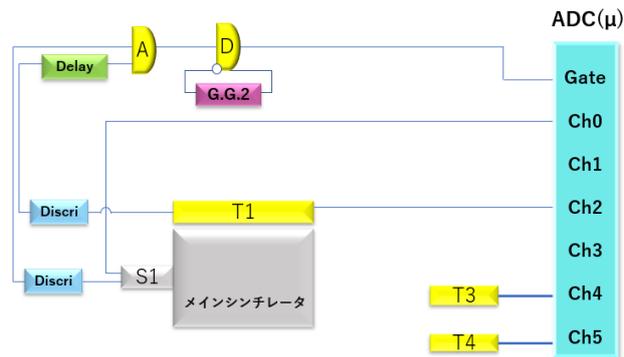


図 2.15 ADC (μ) の回路図

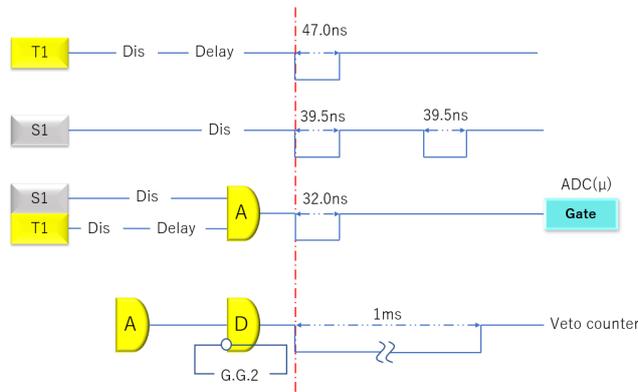


図 2.16 ADC (μ) のロジック図

Coincidence D は Veto カウンターの役割をしており、この Veto カウンターが開いている約 1ms の間は次の μ 粒子が入射した信号を受け付けない。これは、ADC のデータ収集中に次の信号が発生しないための対策である。

2.5.2 入射した μ 粒子が静止後、崩壊して生成された電子のエネルギー測定 (ADC (e) のセットアップ)

μ 粒子がメインシンチレーター内に入射し静止、その後崩壊して放出した電子の信号を ADC で測定するために、ADC(e) の Gate に入力するアナログ信号を作成する。図 2.17 はこの電子用のゲートの作成に用いた回路の配線図で、図 2.18 はそのタイミング図である。

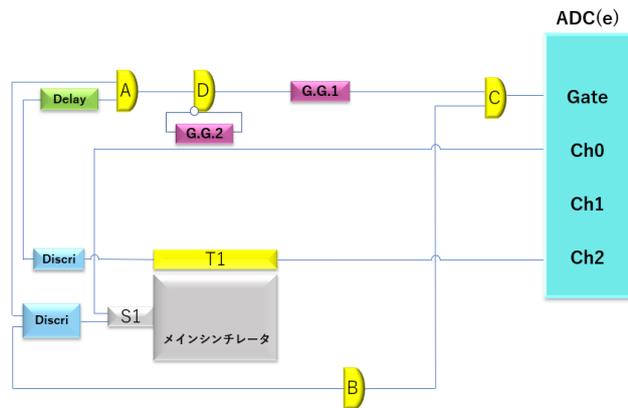


図 2.17 ADC (e) の回路図

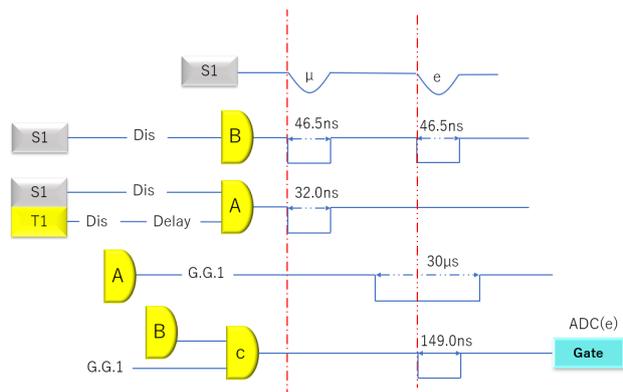


図 2.18 ADC(e) のロジック図

2.5.3 μ 粒子の寿命測定 (TDC のセットアップ)

μ 粒子の寿命を測定するために、TDC の start には μ 粒子がメインシンチレータ内に入射した時の信号を stop には μ 粒子がメインシンチレータ内に入射し静止、その後崩壊して電子を放出したときの信号を入力する。スタート信号には ADC(μ) の Gate 信号と同様の信号を、ストップ信号には ADC(e) の Gate 信号と同様の信号を入力している。以下に回路図とパルスのロジック図を示す。

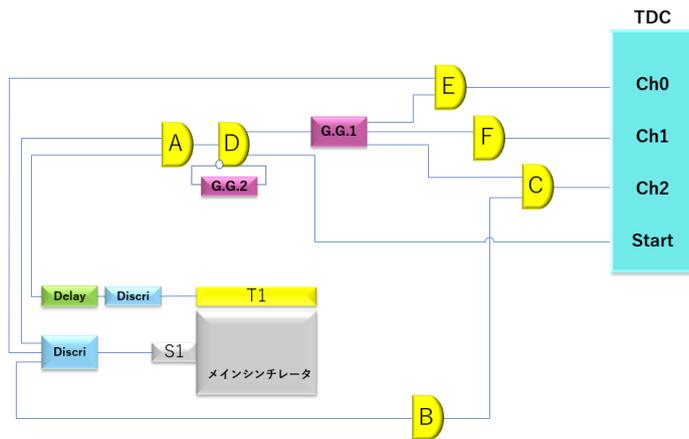


図 2.19 TDC の回路図

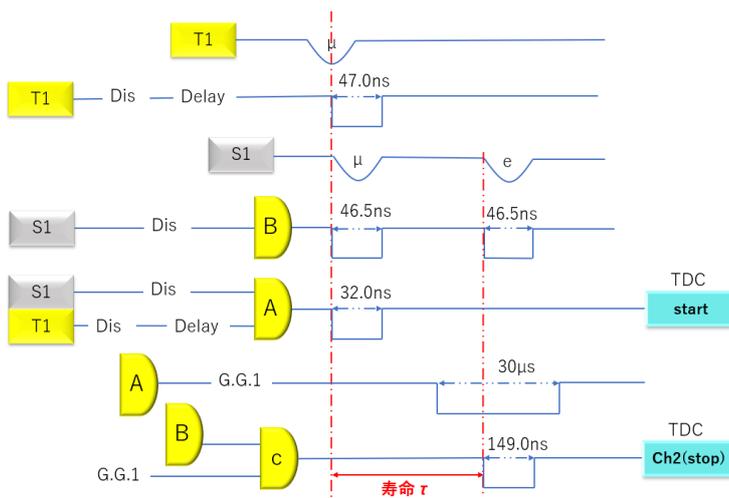


図 2.20 TDC のロジック図



図 2.21 左図：取り付けたネオジウム磁石、 右図：磁石を取り付けるヨーク

2.5.4 ネオジウム磁石

本実験では、 μ 粒子の負電荷を区別して寿命測定したい。そのために、2018 年度からネオジウム磁石が作る磁場を用いて区別した。

ネオジウム磁石

ネオジウム磁石は、主成分をネオジウム、鉄、ホウ素とする強力な磁力を持つ磁石である。さびやすいので、一般的には表面にニッケルメッキが施されている。それを図 2.21 のようにヨークに合計 16 個取付け、シンチレータ T3 と T4 の下に設置する。

2.5.5 ローレンツ力

磁石を置いたことにより、メインシンチレータ内に μ^+ 、 μ^- 粒子のどちらが入射されているのかを考える。

ローレンツ力は以下の式である。

$$\mathbf{F} = q(\mathbf{v} \times \mathbf{B}) \quad (2.1)$$

ここで、 \mathbf{F} は粒子にかかる力、 q は粒子の電荷、 \mathbf{v} は入射粒子の速度、 \mathbf{B} は磁石が作る磁束密度である。本実験の簡略図は以下の通りである。

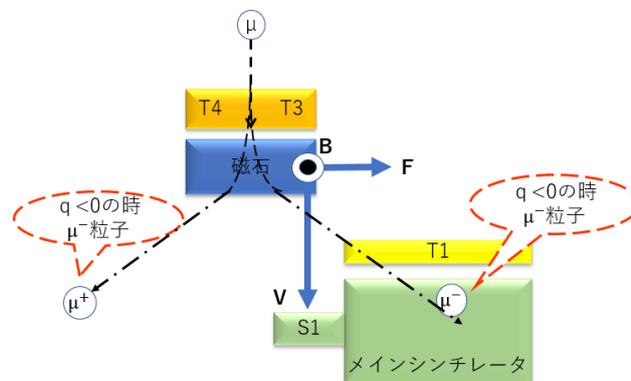


図 2.22 装置の簡略図

$q > 0$ のとき、力 \mathbf{F} は図 2.22 において左向きにかかり、 $q < 0$ の時、力 \mathbf{F} は右向きにかかる。つまり、本実験においてメインシンチレーター内に入射しているのは μ^- 粒子であるとわかる。

第 3 章

μ 粒子の寿命測定

3.1 測定原理

μ 粒子崩壊の時間的振る舞いは、放射性崩壊の指数関数法則に従う。本実験ではこの法則を用いて μ 粒子の寿命を測定する。

1 個の粒子が単位時間当たりに崩壊する確率を λ とし（これを崩壊定数という）、ある時間 t に存在する $N(t)$ 個の粒子が独立した粒子の集合とすると、微小時間 dt の間に崩壊する粒子の個数 dN は次式で表される。

$$dN = -\lambda N(t)dt \quad (3.1)$$

式 (3.1) を積分すると、

$$\begin{aligned} \int_{N_0}^N \frac{dN}{N} &= -\lambda \int_{t_0}^t dt \\ [\ln N]_{N_0}^N &= -\lambda [t]_{t_0}^t \\ \ln \frac{N}{N_0} &= -\lambda(t - t_0) \end{aligned} \quad (3.2)$$

$t_0 = 0$ の時、式 (3.2) は、

$$N(t) = N_0 e^{-\lambda t} \quad (3.3)$$

となる。 $N(t)$ は時刻 t で崩壊せずに残っている粒子の個数、 N_0 は初期時刻 t_0 での粒子の個数である。ここで、時刻 $t \rightarrow t + dt$ の間の微小時間 dt で崩壊する粒子の個数は式 (3.1) で表されるので、 N_0 個全ての粒子の生存時間を足し合わせたものは次式となる。

$$\begin{aligned} L &= \int_0^{\infty} tN(t)\lambda dt \\ &= \int_0^{\infty} tN_0 e^{-\lambda t} \lambda dt \\ &= \left[-\frac{1}{\lambda} N_0 e^{-\lambda t} \right]_0^{\infty} \\ &= \frac{N_0}{\lambda} \end{aligned}$$

よって、粒子の平均生存時間 L/N_0 、すなわち粒子の平均寿命 τ は、

$$\begin{aligned}\tau &= \frac{L}{N_0} = \frac{1}{\lambda} \\ \lambda &= \frac{1}{\tau}\end{aligned}\quad (3.4)$$

である。これを式 (3.3) に代入すると、

$$N(t) = N_0 e^{-\frac{t}{\tau}} \quad (3.5)$$

が得られる。式 (3.5) を図 3.1 に示す。

また、式 (3.5) を t で微分すると、単位時間当たりの崩壊数を表す式が得られる。

$$\frac{dN}{dt} = -\frac{N_0}{\tau} \exp\left(-\frac{t}{\tau}\right) \quad (3.6)$$

dN/dt は $t \rightarrow t + dt$ の間に N 個の粒子が $N + dN$ 個に減少したことを表すので、 dN は負である。 $t \rightarrow t + dt$ の間に放火した粒子の個数を dN_{decay} とすると、 $dN_{decay} = -dN$ なので、式 (3.6) は

$$\frac{dN_{decay}}{dt} = \frac{N_0}{\tau} \exp\left(-\frac{t}{\tau}\right) \quad (3.7)$$

となる。 μ 粒子がシンチレーターに入射、内部で静止後、崩壊して電子を放出するまでの時間を TDC で測定することで得られる TDC 分布が式 (3.7) に従っている。実験で得た TDC 分布を実験式

$$y(t) = p_0 \exp\left(-\frac{t}{p_1}\right) \quad (3.8)$$

で Fit することにより平均寿命 τ が算出できる。図 3.2 は式 (3.7) 及び TDC 分布の概形図である。

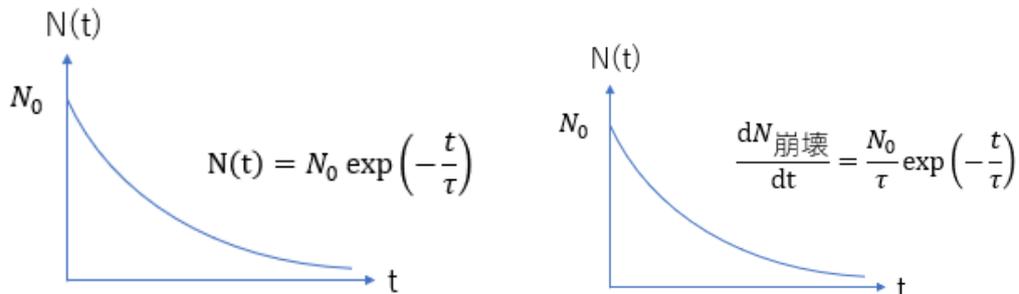


図 3.2 TDC 分布の概形図

図 3.1 放射性崩壊の図

3.2 データ収集

本実験では、第 2 章で示したセットアップで μ 粒子の寿命を測定した。用いたプログラムは付録 C に示した。以下、利用したプログラムの概要とデータ収集時に問題になったことやそのために取った対策をまとめておく。

1. データ収集プログラム

(a) ele_mag.c

本 C プログラムはデータ収集プログラム本体である。このプログラムのフローチャートを図 2.13 に示した。

(b) ele_mag.csh

csh は、ele_mag.c の実行モジュール ele_mag.c を起動するシェルスクリプトである。ele_mag.c のみだと手入力でファイル名とイベント数を指定、イベントを取り終えたあと再び、ele_mag.c を起動する必要があったが、ele_mag.csh 内で自動的に、保存するファイル名やイベント数を指定できるようにした。ファイル名にプログラム開始の日時が入るようにしている。

(c) watch_daq.csh

一日データ収集を続けていると夜の間データ収集が止まってしまうということが頻繁に発生した。watch_daq.csh は ele_mag.csh を動かすスクリプトで、イベント数を指定でき、ele_mag.c · ele_mag.csh が動いているかを自動的にある与えられた時間ごとに確認する操作を組み込んでいる。プログラムが動いていることを確認する操作としては、ele_mag.c で作成した ele_mag.c の動作中に作成されるファイルの確認、データを書き出しているファイルのサイズが増加しているか、実行モジュール (ele_mag.c) が動いているか (ps axuw で確認) をチェックする等の操作を行った。さらに指定した条件がクリアできなければ再び ele_mag を起動するようにシェル中でしている。これにより、データ収集が途中で止まる問題は大幅に改善された。しかしながら、本実験では、途中でデータが止まってしまう事が月に 2 回程度発生した。その原因は不明である。この場合には、watch_daq.csh 中で行っている出力ファイルのサイズのチェック等の機能も作動しなくなっており、今後どこで止まっているかの解明とプログラムの改良が必要である。

2. 収集データから root ファイルを作成するプログラム (rootfileOutput_mag.cc)

3. ADC 分布等のヒストグラムを作成する root の C プログラム (histgramOutput2020.cc)

4. μ 粒子の寿命を計算するために TDC 分布をフィットするプログラム (histlifetime2020.cc)

これらのプログラムを用いてデータを収集、及びデータ解析を行った。このうち、データ

収集プログラムの `ele_mag.csh` と `watch_daq.csh` が今年度の実験で新規に作成したものである。これらは上に述べたように、プログラム停止時に `ele_mag.c` を動かし、ファイル名イベント数などを指定していた作業を全て自動化するプログラムである。

本実験で収集したデータの総イベント数、測定時間などを次の表 3.1 に示す。

表 3.1 本実験のデータ収集の詳細

(a) データ収集		
開始日時	2020 年 10 月 15 日 19:44:19	
終了日時	2021 年 2 月 17 日 09:53:30	
測定時間	2,801 時間	
(b) データのイベント数		
イベント数	総イベント	115,900,008
	崩壊したイベント	1,038,320
	崩壊した μ^- のイベント	1,717
1 時間に取りれるイベント数	総イベント	41,378
	崩壊したイベント	370
	崩壊した μ^- のイベント	0.6

表 3.1 に示したように、データ収集は 2020 年 10 月 15 日に開始し、2021 年 2 月 17 日に終了した。この期間の全測定時間は 2801 時間である。収集したデータ量は 116M イベントで、これは 1 時間当たり 41,000 イベントに相当する。測定時間 2801 時間はデータファイルの開始時刻と終了時刻から計算したものであり、途中でデータ収集が止まっていたものを再び動かしたファイルもあることから実際より測定時間が長くなっていると考えられる。止まっていなかったファイルでは 1 時間に約 50,000 イベントがとれることが分かっており途中でデータ収集が止まっていた時間の割合は 20 % (1-41,000/50,000) 程度であることが分かる。1 日 (24 時間全く停止せずにデータ収集を続けることができると 1,200,000 イベントのデータをこのうち、シンチレータ内で静止しその後崩壊する μ 粒子の個数は 1 時間に 370 個で、さらに、電荷が負であると識別できた崩壊 μ 粒子の数は 0.6 イベント/時間で、非常に少数であることが分かる。

3.3 イベントの選定

3.3.1 μ 粒子の寿命測定（電荷の区別なし）

μ 粒子の寿命測定においてデータを使用するイベントは「 μ 粒子がメインシンチレーター内で崩壊したイベント」である。実験で取得した全 115,900,008 イベントのうち、次の条件に合うイベントを選定する。

信号事象の選定条件

1. μ 粒子がメインシンチレーターに入射

T1,S1 のコインシデンスが ON になっている。すなわち、データを取得できている。T1 を μ 粒子が通過したときに、ADC(μ) で取得した T1 の値 (mut1 と呼ぶ) が、カウンターが鳴っているとみなす値を上回っているイベントである。mut1 のヒストグラムを図 3.3 に示す。T1 カウンターの端を通過したイベントをのぞくために、mut1 > 500 のとき、T1 を μ 粒子が通過したと見なした。

2. μ 粒子がメインシンチレーターで崩壊

メインシンチレーター内で μ 粒子が静止しその後崩壊すると電子が放出される。つまり ADC(e) で取得した S1 の値 (以下 es1 と呼ぶ) が、ゼロでない値を持つはずである。es1 のヒストグラムを図 3.3 に示す。ここでも、 μ 粒子の崩壊がメインシンチレーターの端で起こったものでないことを保証するために、es1 の値が es1 > 100 であることを要求した。

3. TDC がタイムアウトしていない

TDC がタイムアウトした時、戻り値は 4000 カウント以上の値を示す。したがって、TDC カウント数が 4000 カウント以下であることを要求した。

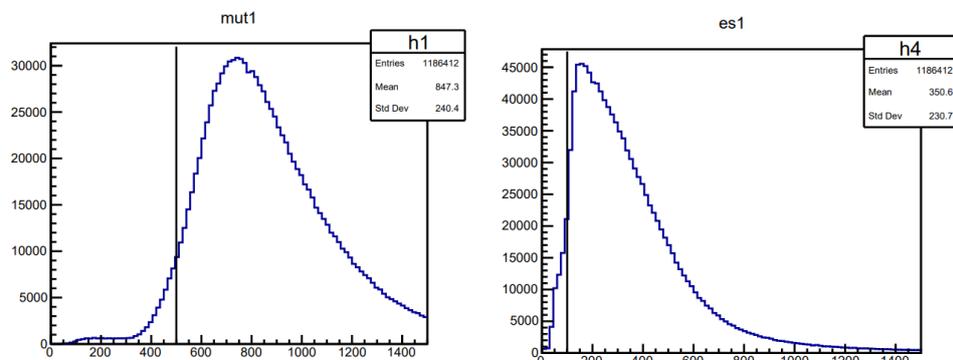


図 3.3 mut1,es1 の ADC 分布

以上 3 つの選定条件に合致した事象は全部で 1,038,320 イベントである。

3.3.2 μ^- 粒子の寿命測定（電荷の区別あり）

電荷区別ありの μ^- 粒子の測定に用いた装置の模式図を図 3.4 に示す。 μ^- 粒子の寿命測定においては、電荷が負であると分かっている μ^- 粒子を特定する必要がある。そのためには、この測定で使用したイベントには、「磁石を通過し、区別した μ^- 粒子のうち、メインシンチレータ内で崩壊したイベント」である。そのために、実験で取得した全 115,900,008 イベントのうち、次の条件に合うイベントを選定する。

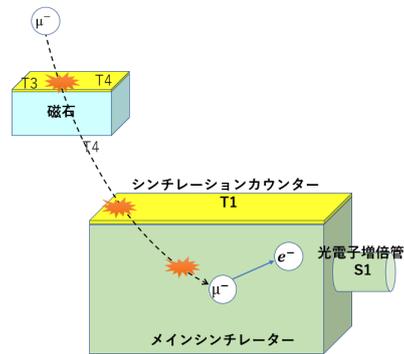


図 3.4 μ^- 粒子（電荷区別あり）の寿命測定の模式図

1. μ^- 粒子が T3 または T4 を通過

磁石を通過し、 μ^- 粒子の電荷が区別されていることを保証するために、磁石の真上においた。T3 または T4 のどちらかに μ^- 粒子の通過を示すヒットがあること、具体的には、T3 または T4 の ADC の値がある値以上を上回っていることを要求する。T3 と T4 の ADC 分布を図 3.6 に示す。両方のヒストグラムに粒子の通過を示す m.i.p のピークが見えている。

2. 前章と同様の選定条件

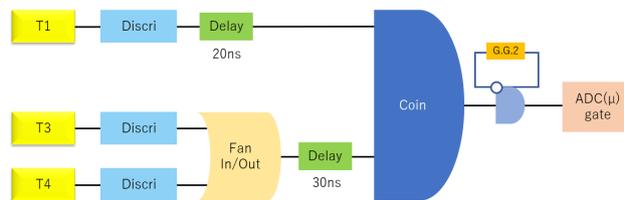


図 3.5 ADC(μ) に入力した Gate 信号のロジック図

図 3.6 に示すように、光電子増倍管のノイズをのぞいて、m.i.p の事象を選択するように、図の縦棒以上の ADC の値をどちらかのカウンターが持つ条件 ($300 < \text{mut3}$ または

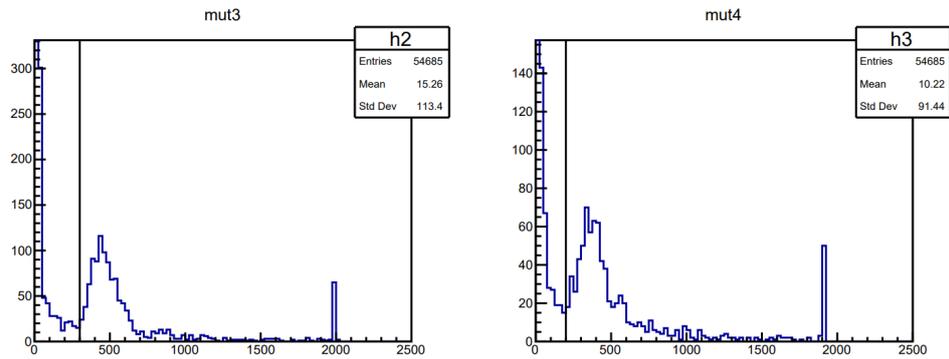


図 3.6 左図 : T3 カウンターの ADC(mut3)、右図 : T4 カウンターの ADC(mut4) 分布

200 < mut4) を要求した。

この選定条件に加えて、前章に示した条件 1,2,3 を全て満たした事象は全部で 1,717 イベントであった。

3.4 TDC の時間較正

前節で得られた TDC カウント数 p_1 を秒に換算するために、TDC カウント数と実際の時間との対応を時間間隔が分かった信号を用いて調べる必要がある。これを TDC の時間較正と呼ぶ。

この TDC の時間較正を行うために、Clock Generator で作成した 1 つ目のパルスを Start 信号、そこから特定の時間だけ Delay させた 2 つ目のパルスを Stop 信号として、その間の時間を TDC で測定する回路を別途セットアップした。TDC の時間較正に用いるロジックを図 3.8 に示す。パルスを Delay させる時間を変化させ、それに応じた TDC のカウントの変化を見ることで、TDC の 1 カウント当たりの時間を算出することができる。

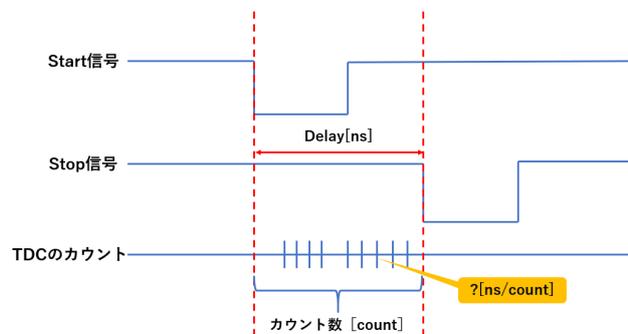


図 3.7 TDC 時間較正

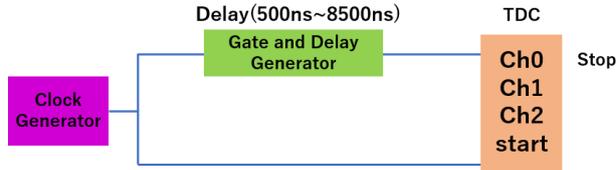


図 3.8 TDC 時間較正についてのロジック

スタートとストップパルスの時間間隔の調整は、Gate and Delay Generator(G,G)で行った。G.G. では Stop に入力する 100Hz のパルスの Delay time を 500ns~8500ns まで 500ns ずつ変化させ、それぞれの場合について TDC で Start と Stop の間隔を測定した。本来 20 μ s まで測定するべきだが、Gate and Delay Generator の Delay の変化の限界が 8500ns であったためそれまでの値を取った。実際の Start と Stop の間隔はオシロスコープで測定し、その読み取り誤差は、オシロスコープで start と stop の時間差を測定した時に、オシロスコープの最小めもりの 1/2 とした。表 3.2 に示すように、その誤差の値は時間のレンジ後ごとに異なり、10nsec から 100nsec である。

ここで、TDC で測定したカウント数は 1000 イベントのデータの平均値である（図 3.9）。この結果を横軸が Delay time [ns]、縦軸が TDC カウント数 [count] としてグラフを図 3.10 である。図は y 軸が TDC[count]、x 軸が Delay time[ns] であり、TDC の全測定誤差 Δy は実次式により求めた。

$$\Delta y_t = 0.2[\text{count/ns}] \times \Delta x$$

$$\Delta y = \sqrt{\Delta y_{\text{mean}}^2 + \Delta y_t^2}$$

ここで、 Δy_{mean} は図 3.9 で示した TDC 分布から決まる TDC 値のセンターの値であり、この誤差は次の述べる誤差に比べて十分小さい。 Δy_t は、x 軸の誤差に相当する Delay time をオシロスコープで決めた時の読み取り誤差の影響である。式 0.2count/ns は直線の傾きである。図 3.10 の右図の y 軸は、「Pull = (TDC カウント数-Fit 関数の値) / (y 軸の誤差)」をプロットしたものである。

データを 1 次の関数 ($y = p_1x + p_0$) でフィットした結果、直線の傾きとオフセットの値として、

$$p_1 = 0.2003 \pm 0.0013$$

$$p_0 = -2.06 \pm 2.75$$

$$\chi^2/\text{ndf} = 0.1056/15$$

を得た。

オフセット p_0 の値はゼロと無矛盾であり、Delay time と TDC カウント数は原点を通る綺麗な比例関係にあることが分かる。 $\frac{1}{\text{傾き}_a}$ は TDC 1 カウント当たりの時間 $\alpha[\text{ns}]$ に

表 3.2 Delay time と TDC カウント数

Delay time[ns]	500±10	1000±20	1500±20	2000±20
TDC[count]	97.752±0.019	198.220±0.016	299.609±0.020	398.450±0.017
	2500±50	3000±50	3500±50	4000±50
	497.889±0.017	600.421±0.016	700.063±0.016	799.606±0.020
	4500±50	5000±50	5500±100	6000±100
	898.942±0.018	998.872±0.018	1098.771±0.018	1198.591±0.020
	6500±100	7000±100	7500±100	8000±100
	1299.958±0.018	1398.594±0.020	1498.123±0.015	1598.454±0.017
	8500±100			
	1698.211±0.017			

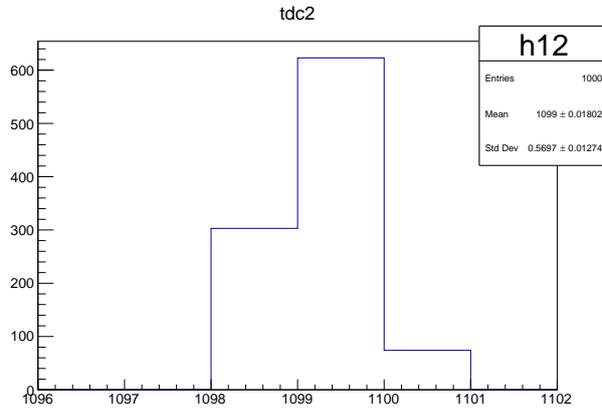


図 3.9 各 Delay time での TDC の値 (平均)

相当する。これにより、

$$\alpha = \frac{1}{0.2003} = 5.00 \quad \delta\alpha = \sqrt{\left(\frac{\delta p_1}{p_1}\right)^2} \times \alpha = 0.03255 \quad (3.9)$$

$$\therefore \alpha = 5.00 \pm 0.03[ns] \quad (3.10)$$

と求められる。

なお、オシロスコープでの読み取り誤差を今回は最小メモリの 1/2 と評価したが、それを用いたフィット結果の χ^2/ndf は 0.105/15 と誤差の大きさが正常な場合に予想される値である 1 と比べてだいぶ小さい。これは読み取り誤差が実際にはもっと小さいことが予想される。そこで、最小メモリの 1/5、1/10 であると変更して、再度フィットを行ったが、 χ^2/ndf の値はそれほど大きく変化せず、 p_1 もセンターの値はそのまま、誤差が多少小さくなるだけであった。現在の誤差の評価で用いている基準を最小メモリ 1/2 は誤

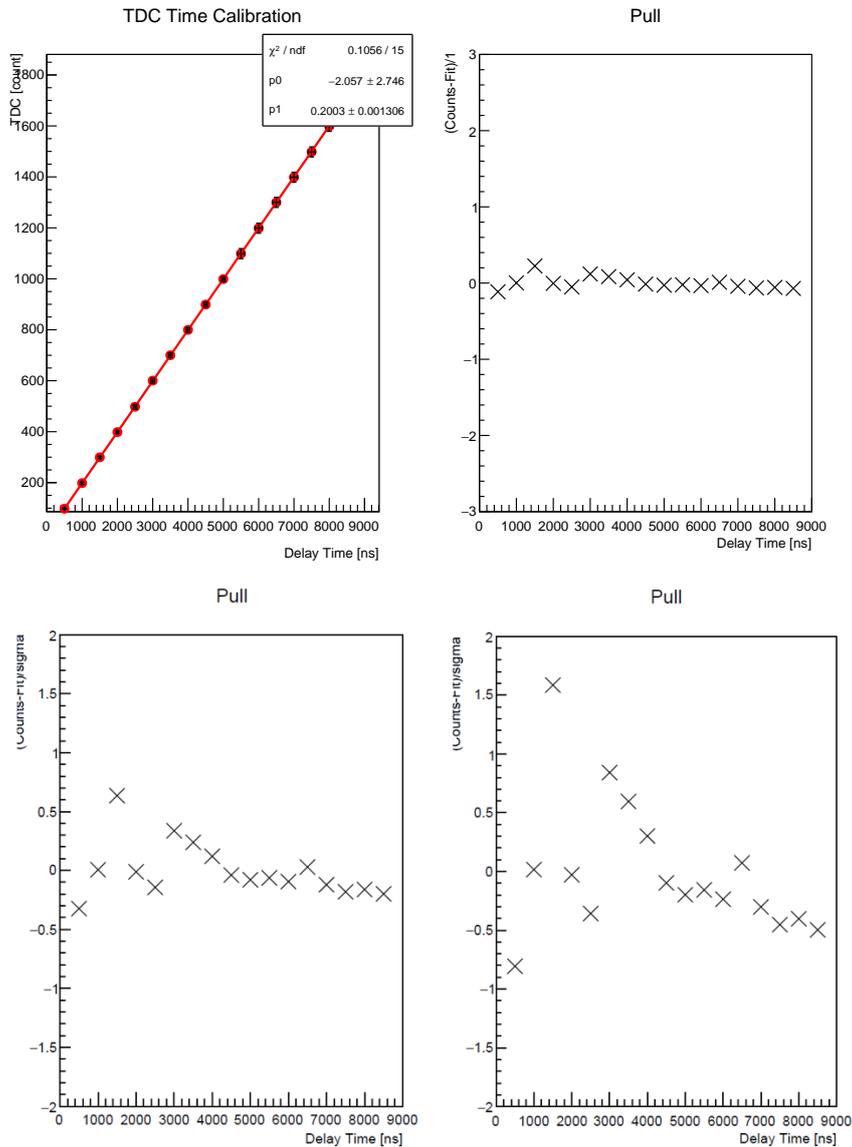


図 3.10 TDC の時間較正の結果

y 座標の誤差の大きさは、プロットしている点のサイズより小さい。Pull のグラフは順に y 軸の誤差に 1/1、1/2、1/5 かけた値をプロットしたものである。

差を大きく見積もっていることになるが、誤差を大きめに見積もった保守的な値として以後の解析に用いることにする。

3.5 μ 粒子の寿命の結果

本章では、データから μ 粒子の寿命を求めるために行った様々なフィットの方法とその結果について報告する。

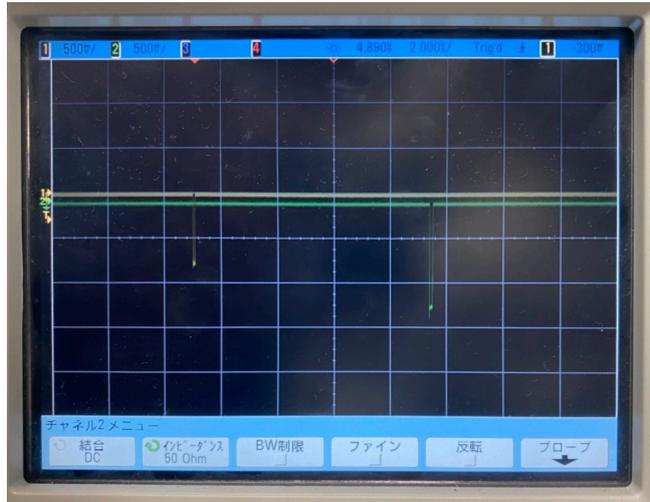


図 3.11 オシロスコープ

このように 1 マス (約 1 cm 四方) に最小メモリが 5 メモリある

3.5.1 μ 粒子の平均寿命 (電荷の区別なしの場合)

選定条件、電荷区別なしでのデータ解析の結果を示す。電荷区別なしの解析については、

1. 先行実験と同じ方法でのフィット
2. より広い TDC のレンジのデータのフィット

の順で行う。

1. 先行実験と同様に、イベントの選定を行った μ 粒子の TDC 分布を第 3.1 節で述べたように式 (3.8) で Fit する。

$$y(t) = p_0 \exp\left(-\frac{t}{p_1}\right)$$

ここで右図 Pull とは、

$$\text{Pull} = \frac{\text{ヒストグラムの値 } y[\text{counts}] - \text{Fit 関数の値 } y_{\text{fit}}}{\sqrt{\text{ヒストグラムの値 } y[\text{counts}]}}$$

を算出しプロットしたものである。 μ 粒子の平均寿命を表す TDC カウント数 $p_1[\text{count}]$ 、第 3.3 章で求めた TDC1 カウント当たりの時間 $\alpha[\text{ns}]$ を用いて μ 粒子

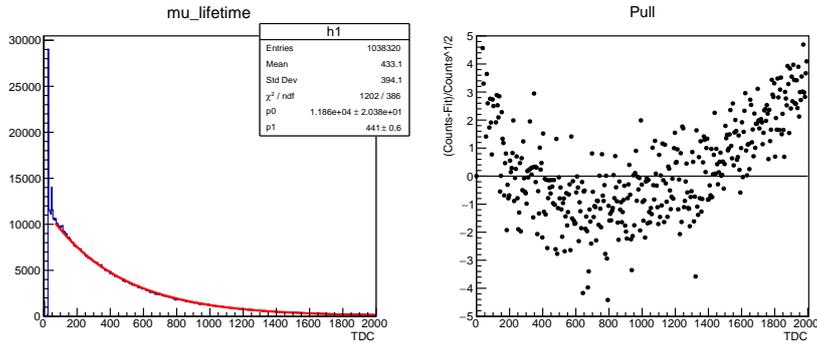


図 3.12 μ 粒子の TDC 分布

の寿命 $\tau_\mu[\text{ns}]$ は次のように求められる。

$$\begin{aligned}
 \tau_\mu[\text{ns}] &= p_1[\text{count}] \times \alpha[\text{ns/count}] \\
 &= 441.0 \times 5.00 \\
 &= 2205.0[\text{ns}] \\
 &= 2.2050[\mu\text{s}]
 \end{aligned} \tag{3.11}$$

また、平均寿命 τ_μ の誤差 $\delta\tau_\mu$ は間接誤差の誤差伝搬により、以下の式を用いて

$$\begin{aligned}
 \frac{\sigma_{\tau_\mu}}{\tau_\mu} &= \sqrt{\left(\frac{\sigma_\alpha}{\alpha}\right)^2 + \left(\frac{\delta p_1}{p_1}\right)^2} \\
 &= \sqrt{\left(\frac{0.03}{5.00}\right)^2 + \left(\frac{0.6}{441.0}\right)^2} \\
 &= 0.006152
 \end{aligned} \tag{3.12}$$

$$\therefore \sigma_{\tau_\mu} = 2.2050 \times 0.006152 = 0.013565 = 0.014[\mu\text{s}] \tag{3.13}$$

となる。このとき、 δp_1 は p_1 の誤差、 σ_α は α の誤差である。測定で求められた値 p_1, α より、本実験で得られた結果から導き出される μ 粒子の寿命 τ_μ は、

$$\tau_\mu = 2.205 \pm 0.014[\mu\text{s}]$$

となった。

ここで、グラフの χ^2 の値を見てみる。 χ^2 は以下の式で与えられ、この χ^2 を自由度 ndf で割った値が 1 に近づくと、うまく Fit 出来ているといえる。

$$\chi^2 = \sum_N \frac{(y_i - f(x_i))^2}{\delta y_i^2}$$

ここで、 x_i は bin の中心値、 y_i は各 bin の係数、 $f(x_i)$ は x_i における期待値である。しかし、グラフの値は、

$$\chi^2 = 1202$$

となっていて、

$$\chi^2/ndf = 1202/386 = 3.1139$$

となる。また、Pull の分布が 2 次曲線の形をしており、一様でない。これは何らかのシステムティックスが残っていることを示唆している。よって、Fit が上手くできているとは言えない。

- より広い TDC のレンジで、Fit 関数も見直すことにした。時間は TDC 値で最大 4000 カウントまで測定されているので、TDC を伸ばした。その時の TDC 分布を図 3.12 に示す。ここで、縦軸を対数グラフにしている。この対数グラフからわかるように、寿命の分布は exp の関数から予想される直線ではなく、TDC が 2500 カウント以降で、平坦な成分を (1 % 程度) 含んでいることが分かる。この成分は図 3.14 に示したように、 μ 粒子の崩壊によるものではなく、宇宙線が 2 個同時に入射したランダムな事象と思われる。このランダムな事象の効果を考慮するために、Fit 関数に定数項を付け加えた。改めて、Fit 関数は次のようになる。

$$y(t) = p_0 \exp\left(-\frac{t}{p_1}\right) + \alpha$$

縦軸は定数項が分かりやすいように log スケールで表示している。右の Pull の分布から分かるように、このように定数項を加えたときに Pull 分布はフラットで、関数がデータをよく再現していることが分かる。また、 $\chi^2/ndf = 1.0667$ であり 1 に近いことから、データをよくフィットしていると言える。

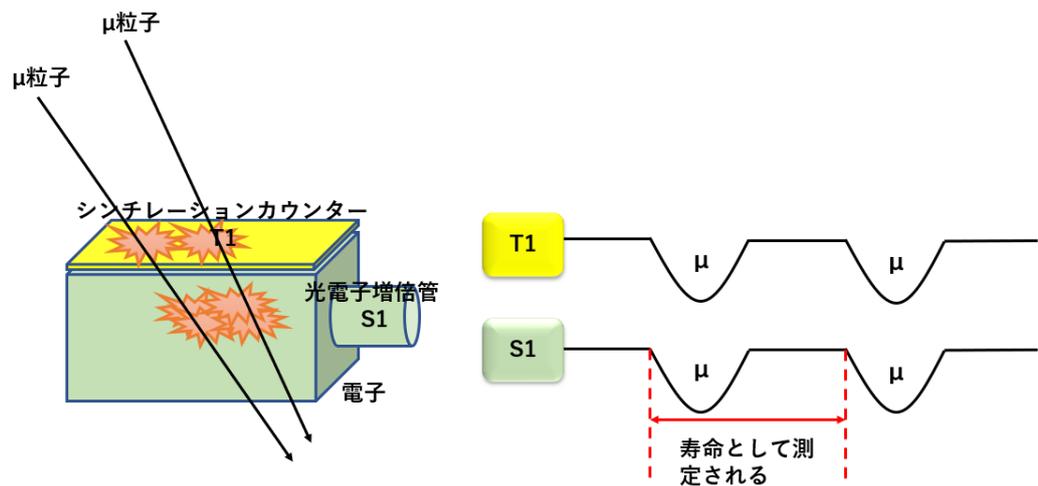
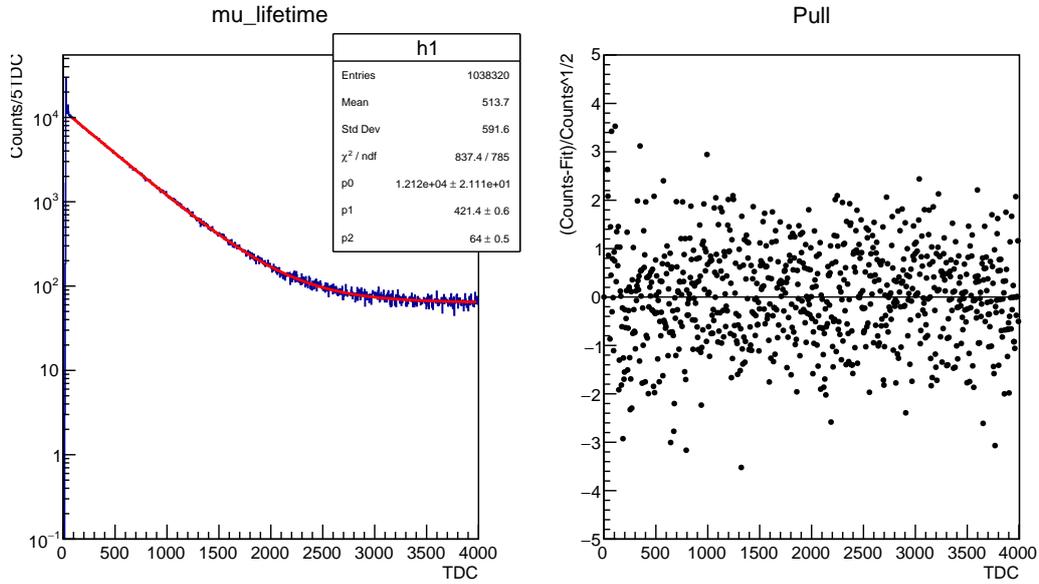


図 3.13 μ 粒子が同時に入射する事象のイメージ図

さらに、フィットの安定性を確認するために、データを $ADC(\mu)$ や $ADC(e)$ の領域ごとにサブサンプルに分けてフィットした結果を以下に示す。図 3.15 に分割した $ADC(\mu)$ の領域と $ADC(e)$ の領域を示す。各領域を 3 つに分けた。それぞれの

図 3.14 μ 粒子の TDC 分布

サブサンプルの領域を表 3.3 に示す。条件 1,2,3 は ADC(e) を 3 つに分けた結果であり、条件 4,5,6 は ADC(μ) を 3 つの領域に分けた結果である。表 3.3 の番号、1 ~ 6 に従って TDC 分布のグラフを図 3.16 ~ 図 3.21 に示す。

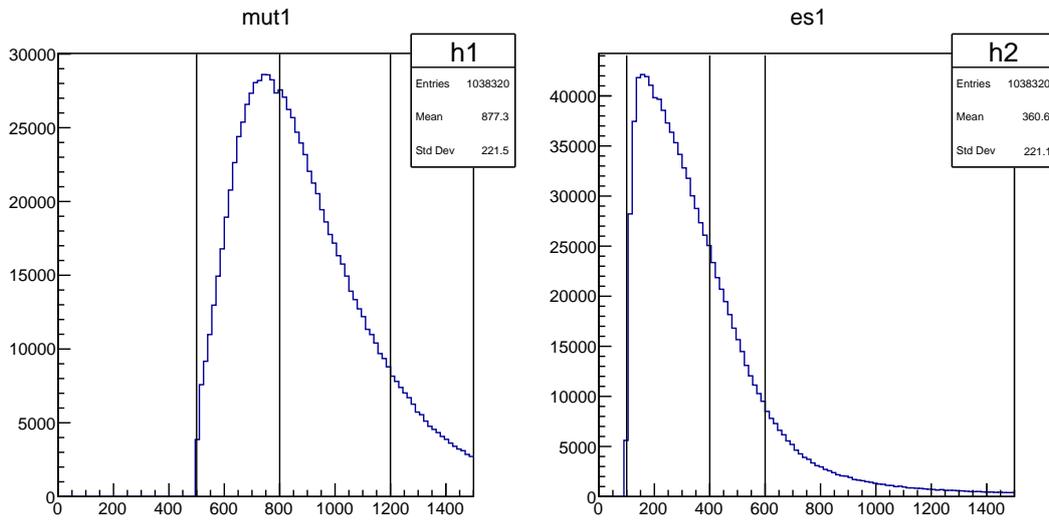
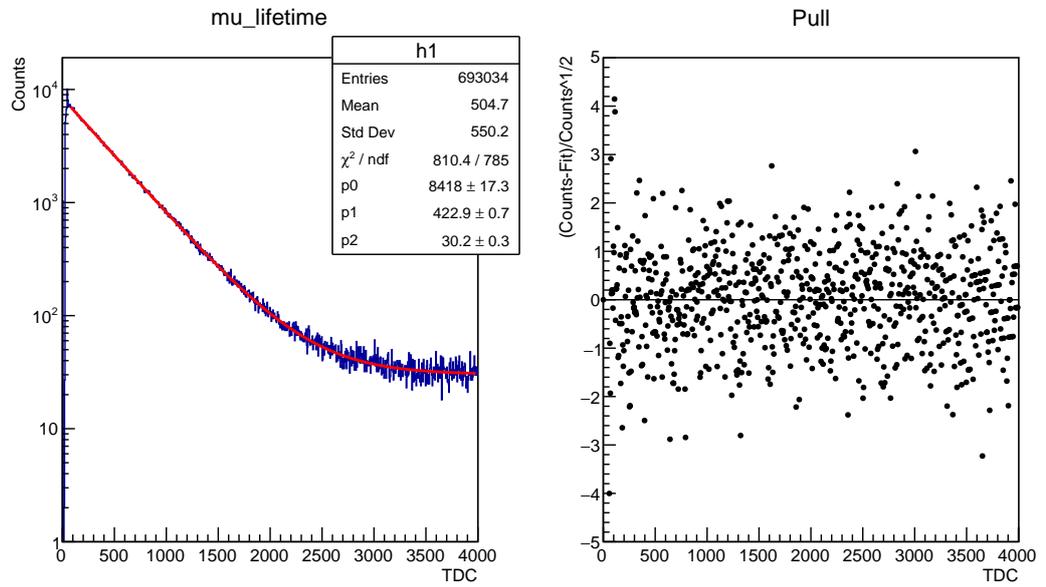


図 3.15 ADC のカット値

表 3.3 各条件での μ 粒子の平均寿命 (横軸 4000、bin 数 800、定数項あり)

条件	mut1	es1	χ^2/ndf	$\tau_\mu \pm \sigma_{\tau_\mu} [\mu s]$
1		$400 > es1 > 100$	1.032	2.115 ± 0.013
2	$mut1 > 500$	$600 > es1 > 400$	1.121	2.091 ± 0.014
3		$es1 > 600$	1.125	2.093 ± 0.016
4	$800 > mut1 > 500$		1.000	2.105 ± 0.013
5	$1200 > mut1 > 800$	$es1 > 100$	0.994	2.112 ± 0.013
6	$mut1 > 1200$		1.140	2.102 ± 0.014
	$mut1 > 500$	$es1 > 100$	1.067	2.107 ± 0.013

図 3.16 条件 1 ($mut1 > 500$ 、 $400 > es1 > 100$) の TDC 分布

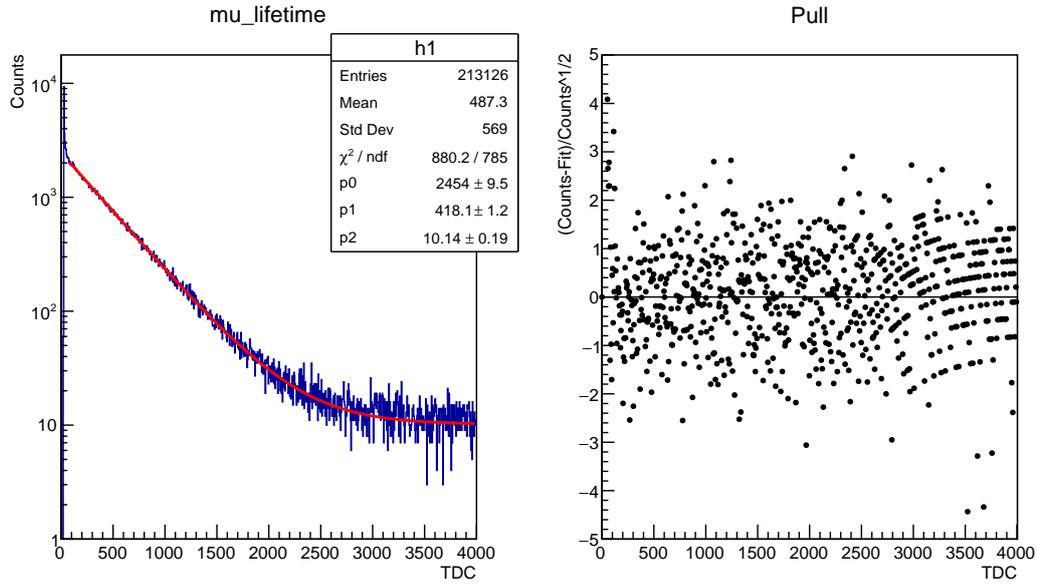


図 3.17 条件 2 ($\text{mut1} > 500$, $600 > \text{es1} > 400$) の TDC 分布

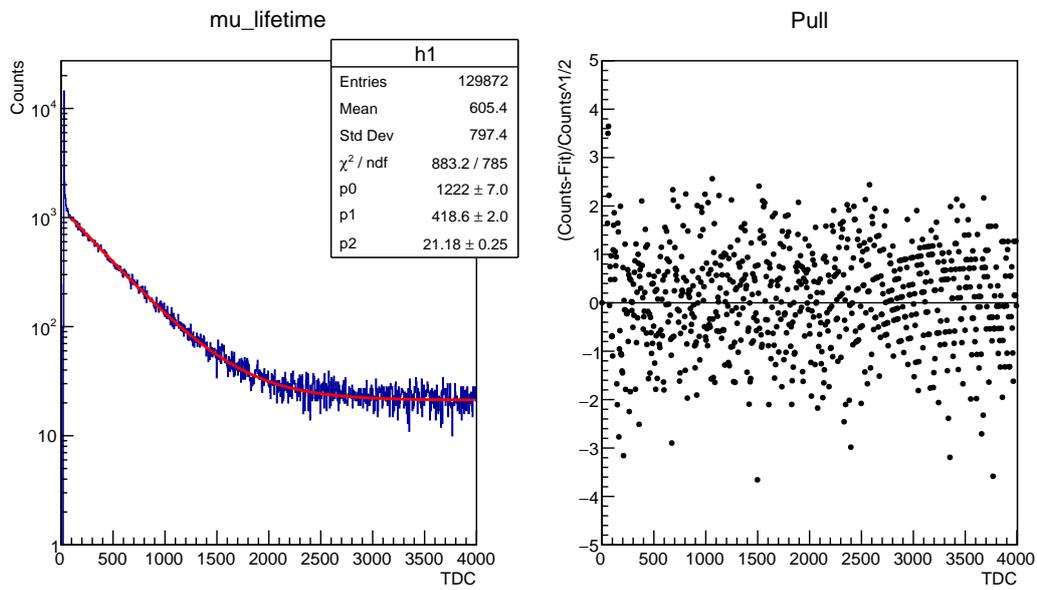


図 3.18 条件 3 ($\text{mut1} > 500$, $\text{es1} > 600$) の TDC 分布

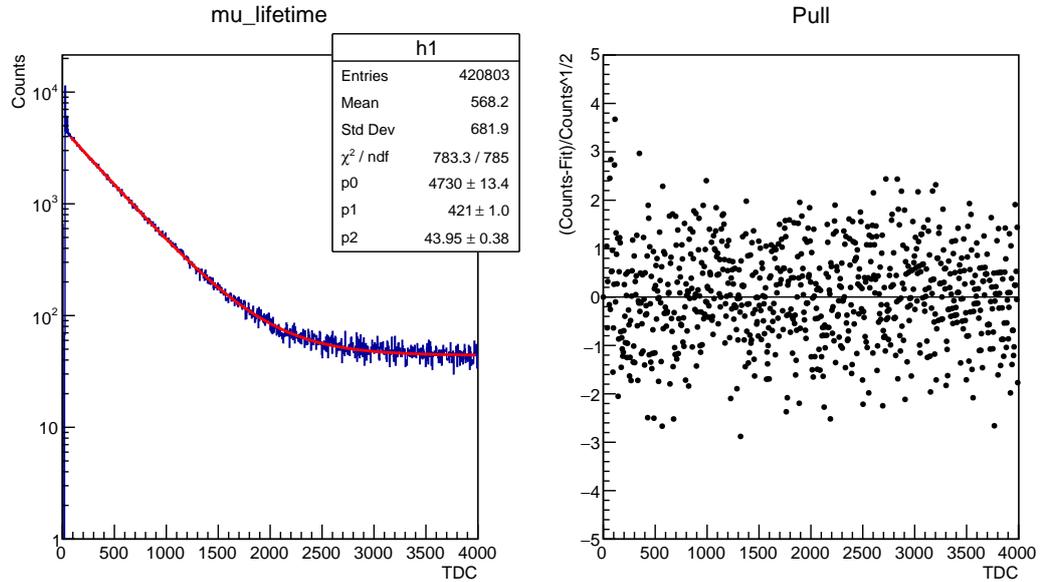


図 3.19 条件4 ($800 > \text{mut1} > 500$, $\text{es1} > 100$) の TDC 分布

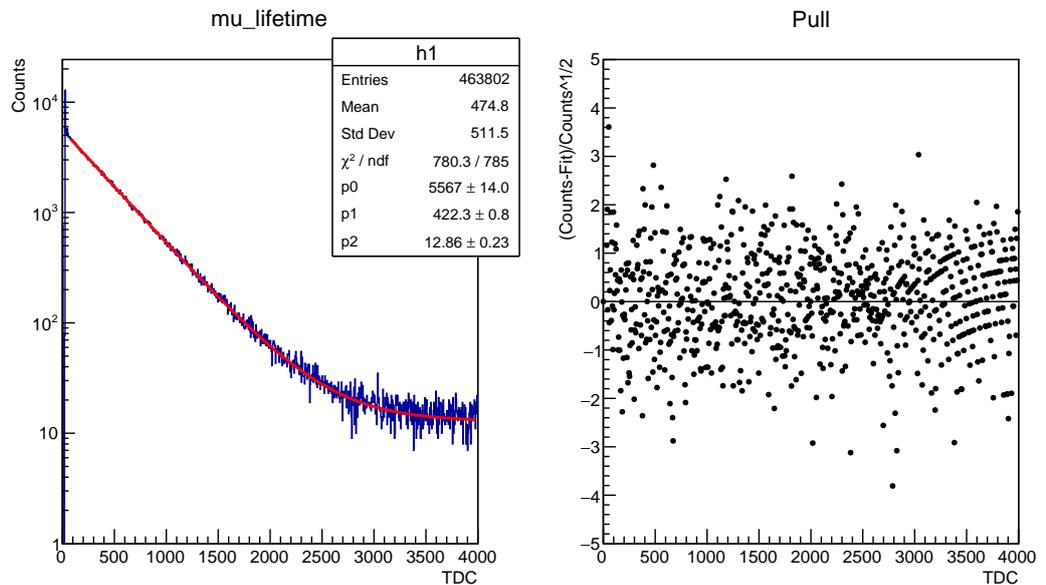


図 3.20 条件5 ($1200 > \text{mut1} > 800$, $\text{es1} > 100$) の TDC 分布

図 3.16～図 3.21 のグラフ及び Pull 分布、 χ^2 / ndf の値から分かるように、どの場合にもよくデータがフィットされている。求められた寿命の結果を表 3.3 にまとめるとともに図 3.22 示した。ここで、default とは最も緩いカット ($\text{mut1} > 500$ 、 $\text{es1} > 100$) の範囲でのフィットの結果である。図 3.22 から分かるように、どこを切り取っても結果が統計の範囲内で安定しているといえる。

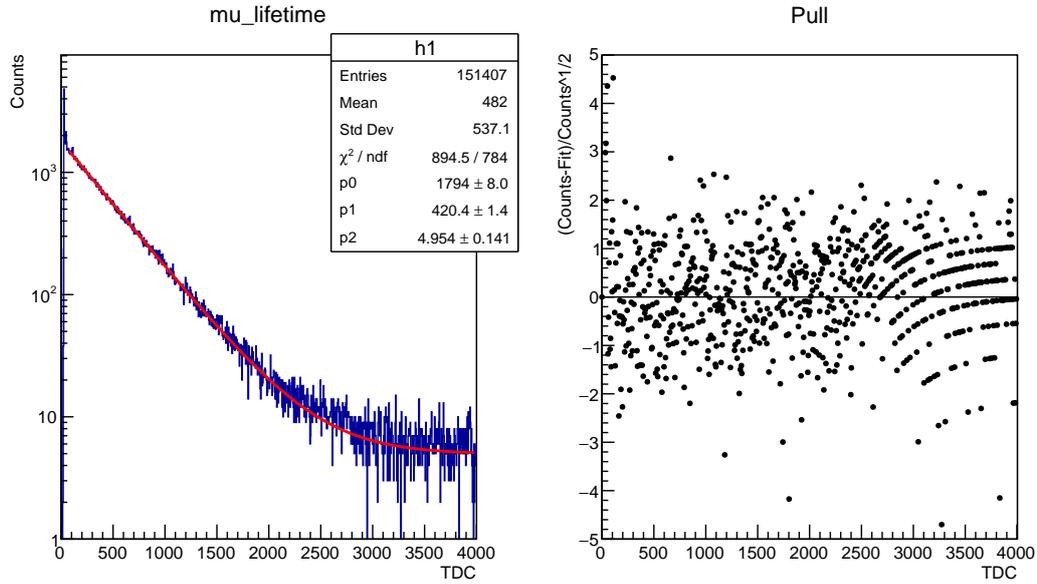


図 3.21 条件 6 ($\text{mut1} > 1200$, $\text{es1} > 100$) の TDC 分布

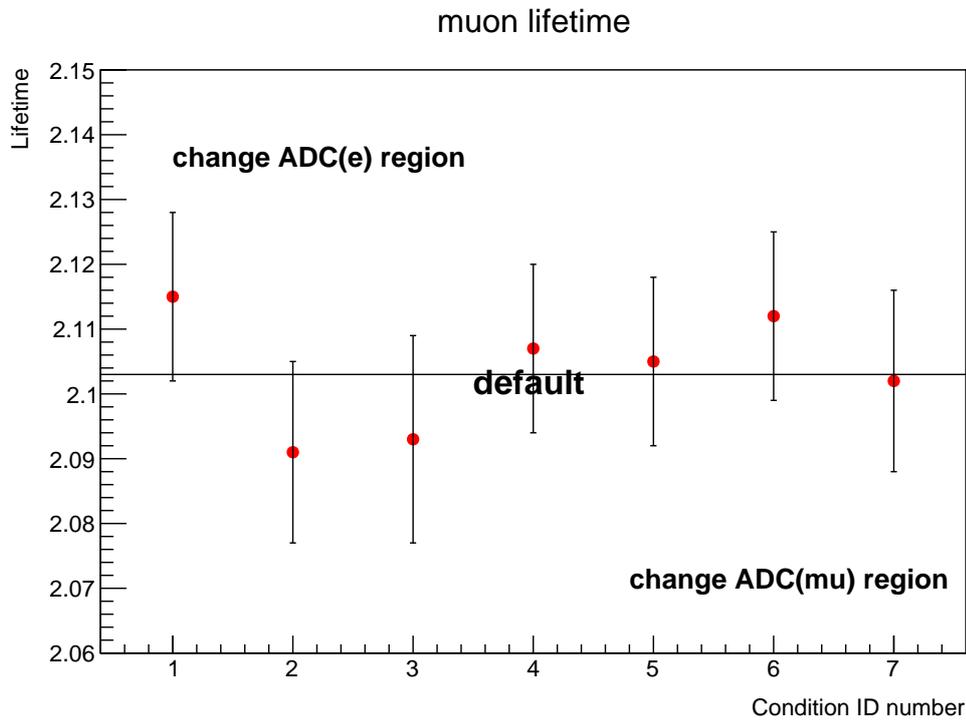


図 3.22 各サブサンプルから求めた μ 粒子の平均寿命

これらより、本実験での電荷区別なし ($\text{mut1} > 500$, $\text{es1} > 100$) での結果、

$$\tau_{\mu} = 2.107 \pm 0.013$$

を用いて考察することにする。

3.5.2 μ^- 粒子の平均寿命（電荷の区別あり）

本章では、負電荷の μ^- 粒子 (μ^-) と識別された 1,717 イベントをもとに平均寿命の測定について報告する。

まず、前小節と同様にまず定数項なしでプロットしてみる。TDC 分布の Fit し、寿命を求める。ここで、選定条件はすでに示した電荷区別なしでの $\text{mut1} > 500$ 、 $\text{es1} > 100$ と電荷区別ありでの $\text{mut3} > 300$ 、 $\text{mut4} > 200$ での図である。

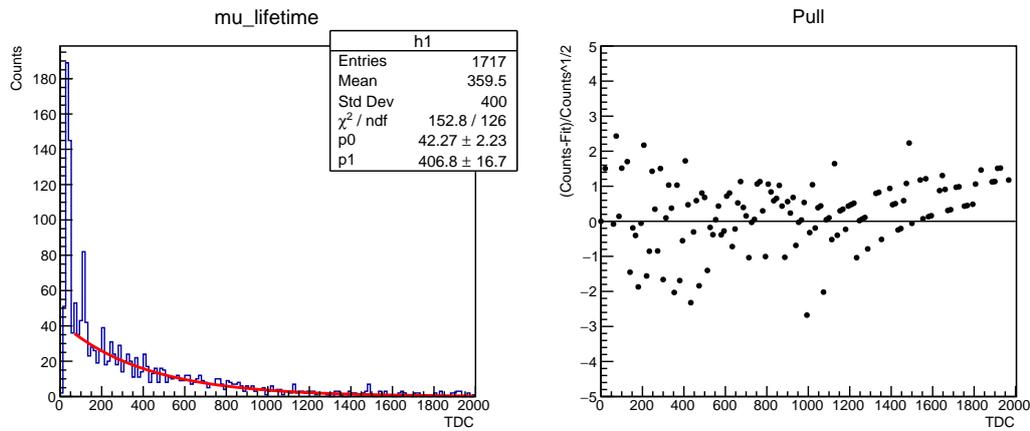


図 3.23 μ^- 粒子の TDC 分布

このグラフの χ^2/ndf を見てみると、

$$\chi^2/ndf = 152.8 \pm 126 = 1.2126$$

であり 1 に近い。フィットに問題はないと考えられる。

電荷区別ありでも前小節と同様に横軸を 4000、定数項を加えた Fit をしてみた。電荷区別ありではエントリーが非常に少ないので Loglikelihood のオプションをつかしてフィットした。グラフより、2000 以降ではほとんどエントリーがなく、定数項 p_2 の値も約 1.0 であるので、電荷区別ありの Fit では第 3.1 章で述べた式のままの Fit を採用し、考察でも TDC2000 までで定数項なしでの値について述べる。前小節と同様に、

$$\begin{aligned} \tau_{\mu^-}[ns] &= p_1[count] \times \alpha[ns/count] \\ &= 406.8 \times 5.00 \\ &= 2034.00[ns] \\ &= 2.034[\mu s] \end{aligned} \tag{3.14}$$

平均寿命 τ_{μ^-} の誤差 $\delta\tau_{\mu^-}$ は、

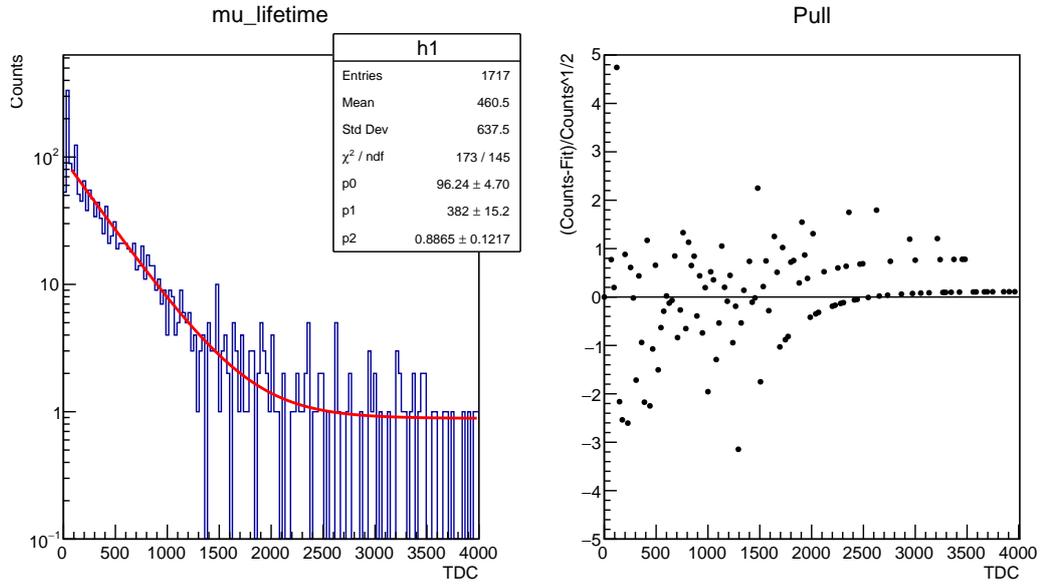


図 3.24 μ^- 粒子の TDC 分布 (横軸 4000、定数項あり、loglikelihood でのフィット)

$$\begin{aligned}
 \frac{\sigma_{\tau_{\mu^-}}}{\tau_{\mu^-}} &= \sqrt{\left(\frac{\delta\alpha}{\alpha}\right)^2 + \left(\frac{\delta p_1}{p_1}\right)^2} \\
 &= \sqrt{\left(\frac{16.7}{406.8}\right)^2 + \left(\frac{0.03}{5.00}\right)^2} \\
 &= 0.041488
 \end{aligned} \tag{3.15}$$

$$\therefore \sigma_{\tau_{\mu^-}} = 2.034 \times 0.041488 = 0.084[\mu s] \tag{3.16}$$

したがって、本実験で得られた結果から導きだされる μ^- 粒子の寿命 τ_{μ^-} は、

$$\tau_{\mu^-} = 2.03 \pm 0.08[\mu s]$$

となった。

3.6 考察

3.6.1 μ 粒子の寿命解析

本章では、本実験の結果について考察する。また、本実験で得られたのデータと一昨年からの先行実験によるものとの比較を行う。本実験では、先行実験と同じセットアップで行ったがテスト運転でのデータをふまえて、シンチレーターの性能が下がったと考えシンチレーターにかける HV の値の変更などを行った。また、HV の ch が減ったことから、メインシンチレーターの周りのシンチレーターは用いていない。データの収集方法は先行実験と同じであり、先行実験と同様のプログラムを使用した。このように先行実験とは実験方法などに変更点があり、先行実験とは選定条件が異なる。そこで、電荷区別あり・なしそれぞれにおいて、

1. 各実験の選定条件での解析結果
2. 先行実験で得られたデータを本実験の選定条件で選定し平均寿命を求めたもの

の両方を比較する。

3.6.2 電荷区別あり

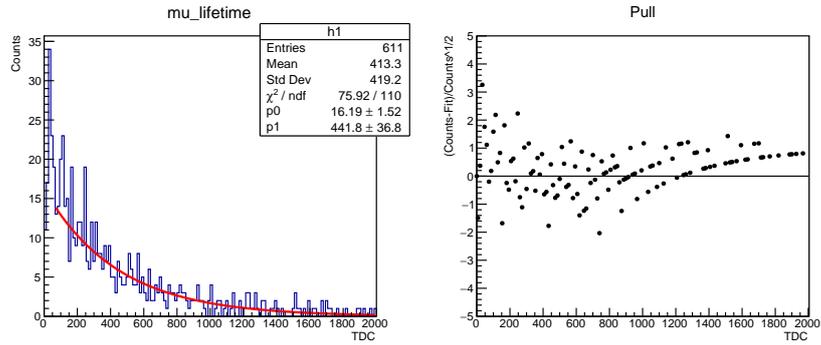
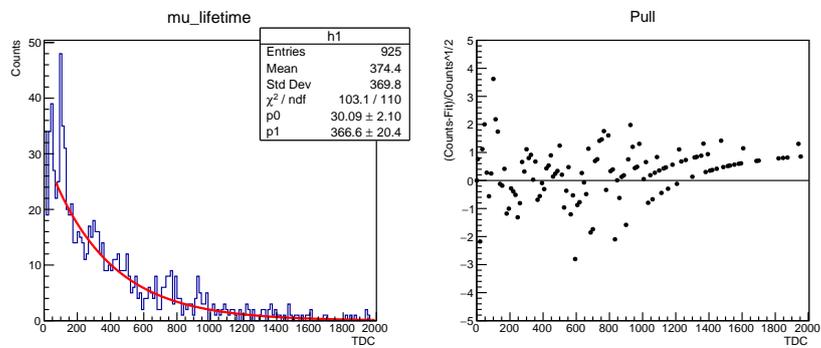
表 3.4 各選定条件での μ^- 粒子の平均寿命
この値は比較のみ。以降用いない

年度	測定時間 [h]	エントリー数	$\tau_{\mu} [\mu s]$
2018	336	860	2.16 ± 0.11
2019	1388	1253	2.04 ± 0.10
2020	2801	1,717	2.03 ± 0.08

表 3.5 本実験の条件での μ^- 粒子の平均寿命
以降、この値を用いる

年度	測定時間 [h]	エントリー数	$\tau_{\mu^-} [\mu s]$
2018	336	611	2.21 ± 0.18
2019	1388	925	1.83 ± 0.10
2020	2801	1717	2.03 ± 0.08
2018+2019+2020	4525	3253	1.97 ± 0.05
文献値			2.02 ± 0.02

どれも誤差の範囲内で文献値 (物質中の μ^- の寿命) とよく一致している。図 3.25～図 3.28 より、TDC 分布と Pull を比較していく。Pull のグラフでは曲線が見られるが、ヒ

図 3.25 μ^- 粒子の TDC 分布 (2018 年度)図 3.26 μ^- 粒子の TDC 分布 (2019 年度)

ストグラムの値が少なく、Fit 関数の影響をそのまま受けている。これは先行実験と同じであり、まだエントリーが少ないことが考えられる。多くのデータをとるためにはより時間が必要だが、時間で解決するのは最良ではないと考える。本実験では約 5 万エントリーをとるのに 1 時間かかる。その中でも崩壊している粒子はすくなく、さらに μ^- 粒子を選定すると 1 時間ではほとんど確認できない。よってさらにシンチレータを増やすなどのセットアップの改良が必要である。電荷区別するための磁石を増やすことや、磁石の上に置くシンチレータを購入・増やす必要がある。また、3 年分のデータの解析結果である $\tau_{\mu^-} = 1.97 \pm 0.05$ は、文献値とよく一致している。また、電荷区別ありでは、第 1.4.3 章で示したように、本実験で用いたプラスチックシンチレータの特性より、 μ^- 粒子の寿命の文献値は表 1.1 での C の Mean life の値を用いる。

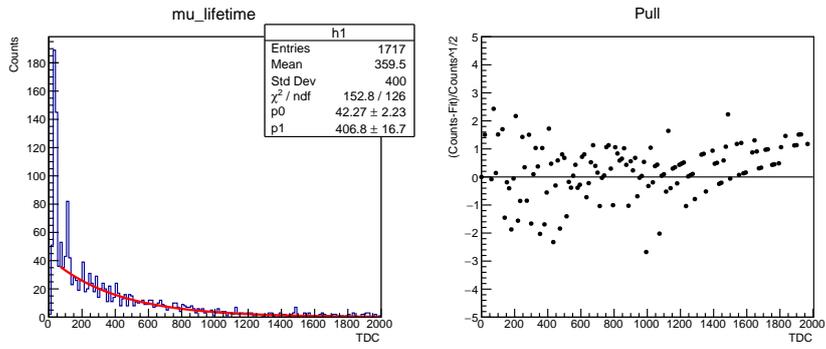


図 3.27 μ^- 粒子の TDC 分布 (2020 年度)

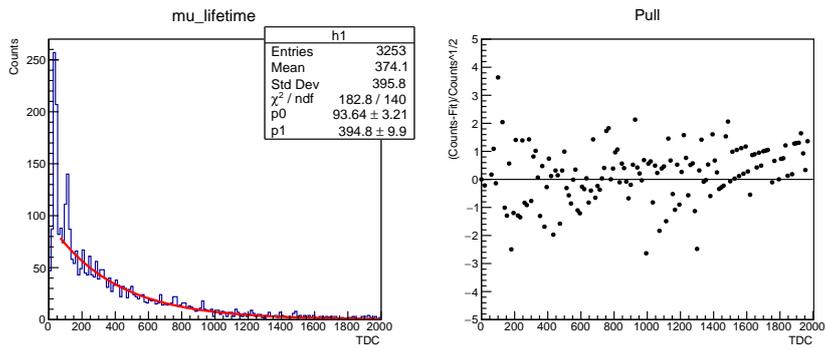


図 3.28 μ^- 粒子の TDC 分布 (2018+2019+2020 年度)

3.6.3 電荷区別なし

1. 各実験の選定条件での解析結果

まず、各実験の選定条件での結果を比較する。本実験の値は、先行実験と同じ横軸 2000 のまま、Fit 関数に定数項を加えていない式での結果を用いた。

表 3.6 各選定条件での μ 粒子の平均寿命
比較のみ、この値は以降用いない

年度	測定時間 [h]	エントリー数	$\tau_{\mu}[\mu s]$
2018	336	351,864	2.230±0.007
2019	1,388	458,336	2.221±0.006
2020	2,801	1,038,320	2.205± 0.014

2. 先行実験で得られたデータを本実験の選定条件で選定し平均寿命を求めたもの

次に、本実験での選定条件で先行実験のデータを選定し平均寿命を求めた物を比較する。先行実験でのデータに 2000 以上の TDC のイベントがほとんどなく、結果で用いた横軸 4000、定数項ありでの Fit ができないため次の条件で比較する。

- $mut1 > 500$
- $es1 > 100$
- $tdc2 < 4000$
- Fit 関数 : $y(t) = p_0 \exp(-\frac{t}{p_1})$
- TDC のレンジ:2000

である。この条件で先行実験のみ（2018 年度、2019 年度それぞれの値）と、本実験、また先行実験と本実験のデータ（2018 + 2019 + 2020 年度）によるものを以下にまとめ比較する。

表 3.7 本実験の選定条件での μ 粒子の平均寿命
まとめでは τ_{μ} とする

年度	測定時間 [h]	エントリー数	$\tau_{\mu}[\mu s]$
2018	336	233,515	2.210±0.014
2019	1,388	449,137	2.194±0.014
2020	2,801	1,038,320	2.205± 0.014
2018+2019+2020	4525	1,720,972	2.205±0.013

図 3.22～図 3.25 より、TDC 分布と Fit 関数、Pull を比較する。Pull は TDC 分布に対して Fit が上手く行われているかを確認するために用いた。そこで、図を見ると、ほとんど同じような形をしていることが分かる。これは先行実験同様寿命の異なる μ 粒子と μ^- 粒子が混在していることによると考えられる。また、本実

験でのデータと先行実験 (2018、2019) を比較してみると、本実験の χ^2/ndf の値が大きい事がわかる。さらに、定数項を入れ 4000 で Fit したグラフを見たところ、2018 年度と 2019 年度には TDC が 2000 までしかなく、2000 以降はほとんどエントリーが無いことが分かった。2000 でオーバーフローしていることが考えられるが、原因は不明である。先行実験と同じ方法で本実験の TDC 分布がうまく Fit 出来ていなかったのは 2000 以上の値が多くあったからである。よって、データを合わせたグラフでは本実験でのデータの影響から χ^2 の値が多くなっている事が分かる。

このことから TDC4000[カウント] で、Fit 関数に定数項ありでフィットしたものが最もデータを再現できている。その結果得られたものは、結果に示した $\tau_\mu = 2.1[\mu s]$ である。

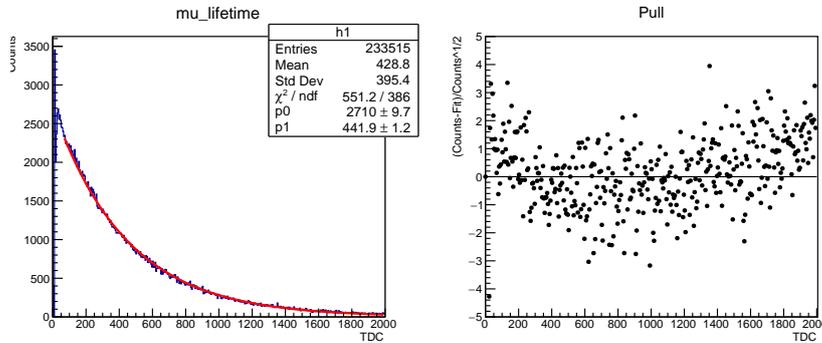


図 3.29 μ 粒子の TDC 分布 (2018 年度)

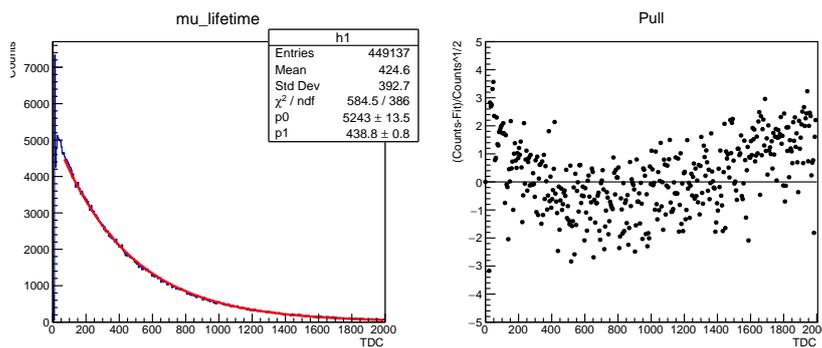
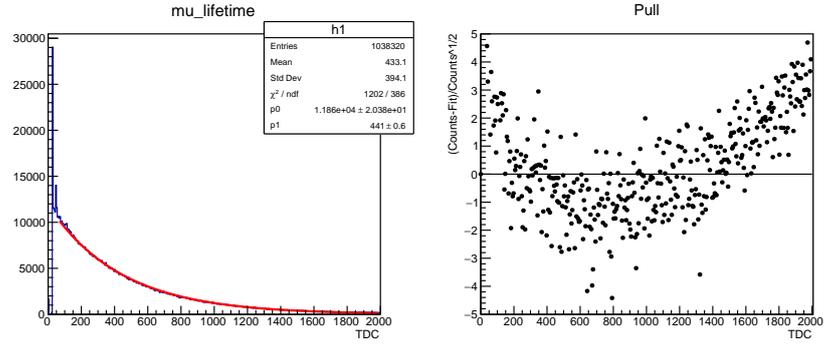
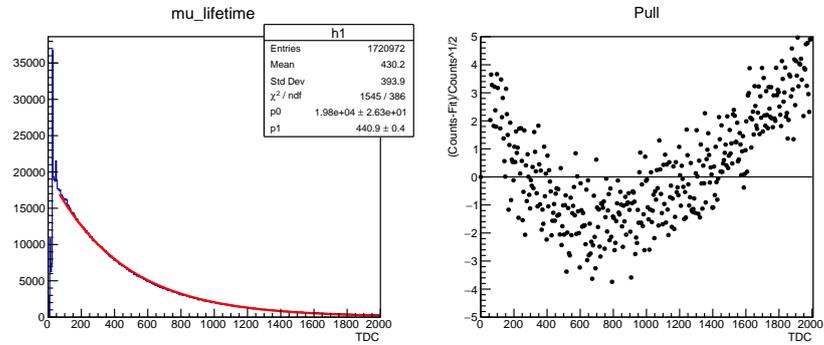


図 3.30 μ 粒子の TDC 分布 (2019 年度)

ここで、本実験で求めた電荷区別なしでの寿命を μ_{all} とすると、この μ_{all} には μ^+ と μ^- が同数に混じっていると仮定できる。すると、寿命に関して、

$$\tau_{\mu_{all}} = \frac{\tau_{\mu^+} + \tau_{\mu^-}}{2}$$

図 3.31 μ 粒子の TDC 分布 (2020 年度)図 3.32 μ 粒子の TDC 分布 (2018+2019+2020 年度)

が成り立つ。ここから、測定結果である

$$\tau_{\mu_{all}} = 2.107 \pm 0.013[\mu s]$$

$$\tau_{\mu^-} = 2.03 \pm 0.08[\mu s]$$

を用いて、 μ^+ の寿命を求める。

$$\begin{aligned} \tau_{\mu^+} &= 2\tau_{\mu_{all}} - \tau_{\mu^-} \\ &= 2 \times 2.107 - 2.103 \\ &= 2.184 \end{aligned} \quad (3.17)$$

$$\begin{aligned} \sigma_{\tau_{\mu^+}} &= \sqrt{(\sigma_{\tau_{\mu_{all}}})^2 + (\sigma_{\tau_{\mu^-}})^2} \\ &= 0.084119 = 0.08 \end{aligned} \quad (3.18)$$

$$\therefore \tau_{\mu^+} \pm \sigma_{\tau_{\mu^+}} = 2.18 \pm 0.08[\mu s]$$

この値を PDG に乗っている真空中の μ 粒子の寿命 $\tau_{\mu} = 2.197034 \pm 0.000021$ と比較すると、誤差の範囲で一致する。また、本実験では μ^- を区別して測定したが磁石の向きを反対にかえ μ^+ の寿命を測定すると、この値と一致すると考えられる。[1]

3.6.4 考察のまとめ

次に各イベント数と平均寿命をまとめた表を示す。表 3.8 の τ_{μ} は TDC を 2000 カウントまで、定数項なしで統一した先行実験を含む結果であり、 $\tau_{\mu_{all}}$ としているものが TDC4000 カウントまでの定数項ありの Fit 関数で求めた本実験の電荷区別なしでの最終的な結果である。

表 3.8 μ 粒子（電荷の区別あり、なし）の平均寿命

	測定時間 [h]	取得した全イベント数	$\tau_{\mu} [\mu s]$ (イベント数)	$\tau_{\mu_{all}} [\mu s]$ (イベント数)	$\tau_{\mu^{-}} [\mu s]$ (イベント数)
2020 年度	2,801	115,900,008	2.205 ± 0.014 (1,038,320)	2.107 ± 0.013	2.03 ± 0.08 (1,717)
2019 年度	1,388	507,761	2.194 ± 0.014 (449,137)		2.01 ± 0.13 (925)
2018 年度	336	386,521	2.210 ± 0.014 (233,515)		2.21 ± 0.18 (611)
2018~2020	4525	116,794,290	2.205 ± 0.013 (1,720,972)		1.97 ± 0.05 (3,252)
文献値					2.02 ± 0.02

第 4 章

まとめ

表 4.1 本実験の結果

測定時間 [h]	取得した全イベント数	$\tau_{\mu_{all}}$ [μs] (イベント数)	τ_{μ^-} [μs] (イベント数)	τ_{μ^+}
2,801	115,900,008	2.107 ± 0.013 (1,038,320)	1.97 ± 0.05 (1,717)	2.18 ± 0.08
文献値			2.02 ± 0.02	

表 4.1 にあるように μ^- 粒子の寿命 3 年間の蓄積されたデータの解析結果は 1.97 ± 0.05 となった。さらに、本実験での μ^- 粒子の平均寿命 τ_{μ^-} は $2.03 \pm 0.08 [\mu s]$ となった。この結果はどちらも文献値 $2.02 \pm 0.02 [\mu s]$ と誤差の範囲で一致した。さらに、電荷区別なしの結果は、本実験での測定値 $\tau_{\mu_{all}} = 2.107 \pm 0.013 [\mu s]$ と $\tau_{\mu^-} = 2.03 \pm 0.08 [\mu s]$ から求めた μ^+ の値 $\tau_{\mu^+} = 2.18 \pm 0.08 [\mu s]$ は真空中の μ 粒子の寿命 $2.197034 \pm 0.000021 [\mu s]$ [1] と誤差の範囲で文献値と一致し、文献値から求めた値とも誤差の範囲で一致した。

今後の課題として考えられる事を以下に示す。まず、昨年と同様にシンチレーションカウンタ T4 の波形が小さかったことなどから、光倍率が下がってきていることなどが考えられる。よって、新しいシンチレーションカウンターに入れ替えるなど、検討が必要であると考える。

さらに、今回もメインシンチレーターに入射した μ^- 粒子は 2 時間に約 1 イベントであったため先行実験と比較すると測定時間に対して取得したイベント数が少なかった。効率を高めるためには磁石の増設、シンチレーターの購入・増設などのセットアップの検討が必要だと考えられる。その他の実験機器については、実験の初めに HV の機器が使えなくなったため、チャンネル数を減らして測定した。現在は 8ch まで使えるが今後測定するならば新しい機器を購入する必要がある。

また、考察でも述べたように、磁場の向きを変更し μ^+ の寿命を測定してみると、本実験での測定値から求めた計算結果が正しいかが確認できる。

謝辞

卒業研究を進めるに当たり、ご指導やご協力頂いた皆様に深く感謝いたします。

特に、終始ご指導いただきました指導教官の林井先生に心より感謝申し上げます。お忙しい中、セットアップやプログラムについて、グラフの見方やデータの解析など様々な事にご指導、アドバイスいただきました。また、プログラムを書いて下さり改良して下さった蜂谷先生、的確なアドバイスやご指導、進捗状況を気にかけてくださりました宮林先生、下村先生、先輩方、大変お世話になりました。さらに、同回生の並本さん、西森さん、高濱さん、黒田さん、五屋さんにはいつも励ましていただき、モチベーションを保てました。卒業研究を進めることができたのは皆様のおかげです。感謝の意を持って謝辞に変えさせていただきます。

参考文献

- [1] K. Nakamura, *etal*, PDG, Journ. Phys. G, **37**,127 (2010).
- [2] 長嶋順清 「素粒子物理学の基礎 II」 朝倉書店 (1998).
- [3] 林井久樹 「素粒子の標準理論 (講義ノート)」
- [4] T.Suzuki and D.F. Measday, “Total nuclear capture rates for negative muons “
Phys Rev **C35**, 2217, June (1987).
- [5] John R.Taylor 林茂雄・馬場涼訳「計測における誤差解析入門」 東京化学同人
(2000).
- [6] 市川真有 永松えりな 皆吉遙 「 μ 粒子の寿命及び崩壊電子のエネルギースペクトラムの研究」 奈良女子大学理学部物理科学科 高エネルギー物理学研究室 卒業論文 (2016).
- [7] 石黒稚可子 藪内晶友美 「 μ 粒子の寿命及び放出電子のエネルギースペクトラム測定」 奈良女子大学理学部数物科学科物理学コース 高エネルギー物理学研究室 卒業論文 (2017).
- [8] 奥田真利子 「負電荷ミュー粒子と正電荷ミュー粒子の寿命測定」 奈良女子大学理学部数物科学科物理学コース 高エネルギー物理学研究室 卒業論文 (2018).
- [9] 下総七海 「 μ^- 粒子の寿命測定」 奈良女子大学理学部数物科学科物理学コース 高エネルギー物理学研究室 (2019).

付録 A

ファインマン図からの崩壊幅

付録 A に、標準理論に基づいた μ 粒子の寿命の計算をファインマン図を用いて計算する方法をまとめておく。できる限り計算の過程を飛ばさずに追うことを心がけた。

ファインマン図からの崩壊幅

粒子 A が B,C,D・・・へ崩壊する過程を考える。(今は便宜上 D までとする)

$$A \rightarrow B + C + D$$

崩壊幅 Γ は 1 個の粒子 A が単位時間に崩壊する確率を与える。したがって、微分崩壊率は、

$$\begin{aligned} \text{微分崩壊率} &= d\Gamma \\ &= \text{体積 } V \text{ 中で 1 個の粒子が単位時間に } d\mathbf{P}_B, d\mathbf{P}_C, d\mathbf{P}_D \text{ の運動量の領域に崩壊する確率} \\ &= \frac{P(\text{確率})}{T(\text{時間})} \\ &= \frac{|\langle f|S|i\rangle|^2}{2m \cdot T} \cdot \frac{d^3\mathbf{P}_B \cdot d^3\mathbf{P}_C \cdot d^3\mathbf{P}_D}{(2\pi\hbar)^3 2E_B \cdot (2\pi\hbar)^3 2E_C \cdot (2\pi\hbar)^3 2E_D} \end{aligned} \quad (\text{A.1})$$

となる。ここで、 $|\langle f|S|i\rangle|^2$ は体積 V 中で、時間 T の間に $2E_1 V$ 個の粒子 1 が n 個 $i=2,3,\dots,n-1$ の粒子に崩壊する確率を与える。 $|\langle f|S|i\rangle| = M_{fi}$ は遷移振幅である。上記の状態で $S = i\delta(P_A - P_B - P_C - P_D)$ とおいて考えると、

$$|\langle f|S|i\rangle|^2 \sim T(2\pi)^4 \delta(E_A - E_B - E_C - E_D) \delta^3(\mathbf{P}_A - \mathbf{P}_B - \mathbf{P}_C - \mathbf{P}_D) M_{fi}^2 \quad (\text{A.2})$$

となる。体積 V 中には、 $2E_i V$ 個の粒子 i が存在するので、1 個の粒子にするには、 $2E_i V$ で割る必要がある。

体積 V 中で、単位時間に 1 個の粒子が崩壊し、終状態 (f) にそれぞれ 1 個の i 番目の粒子が微小運動量素辺 $\prod_i d^3 p_i$ 中の運動量を持つ確率を粒子 1 の静止系 ($2E_i = 2m$) で考え

る。上記の式より、

$$\begin{aligned} d\Gamma &= \frac{T(2\pi)^4 \delta^4(p_1 - p_f) |M_{fi}|^2}{T2m} \prod_{i=2}^{n+1} \frac{d^3 p_i}{(2\pi)^3 2E_i} \\ &= \frac{(2\pi)^4 \delta^4(p_1 - p_f) |M_{fi}|^2}{2E_1} \prod_{i=2}^{n+1} \frac{d^3 p_i}{(2\pi)^3 2E_i} \end{aligned} \quad (\text{A.3})$$

よって、

$$d\Gamma = \frac{|M_{fi}|^2}{2m} dLips \quad (\text{A.4})$$

となる。ここで、位相因子 $dLips$ は、

$$dLips = (2\pi)^4 \int \prod_{i=1}^3 \frac{d^3 p_i}{(2\pi)^3 2E_i} \delta^4(p_i - p_f) \quad (\text{A.5})$$

となる。

次に μ^- 粒子の崩壊幅を考える。弱い相互作用はフェルミ理論、または一般化した V-A 理論でよく記述できることが知られている。ここで、 $\mu^- \rightarrow e^- + \nu_\mu + \bar{\nu}_e$ の崩壊幅を計算する。核粒子の 4 元運動量などを図に示す。ウィークボソン $M_w = 82\text{GeV}$ の質量は m_μ と比べて大きいので、W ボソンのプロパゲータは、

$$\frac{ig_{\mu\nu}}{q^2 + M_w^2} \sim \frac{ig_{\mu\nu}}{M_w^2} \quad (\text{A.6})$$

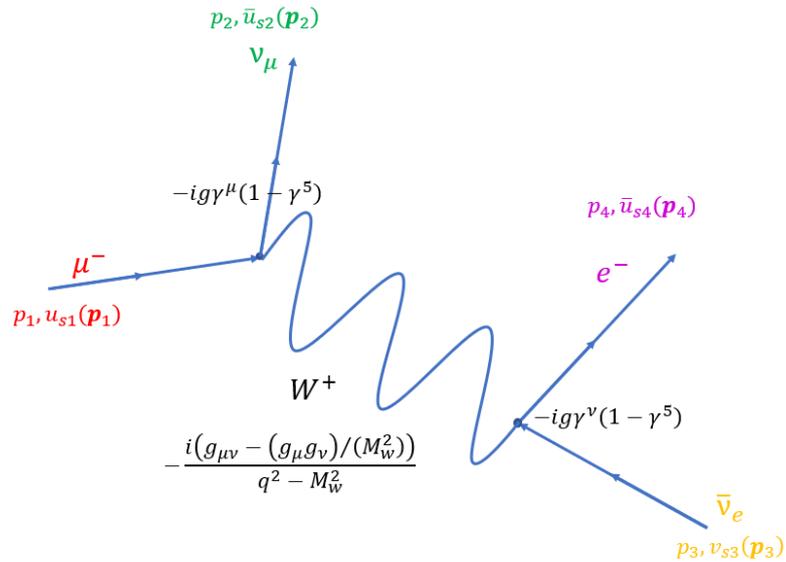


図 A.1 μ^- 粒子のファインマン図

のように近似できる。したがって、弱い相互作用の結合定数 g と M_w^2 はフェルミ定数 G_F で、

$$\frac{G_F}{\sqrt{2}} = \frac{g^2}{M_w^2}$$

で置き換えることができる。 M_{fi} はファインマン図から計算できるので、上記のファインマン図に対応する振幅 M は、

$$M = \frac{G_F}{\sqrt{2}} [\bar{u}_{s4}(\mathbf{p}_4) \gamma_\mu (1 - \gamma^5) u_{s3}(\mathbf{p}_3)] [\bar{u}_{s2}(\mathbf{p}_2) \gamma^\mu (1 - \gamma^5) u_{s1}(\mathbf{p}_1)] \quad (\text{A.7})$$

となる。ここで、

- 始状態のスピンは平均値をとる
- 終状態のスピンは確率で足し合わせる

である。

$$\bar{M} = \frac{1}{2} \sum_{s1} \sum_{s2, s3, s4} |M|^2 \quad (\text{A.8})$$

また、

$$J^\mu(p_2, p_1) \equiv \bar{u}_{s2}(\mathbf{p}_2) \gamma^\mu (1 - \gamma^5) u_{s1}(\mathbf{p}_1)$$

$$J_\mu(p_3, p_4) \equiv \bar{u}_{s4}(\mathbf{p}_4) \gamma_\mu (1 - \gamma^5) u_{s3}(\mathbf{p}_3)$$

とおく、すると、

$$M = \frac{G_F}{\sqrt{2}} J^\mu(p_2, p_1) J_\mu(p_3, p_4) \quad (\text{A.9})$$

である。なので \bar{M} は、

$$\begin{aligned} |\bar{M}|^2 &= \frac{1}{2} \sum_{s1} \sum_{s2, s3, s4} |M|^2 \\ &= \frac{1}{2} \sum_{s1} \sum_{s2, s3, s4} M^\dagger M \\ &= \frac{1}{2} \left(\frac{G_F}{\sqrt{2}} \right)^2 \sum_{s1} \sum_{s2, s3, s4} J^\nu(p_1, p_2) J_\nu(p_3, p_4) J^{\mu, \dagger}(p_1, p_2) J_\mu^\dagger(p_3, p_4) \\ &\equiv \left(\frac{G_F}{\sqrt{2}} \right)^2 \cdot 8 \cdot K^{\mu\nu}(p_1, p_2) K_{\mu\nu}(p_3, p_4) \end{aligned} \quad (\text{A.10})$$

ここで、テンソル $K^{\mu\nu}$ は、

$$K^{\mu\nu}(p_1, p_2) \equiv \frac{1}{4} \sum_{s1} \sum_{s2} J^{\dagger, \mu}(p_2, p_1) J^\nu(p_2, p_1) \quad (\text{A.11})$$

$$K_{\mu\nu}(p_3, p_4) \equiv \frac{1}{4} \sum_{s3} \sum_{s4} J_\mu^\dagger(p_3, p_4) J_\nu(p_3, p_4) \quad (\text{A.12})$$

ここで、 $K^{\mu\nu}(p_1, p_2)$ を計算してみると、

$$\begin{aligned}
K^{\mu\nu}(p_1, p_2) &= \frac{1}{4} \sum_{s_1} \sum_{s_2} [\bar{u}_{s_2}(\mathbf{p}_2) \gamma^\mu (1 - \gamma^5) u_{s_1}(\mathbf{p}_1)]^\dagger [\bar{u}_{s_2}(\mathbf{p}_2) \gamma^\nu (1 - \gamma^5) u_{s_1}(\mathbf{p}_1)] \\
&= \frac{1}{4} \sum_{s_1} \sum_{s_2} (\bar{u}_{s_1}(\mathbf{p}_1) \gamma^\mu (1 - \gamma^5) u_{s_2}(\mathbf{p}_2)) (u_{s_2}(\mathbf{p}_2) \gamma^\nu (1 - \gamma^5) u_{s_1}(\mathbf{p}_1)) \\
&\text{ここで、行列要素を考え、Tr の表現を用いると、} \\
&= \frac{1}{4} \text{Tr}[(\not{p}_2 + m_2) \gamma^\mu (1 - \gamma^5) (\not{p}_1 + m_1) \gamma^\nu (1 - \gamma^5)] \\
&= \frac{1}{4} \text{Tr}(\not{p}_2 \gamma^\mu (1 - \gamma^5) \not{p}_1 \gamma^\nu (1 - \gamma^5)) + \frac{1}{4} m_\mu \text{Tr}(\not{p}_2 \gamma^\mu (1 - \gamma^5) \gamma^\nu (1 - \gamma^5))
\end{aligned} \tag{A.13}$$

ここで、 $m_2 = m_\nu = 0, m_1 = m_\mu$ とし、前の項と後ろの項をそれぞれ考えると、後ろの項は 0 となるので、

$$\begin{aligned}
\text{前の項} &= \frac{1}{4} \text{Tr}(\not{p}_2 \gamma^\mu (1 - \gamma^5) \not{p}_1 \gamma^\nu (1 - \gamma^5)) \\
&= \frac{1}{4} \text{Tr}(p_{2\alpha} \gamma^\alpha \gamma^\mu (1 - \gamma^5) p_{1\beta} \gamma^\beta \gamma^\nu (1 - \gamma^5)) \\
&= \frac{1}{4} [2p_{2\alpha} p_{1\beta} \text{Tr}(\gamma^\alpha \gamma^\mu \gamma^\beta \gamma^\nu (1 - \gamma^5))] \\
&= \frac{1}{4} [2\text{Tr}(\not{p}_2 \gamma^\mu \not{p}_1 \gamma^\nu) - 2\text{Tr}(\not{p}_2 \gamma^\mu \not{p}_1 \gamma^\nu \gamma^5)] \\
&= \frac{1}{4} [8(p_2^\mu p_1^\nu + p_2^\nu p_1^\mu - (p_1 \cdot p_2) g^{\mu\nu}) + 8ip_{2\alpha} p_{1\beta} \varepsilon^{\alpha\mu\beta\nu}] \\
&= 2(p_2^\mu p_1^\nu + p_2^\nu p_1^\mu - (p_1 \cdot p_2) g^{\mu\nu}) + 2ip_{2\alpha} p_{1\beta} \varepsilon^{\alpha\mu\beta\nu}
\end{aligned} \tag{A.14}$$

よって、

$$\begin{aligned}
P^{\mu\nu}(p_1, p_2) &= 2(p_1^\mu p_2^\nu + p_2^\mu p_1^\nu - (p_1 \cdot p_2) g^{\mu\nu}) \\
Q^{\mu\nu}(p_1, p_2) &= 2ip_{2\alpha} p_{1\beta} \varepsilon^{\alpha\mu\beta\nu}
\end{aligned}$$

を用いると、

$$K^{\mu\nu}(p_1, p_2) = P^{\mu\nu}(p_1, p_2) + Q^{\mu\nu}(p_1, p_2) \tag{A.15}$$

となる。したがって、

$$\begin{aligned}
|\bar{M}|^2 &= 8 \left(\frac{G_F}{\sqrt{2}} \right)^2 (K^{\mu\nu}(p_1, p_2) K_{\mu\nu}(p_3, p_4)) \\
&= 8 \left(\frac{G_F}{\sqrt{2}} \right)^2 (P^{\mu\nu}(p_1, p_2) + Q^{\mu\nu}(p_1, p_2))(P^{\mu\nu}(p_3, p_4) + Q^{\mu\nu}(p_3, p_4))
\end{aligned} \tag{A.16}$$

ここで、上記の式の 4 つの内積を順に考える。

- $P^{\mu\nu}(p_1, p_2)P_{\mu\nu}(p_3, p_4)$ の内積

$$\begin{aligned}
P^{\mu\nu}(p_1, p_2)P_{\mu\nu}(p_3, p_4) &= 4(p_1^\mu p_2^\nu + p_2^\mu p_1^\nu - (p_1 \cdot p_2)g^{\mu\nu}) \\
&\quad (p_{3\mu} p_{4\nu} + p_{4\mu} p_{3\nu} - (p_3 \cdot p_4)g_{\mu\nu}) \\
&= 4(2(p_1 \cdot p_3)(p_2 \cdot p_4) + 2(p_1 \cdot p_4)(p_2 \cdot p_3) \\
&\quad - 4(p_3 \cdot p_4)(p_1 \cdot p_2) + (p_1 \cdot p_2)(p_3 \cdot p_4)g^{\mu\nu}g_{\mu\nu}) \\
&= 8(p_1 \cdot p_3)(p_2 \cdot p_4) + 8(p_1 \cdot p_4)(p_2 \cdot p_3) \\
&= 8((p_1 \cdot p_3)(p_2 \cdot p_4) + (p_1 \cdot p_4)(p_2 \cdot p_3))
\end{aligned} \tag{A.17}$$

- $Q^{\mu\nu}(p_1, p_2)Q_{\mu\nu}(p_3, p_4)$ の内積

$$\begin{aligned}
Q^{\mu\nu}(p_1, p_2)Q_{\mu\nu}(p_3, p_4) &= (2i)^2 p_{2\alpha} p_{1\beta} \varepsilon^{\alpha\mu\beta\nu} \times p_3^\sigma p_4^{rho} \varepsilon_{\sigma\mu\rho\nu} \\
&= -4p_{2\alpha} p_{1\beta} p_3^\sigma p_4^\rho \varepsilon^{\alpha\mu\beta\nu} \varepsilon_{\sigma\mu\rho\nu} \\
&= -8(p_{2\alpha} p_{1\beta} p_3^\sigma p_4^\rho (\delta_\sigma^\alpha \delta_\rho^\beta - \delta_\rho^\alpha \delta_\sigma^\beta)) \\
&= -8(p_{2\sigma} p_{1\rho} p_3^\sigma p_4^\rho - p_{2\rho} p_{1\sigma} p_3^\sigma p_4^\rho) \\
&= -8((p_2 \cdot p_3)(p_1 \cdot p_4) - (p_2 \cdot p_4)(p_1 \cdot p_3)) \\
&= 8((p_2 \cdot p_4)(p_1 \cdot p_3) - (p_2 \cdot p_3)(p_1 \cdot p_4))
\end{aligned} \tag{A.18}$$

- $P^{\mu\nu}(p_1, p_2)Q_{\mu\nu}(p_3, p_4)$ の内積

$$\begin{aligned}
P^{\mu\nu}(p_1, p_2)Q_{\mu\nu}(p_3, p_4) &= 2(p_1^\mu p_2^\nu + p_2^\mu p_1^\nu - (p_1 \cdot p_2)g^{\mu\nu}) (2ip_3^\alpha p_4^\beta \varepsilon^{\alpha\mu\beta\nu}) \\
&\quad \mu\nu \text{ の入れ替えなので、} \varepsilon \text{ が符号を変えて全体が } 0 \text{ となるので} \\
&= 0
\end{aligned} \tag{A.19}$$

以上より、

$$\begin{aligned}
|\bar{M}|^2 &= 8 \left(\frac{G_F}{\sqrt{2}} \right)^2 \times 16(p_1 \cdot p_3)(p_2 \cdot p_4) \\
&= 64G_F^2 (p_1 \cdot p_3)(p_2 \cdot p_4)
\end{aligned} \tag{A.20}$$

よって、

$$\begin{aligned}
d\Gamma &= \frac{|\bar{M}|^2}{2m_\mu} dLips \\
&= \frac{64G_F^2 (p_1 \cdot p_3)(p_2 \cdot p_4)(2\pi)^4}{2m_\mu} \cdot \frac{d^3 p_2 d^3 p_3 d^3 p_4}{(2\pi)^9 E_2 E_3 E_4} \delta^4(p_1 - p_2 - p_3 - p_4)
\end{aligned} \tag{A.21}$$

ここで、ニュートリノ (2・3番) は質量0で検出されない。そこで、ニュートリノの位相空間の積分を行う。 $S^{\mu\nu}$ を次のように定義する。

$$S^{\mu\nu} \equiv \int p_3^\mu p_2^\nu \delta^4(p_1 - p_2 - p_3 - p_4) \frac{d^3 p_2}{2E_2} \frac{d^3 p_3}{2E_3} \tag{A.22}$$

以降 μ 粒子の重心系で考え、 $p_1 = (m_\mu, \mathbf{0})$ である。後の計算で次のように定義する Δ^μ を導入する。

$$\Delta^\mu \equiv p_1^\mu - p_4^\mu = p_2^\mu - p_3^\mu \quad (\text{A.23})$$

$$\Delta^2 \equiv \Delta_\mu \Delta^\mu \quad (\text{A.24})$$

よって、

$$\begin{aligned} S^{\mu\nu} &\equiv \int p_3^\mu p_2^\nu \delta^4(p_1 - p_2 - p_3 - p_4) \frac{d^3 p_2}{2E_2} \frac{d^3 p_3}{2E_3} \\ &= \frac{\pi}{24} [g^{\mu\nu} \Delta^2 + 2\Delta^\mu \Delta^\nu] \end{aligned} \quad (\text{A.25})$$

この証明は次の小節に示す。これを (A.21) に代入すると、

$$\begin{aligned} d\Gamma &= \frac{|\bar{M}|^2}{2m_\mu} dLips \\ &= \frac{64G_F^2 (p_1 \cdot p_3)(p_2 \cdot p_4)(2\pi)^4}{2m_\mu} \cdot \frac{d^3 p_2 d^3 p_3 d^3 p_4}{(2\pi)^9 E_2 E_3 E_4} \delta^4(p_1 - p_2 - p_3 - p_4) \\ &= \frac{64G_F^2 p_{1\mu} S^{\mu\nu} p_{4\nu} (2\pi)^4 d^3 p_4}{2m_\mu (2\pi)^9 E_4} \\ &= \frac{64\pi G_F^2}{2m_\mu 24(2\pi)^5} \frac{d^3 p_4}{2E_4} ((p_1 \cdot p_4)\Delta^2 + 2(p_{1\mu} \cdot \Delta^\mu)(p_{4\nu} \cdot \Delta^\nu)) \\ &= \frac{1}{48\pi^4} \frac{G_F^2}{m_\mu} ((p_1 \cdot p_4)\Delta^2 + 2(p_1 \cdot \Delta)(p_4 \cdot \Delta)) \frac{d^3 p_4}{E_4} \end{aligned} \quad (\text{A.26})$$

いま、 μ 粒子の静止系で考えるので、

$$p_1^\mu = (m_\mu, \mathbf{0})$$

$$p_4^\mu = (E_e, \mathbf{P}_e)$$

また、電子の質量を無視すると $E_e = |\mathbf{P}_e|$ 、 $x \equiv \frac{E_e}{m_\mu/2}$ とおくと、

$$\Delta^\mu = p_1^\mu - p_4^\mu = (\Delta^0, \mathbf{\Delta})$$

$$\Delta^0 = m_\mu - E_e$$

$$\mathbf{\Delta} = \Delta^i = -\mathbf{P}_e$$

$$\begin{aligned} \Delta^2 &= \Delta^{0^2} - \mathbf{\Delta}^2 \\ &= (m_\mu - E_e)^2 - (\mathbf{P}_e)^2 \\ &= (m_\mu - E_e)^2 - E_e^2 \\ &= m_\mu^2(1 - x) \end{aligned} \quad (\text{A.27})$$

となり、次のように表すことができる。

$$p_1 \cdot p_4 = m_\mu E_e$$

$$p_1 \cdot \Delta = m_\mu(m_\mu - E_e) = m_\mu^2 - m_\mu E_e$$

$$\begin{aligned}
p_4 \cdot \Delta &= E_e(m_\mu - E_e) - (\mathbf{P}_e)(-\mathbf{P}_e) \\
&= m_\mu E_e - E_e^2 + \mathbf{P}_e^2 \\
&= m_\mu E_e
\end{aligned}$$

これらを代入して、

$$\begin{aligned}
d\Gamma &= \int \frac{1}{48\pi^4} \frac{G_F^2}{m_\mu} \left[(m_\mu E_e) \left(m_\mu^2 \left(1 - \frac{2E_e}{m_\mu} \right) \right) + 2(m_\mu^2 - m_\mu E_e)(m_\mu E_e) \right] \frac{P^2 dP(4\pi)}{E_e} \\
P &= E_e \text{とおくと} \\
&= \frac{G_F^2(4\pi)}{48m_\mu\pi^4} [m_\mu^3 E_e - m_\mu^2(2E_e^2) + 2m_\mu^3 E_e - 2m_\mu^2 E_e^2] \frac{E_e dE_e}{E_e} \\
&= \frac{G_F^2 m_\mu^5}{96\pi^3} x^2(3-2x)dx, \left(x = \frac{E_e}{m_\mu/2}, 0 \leq x \leq 1 \right)
\end{aligned} \tag{A.28}$$

という電子エネルギースペクトルを得る。さらに、エネルギーについて積分すると全崩壊率が得られる。

$$\Gamma = \frac{G_F^2 m_\mu^5}{192\pi^3} \tag{A.29}$$

また、電子の質量を無視しない場合には、1.3.3の式で示すように関数 $F(x)$ で表される項が現れる。

A.1 3体崩壊の $S^{\mu\nu}$ の計算

この章では次の

$$S^{\mu\nu} \equiv \int p_3^\mu p_2^\nu \delta^4(p_1 - p_2 - p_3 - p_4) \frac{d^3 p_2}{2E_2} \frac{d^3 p_3}{2E_3} \tag{A.30}$$

の具体的な計算方法について解説し、これが式 (A.25) で用いたように、

$$S^{\mu\nu} = \frac{\pi}{24} [g^{\mu\nu} \Delta^2 + 2\Delta^\mu \Delta^\nu] \tag{A.31}$$

となることを示す。ここで、4元ベクトル Δ^μ は

$$\Delta^\mu \equiv p_1^\mu - p_4^\mu = p_2^\mu + p_3^\mu \tag{A.32}$$

$$\Delta^2 \equiv \Delta_\mu \Delta^\mu \tag{A.33}$$

と定義された量で、4元ベクトルの各成分は粒子1と4の変数のみで決まっている。

具体的に μ 粒子の重心系で考えると $p_1 = (m_\mu, \mathbf{0})$ でなので、

$$\Delta^\mu \equiv (\Delta^0, \mathbf{\Delta}) \tag{A.34}$$

$$\Delta^0 = m_\mu - E_4 = E_2 + E_3 \tag{A.34}$$

$$\mathbf{\Delta} = -\mathbf{p}_4 = \mathbf{p}_2 + \mathbf{p}_3 \tag{A.35}$$

$$\Delta^2 = \Delta^\mu \Delta_\mu = m_\mu^2 - 2m_\mu E_4 \tag{A.36}$$

である。

まず、運動量保存を使って、 d^3p_3 の積分を実行する。

$$\begin{aligned} S^{\mu\nu} &= \int p_3^\mu p_2^\nu \delta(m_\mu - E_2 - E_3 - E_4) \delta^3(\mathbf{p}_2 + \mathbf{p}_3 + \mathbf{p}_4) \frac{d^3\mathbf{p}_2}{2E_2} \frac{d^3\mathbf{p}_3}{2E_3} \\ &= \int p_3^\mu p_2^\nu \delta(m_\mu - E_2 - E_3 - E_4) \frac{d^3p_2}{4E_2E_3} \end{aligned} \quad (\text{A.37})$$

$$= \int p_3^\mu p_2^\nu \delta(\Delta^0 - E_2(\rho) - E_3(\rho)) \frac{\rho^2 d\rho d\cos\theta d\phi}{4E_2E_3} \quad (\text{A.38})$$

ここで、表記を簡単にするために粒子2の運動量の絶対値を ρ とおいた ($\rho = |\mathbf{p}_2|$)。また、運動量保存に関するデルタ関数の積分を実行する過程で、 E_2, E_3 が ρ の関数になっていることを明確にするために E_2 と E_3 が ρ の関数であることを陽に示した。以下のニュートリノの質量はゼロとおく。

$$E_2(\rho) = \sqrt{m_{\nu_2}^2 + \rho^2} = \rho \quad (\text{A.39})$$

$$\begin{aligned} E_3(\rho) &= \sqrt{(\Delta - \mathbf{p}_2)^2} \\ &= \sqrt{\Delta^2 - 2|\Delta|\rho\cos\theta + \rho^2} \end{aligned} \quad (\text{A.40})$$

$d\rho$ に関する積分を実行するために、式中のデルタ関数の中身を $f(\rho)$ と定義する。

$$f(\rho) = \Delta^0 - E_2(\rho) - E_3(\rho)$$

この微分は

$$\begin{aligned} \left| \frac{\partial f(\rho)}{\partial \rho} \right| &= \left| -1 - \frac{\rho - |\Delta|\cos\theta}{E_3} \right| \\ &= \left| \frac{-E_3 - \rho + |\Delta|\cos\theta}{E_3} \right| \end{aligned} \quad (\text{A.41})$$

また、 $f(\rho) = 0$ を満たす ρ を ρ_0 とおく。すなわち

$$E_3 = \Delta^0 - \rho_0 \quad (\text{A.42})$$

である。 ρ_0 と E_3 は Δ および $x \equiv \cos\theta$ を用いて、

$$\rho_0 = \frac{(\Delta^0)^2 - \Delta^2}{2(\Delta^0 - |\Delta|x)} \quad (\text{A.43})$$

$$E_3 = \frac{(\Delta^0)^2 + |\Delta|^2 - 2\Delta^0|\Delta|x}{2(\Delta^0 - |\Delta|x)} \quad (\text{A.44})$$

と表すことができる。ここで、 $E_3 > 0$, $\rho_0 > 0$ であることに注意すること。

以上の結果を代入する。

$$\begin{aligned} S^{\mu\nu} &= \frac{1}{4} \int p_3^\mu p_2^\nu \frac{\delta(\rho - \rho_0)}{\left| \frac{\partial f}{\partial \rho} \right|} \frac{\rho^2 d\rho d\cos\theta d\phi}{E_2E_3} \\ &= \frac{1}{4} \int p_3^\mu p_2^\nu \frac{\rho_0}{||\Delta|x - E_3 - \rho_0|} dx d\phi \end{aligned}$$

2番目の行にいくために $d\rho$ で積分し、 $x = \cos\theta$ と置いた。以下に示すように、分母が絶対値であることが重要である。式を用いて分母を書き換えると、

$$\begin{aligned} ||\Delta|x - E_3 - \rho_0| &= \left| \frac{-2(\Delta^0)^2 + 4|\Delta|\Delta^0 - 2|\Delta|^2x^2}{2(\Delta^0 - |\Delta|x)} \right| \\ &= \left| \frac{-(\Delta^0 - |\Delta|x)^2}{\Delta^0 - |\Delta|x} \right| \\ &= |\Delta^0 - |\Delta|x| = \Delta^0 - |\Delta|x \end{aligned} \quad (\text{A.45})$$

となる。途中でマイナスの符号が出てくるが、絶対値を取っているためにこの符号は消える。最後に絶対値の記号をなくしたのは $\Delta^0 - |\Delta|x$ が全ての $x: -1 \rightarrow +1$ に対して、常に正 $>$ であるからである。

以上を代入することにより

$$S^{\mu\nu} = +\frac{1}{8} \int_{-1}^1 dx \int_0^{2\pi} d\phi p_3^\mu p_2^\nu \frac{(\Delta^0)^2 - \Delta^2}{(\Delta^0 - |\Delta|x)^2} \quad (\text{A.46})$$

A.1.1 S^{00} の場合

まず、 $\mu = 0, \nu = 0$ の場合について考える。この場合は $p_2^0 = \rho_0, p_3^0 = E_3$ なので、それらを代入することにより、以下のように変形できる。

$$\begin{aligned} S^{00} &= \frac{1}{8} \int_{-1}^1 dx \int_0^{2\pi} d\phi p_3^0 p_2^0 \frac{(\Delta^0)^2 - \Delta^2}{(\Delta^0 - |\Delta|x)^2} \\ &= \frac{\pi}{16} ((\Delta^0)^2 - \Delta^2)^2 \int_{-1}^1 \frac{(\Delta^0)^2 + \Delta^2 - 2\Delta^0|\Delta|x}{(\Delta^0 - |\Delta|x)^4} dx \end{aligned} \quad (\text{A.47})$$

ここで、積分変数を x から y に変換する。

$$y = |\Delta|x - \Delta^0 \quad (\text{A.48})$$

とおくと、

$$\begin{aligned} dy &= |\Delta|dx \\ x &= \frac{y + \Delta^0}{|\Delta|} \end{aligned} \quad (\text{A.49})$$

$$x: -1 \rightarrow 1 \quad \text{に対して} \quad y: -|\Delta| - \Delta^0 \rightarrow |\Delta| - \Delta^0 \quad (\text{A.50})$$

であることより、

$$\begin{aligned} S^{00} &= \frac{\pi}{16} ((\Delta^0)^2 - \Delta^2)^2 \int_{-|\Delta| - \Delta^0}^{|\Delta| - \Delta^0} \frac{\Delta^2 - (\Delta^0)^2 - 2\Delta^0 y}{y^4} \frac{dy}{|\Delta|} \\ &= \frac{\pi}{48} \left[\frac{((\Delta^0)^2 - \Delta^2)^3}{|\Delta|} \left(\frac{1}{(|\Delta| - \Delta^0)^3} + \frac{1}{(|\Delta| + \Delta^0)^3} \right) \right] \\ &+ \frac{\pi}{48} \left[\frac{((\Delta^0)^2 - \Delta^2)^2}{|\Delta|} 3\Delta^0 \left(\frac{1}{(|\Delta| - \Delta^0)^2} - \frac{1}{(|\Delta| + \Delta^0)^2} \right) \right] \end{aligned}$$

いま、カッコ () 内がそれぞれ

$$\left(\frac{1}{(|\Delta| - \Delta^0)^3} + \frac{1}{(|\Delta| + \Delta^0)^3} \right) = \frac{|\Delta|(2\Delta^2 + 6(\Delta^0)^2)}{(\Delta^2 - (\Delta^0)^2)^3} \quad (\text{A.51})$$

$$\left(\frac{1}{(|\Delta| - \Delta^0)^2} - \frac{1}{(|\Delta| + \Delta^0)^2} \right) = \frac{4|\Delta|\Delta^0}{(\Delta^2 - (\Delta^0)^2)^2} \quad (\text{A.52})$$

となる。したがって、

$$S^{00} = \frac{\pi}{48} [-2\Delta^2 - 6(\Delta^0)^2 + 12(\Delta^0)^2] \quad (\text{A.53})$$

$$= \frac{1}{24} [-\Delta^2 + 3(\Delta^0)^2] \quad (\text{A.54})$$

$$= \frac{1}{24} [\Delta^2 + 2(\Delta^0)^2] \quad (\text{A.55})$$

となる。式 (A.54) から式 (A.55) に行くために 4 元運動量の内積が $\Delta^2 = \Delta^\mu \Delta_\mu = (\Delta^0)^2 - \Delta^2$ であることを用いた。

以上より、 $\mu = 0, \nu = 0$ の時が証明できた。 $\mu = 0, \nu = 0$ の場合が分かれば、係数の値が決まるので、ローレンツ不変性より、 μ や ν がゼロ出ない場合も自動的に決定される。

付録 B

宇宙線が同時に入射した場合の検討

B.1

ここで、の図より、ADC(e) の T1 のグラフを確認してみる。TDC 分布が平坦になっている 2500 カウント以上である条件をつけ足して ADC(e) 分布を見てみる。条件は、

- $mut1 > 500$
- $es1 > 100$
- $tdc > 2500$

とした。この条件で図 B.2 をプロットした。図 B.2 より、et1 にも波高が見られた。図 B.1 のように考えると、この波高部分をカットすると μ 粒子が同時に入射した事象をのぞけるのではないかと考えた。よって、電荷区別なしの条件での et1 のカット値を様々に変更した TDC 分布のグラフを見てみる。

図 B.3 ~ 図 B.6 は、本章の結果でも用いたように、TDC4000 カウントまでで、Fit 関数に定数項を入れたものでプロットした。これらのグラフから平均寿命を計算した。次の表にまとめる。

表の結果を見ると、寿命についてはそれほど電荷区別なしと値は変わらないものと大き

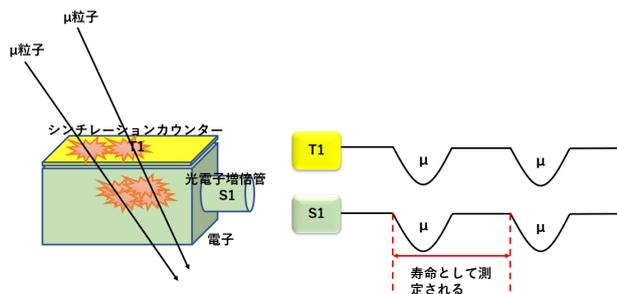


図 B.1 μ 粒子が同時に入射する事象のイメージ図

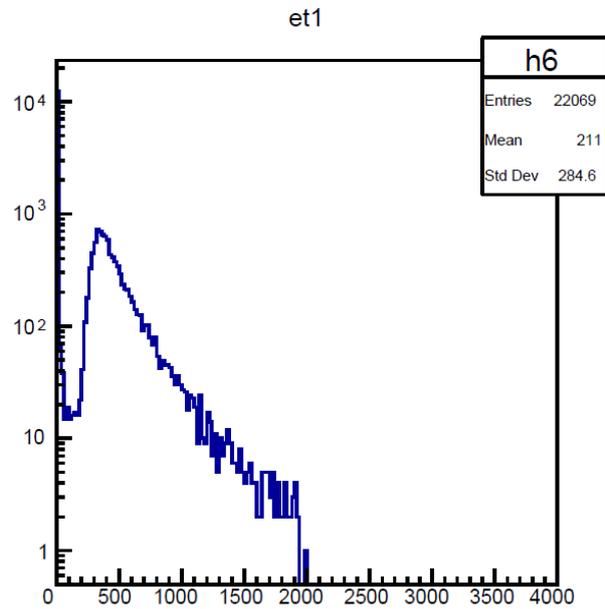


図 B.2 TDC が TDC > 2500 の事象に対する T1 カウンターの ADC (e) の分布

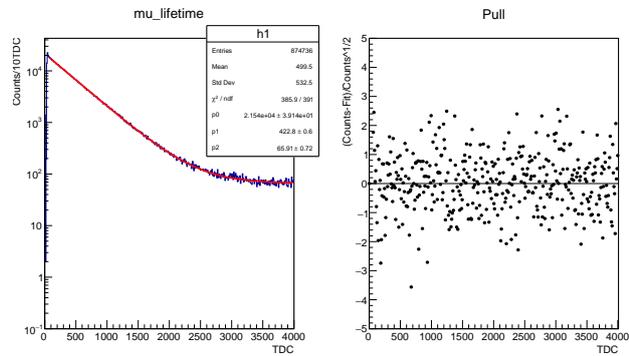


図 B.3 TDC 分布 (et1 < 100)

くなっているものがある。また、et1 < 50 よりカット値を小さくすると Pull のグラフに二次関数系が見られることや、 χ^2/ndf の値が大きくなることから et1 のカット値は 100 または 80 がいいと考えられる。図 B.3～図 B.6 からわかるように、平坦な成分は取り除かれなかった。T1 の ADC(e) では μ 粒子が同時に入った事象ではなく、 μ 粒子の崩壊で発生した電子によるものがほとんどのようである。なので、少し平坦な部分が緩やかに なったようにも見えるが、全てはカット出来ていないことが分かる。et1 になるのは、

- もう一つの μ 粒子が入射した場合
- 崩壊した μ 粒子からの電子

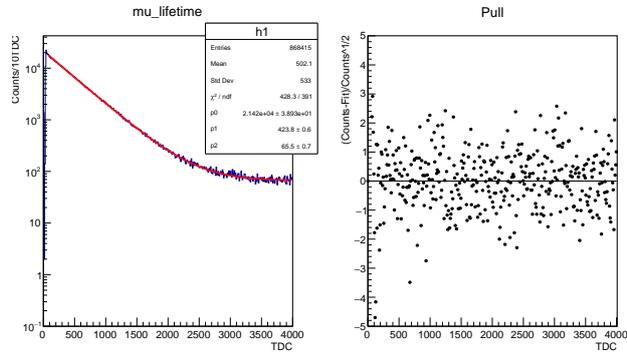


图 B.4 TDC 分布 ($et1 < 80$)

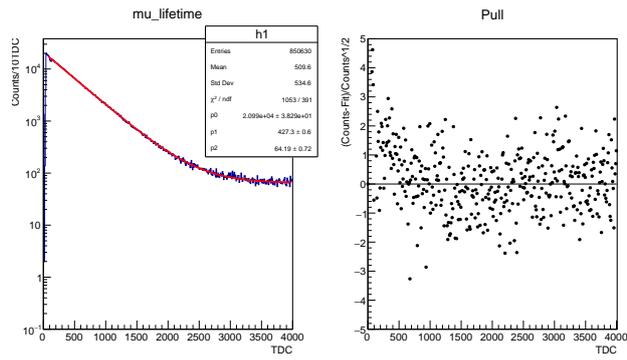


图 B.5 TDC 分布 ($et1 < 50$)

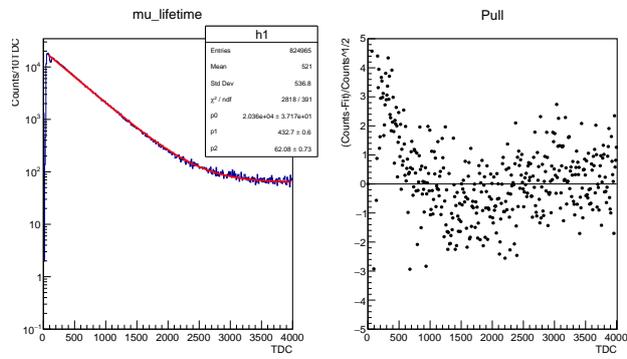


图 B.6 TDC 分布 ($et1 < 30$)

表 B.1 et1 のカット値を入れた μ 粒子の寿命

et1	イベント数	χ^2/ndf	ビン幅
et1 < 2000	1038320	1.067	5
et1 < 100	874736	0.987	10
et1 < 80	868415	1.095	10
et1 < 50	850630	2.693	10
et1 < 30	824965	7.207	10

et1	$p_1 \pm \delta p_1$	$p_2 \pm \delta p_2$	τ_μ
et1 < 2000	421.4 \pm 0.6	64.0 \pm 0.5	2.107 \pm 0.013
et1 < 100	422.8 \pm 0.6	65.9 \pm 0.7	2.114 \pm 0.013
et1 < 80	423.8 \pm 0.6	65.5 \pm 0.7	2.119 \pm 0.013
et1 < 50	427.3 \pm 0.6	64.2 \pm 0.7	2.136 \pm 0.013
et1 < 30	432.7 \pm 0.6	62.1 \pm 0.7	2.164 \pm 0.013

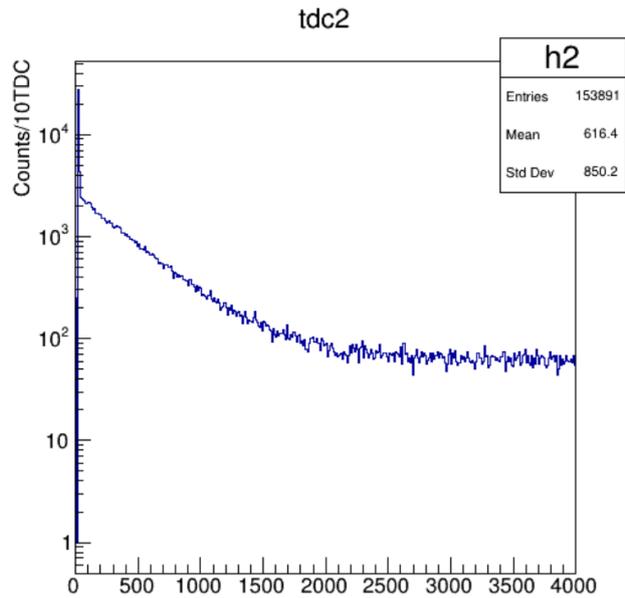


図 B.7 T1 の ADC(e)(et1) の値が 150 以上の時の時間 (TDC) 分布
 分布は一定値ではなく、 μ 粒子の崩壊による直線に下がる成分が存在している。

の 2 つの場合がある。そこで et1 にきついカットを入れると 2 つ目の崩壊した事象もカットしてしまっている可能性があるため、表のように値が小さくなると pull に 2 次の成分が出てくると考えられる。

付録 C

解析に用いたソースコード

C.1 ele_mag.c

CAMAC からのデータ収集プログラム

```

/***** ele_mag.c *****/ created 2012/Feb./1th*****
 * Original version was written by S.Ono 2002/Jan./27th
 * This is simplified version having only CAMAC control/IO part.
 * LAM clear was moved to outside of "if(q!=0)". 2003/Jul./9th
 * Update for stopped muon spectrum reading. 2013/11/27
 * Delete not need sentence. 2015/11/6
 *****/
/**** original title comment *****/
 *****/
#include <fcntl.h>
#include <sys/time.h>
#include <unistd.h>
#include <sys/types.h>
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <math.h>
#include <sys/errno.h>
#include "camlib.h"
#include <time.h>
#include <netinet/in.h>

FILE *fp; /* The file to save the taken data.*/

const char *f_lockname = "/tmp/.ele_mag_lock";

int main(){
    int i_ev, n_ev;
    int q,x,data21,data22,data23,data24,data31,data32,data33;
    //data34:TDC時間較正
    int data1,data2,data3,data4,data5,data6,data7,data8,data9,

```

```

data10 , data11 , data12 ;
int  data25 , data26 ;
int  cadcqn1 , cadcqn2 , ctdcqn ;
int  ch0 , ch1 , ch2 , ch3 , ch4 , ch5 , ch6 , ch7 , ch8 , ch9 , ch10 , ch11 ;
int  lamsrc ;
int  lamch ;

char  fname [36] ;

int  qq ;
int  head ;

// create lock file
FILE *fp_lock ;
fp_lock=fopen(f_lockname , "w") ;
fclose(fp_lock) ;

cadcqn1=1; /* muon ADC-Qmode module number */
cadcqn2=2; /* electron  ADC-Qmode module number */
ctdcqn=3; /* TDC  module number */
ch0=0;
ch1=1;
ch2=2;
ch3=3;
ch4=4;
ch5=5;
ch6=6;
ch7=7;
ch8=8;
ch9=9;
ch10=10;
ch11=11;

/*=====
 * Ask the file name to save the taken data.
 * Also open the data file.
 *=====*/

printf(" File name to save data?\n");
scanf("%s" , fname );
fp=fopen(fname , "w");

/*=====
 * How many events do you take?
 *=====*/
printf(" Number of events?\n");
scanf("%d" , &n_ev );
fprintf(fp , "%d\n" , n_ev );

```

```

/*=====
 * Open CCP interface device file .
 * If it fails , exit .
 *=====*/
if(COPEN()){
    printf("ccp open error\n");
    remove(f_lockname);
    exit(-1);
}

/*=====
 * Initialize CAMAC.
 *=====*/
CSETCR(0);
CGENZ();
CGENC();
CREMI();

lamsrc=cadcqn1;
//lamsrc=cadcqn2;
//lamsrc=ctdcqn; /* time calibration */
lamch=ch0;

printf("LAM source : %d,%d\n" ,lamsrc ,lamch);

/*=====
 * Send enable command to prepare the first event .
 *=====*/
//CAMAC(NAF(cadcqn1 ,ch0 ,26) ,&data1 ,&q ,&x);
/* F=26 is enable . */
CAMAC(NAF(lamsrc ,lamch ,26) ,&data1 ,&q ,&x);
/* F=26 is enable . */
// CAMAC(NAF(cadcqn2 ,ch0 ,26) ,&data1 ,&q ,&x);
/* F=26 is enable . */
// CAMAC(NAF(ctdcqn ,lamch ,26) ,&data1 ,&q ,&x);
/* F=26 is enable . */

/*=====
 * send message to the user .
 *=====*/
printf("CAMAC initilize done.\n");
printf("Number of event = %d\n" ,n_ev);
    time_t current = time(NULL);
    struct tm *timer = localtime(&current);
printf("start time %d/%d/%d %d:%d:%d\n" ,

```

```

        timer->tm_year+1900,
        timer->tm_mon+1,
        timer->tm_mday,
        timer->tm_hour,
        timer->tm_min,
        timer->tm_sec);

/*====
 * Event loop.
 *====*/
i_ev = 1;
while( i_ev <= n_ev )
{

CAMAC(NAF(lamsrc , lamch ,9), &data1, &q, &x);
/* F=9 LAM and ADC(mu) clear. */
CAMAC(NAF(cadcqn2 , ch0 ,9), &data1, &q, &x);
/* F=9 LAM and ADC(e) clear. */
CAMAC(NAF(ctdcqn , ch0 ,9), &data1, &q, &x);
/* F=9 LAM and TDC clear. */
//CAMAC(NAF(cadcqn1 , ch0 ,9), &data1, &q, &x);
/* time calibration */

CAMAC(NAF(lamsrc , lamch ,9), &data1, &q, &x);
CAMAC(NAF(cadcqn2 , ch0 ,9), &data1, &q, &x);
CAMAC(NAF(ctdcqn , ch0 ,9), &data1, &q, &x);

/*-----
 * Test LAM.
 *-----*/

do {
CAMAC(NAF(lamsrc , lamch ,8), &data1, &q, &x); /* F=8 is test LAM.*/
} while ( q==0);

/*-----
 * If no event comes yet, q is set to be 0,
 * otherwise, the digitized event is there!
 *-----*/

/*-----
 * Read the digitized data from the register.
 *-----*/
CAMAC(NAF(cadcqn1 , ch0 ,0), &data21, &qq, &x);
CAMAC(NAF(cadcqn1 , ch1 ,0), &data22, &qq, &x);
CAMAC(NAF(cadcqn1 , ch2 ,0), &data23, &qq, &x);
CAMAC(NAF(cadcqn1 , ch3 ,0), &data24, &qq, &x);
// -20181220 add —
CAMAC(NAF(cadcqn1 , ch4 ,0), &data25, &qq, &x);

```

```

CAMAC(NAF(cadcqn1 , ch5 , 0) , &data26 , &qq , &x) ;
//---
CAMAC(NAF(cadcqn2 , ch0 , 0) , &data1 , &qq , &x) ;
CAMAC(NAF(cadcqn2 , ch1 , 0) , &data2 , &qq , &x) ;
CAMAC(NAF(cadcqn2 , ch2 , 0) , &data3 , &qq , &x) ;
CAMAC(NAF(cadcqn2 , ch3 , 0) , &data4 , &qq , &x) ;
CAMAC(NAF(cadcqn2 , ch4 , 0) , &data5 , &qq , &x) ;
//ch4~ch11がcadcqn1のときADC(μでV1~V8を読む
CAMAC(NAF(cadcqn2 , ch5 , 0) , &data6 , &qq , &x) ;
CAMAC(NAF(cadcqn2 , ch6 , 0) , &data7 , &qq , &x) ;
CAMAC(NAF(cadcqn2 , ch7 , 0) , &data8 , &qq , &x) ;
CAMAC(NAF(cadcqn2 , ch8 , 0) , &data9 , &qq , &x) ;
CAMAC(NAF(cadcqn2 , ch9 , 0) , &data10 , &qq , &x) ;
CAMAC(NAF(cadcqn2 , ch10 , 0) , &data11 , &qq , &x) ;
CAMAC(NAF(cadcqn2 , ch11 , 0) , &data12 , &qq , &x) ;
//2014/11/18:
// Use new adc at slot 5, channel 0,
// since we can not read the adc at channel 11.
//CAMAC(NAF(4,0,2),&data12,&qq,&x);
CAMAC(NAF(ctdcqn , ch0 , 2) , &data31 , &qq , &x) ;
CAMAC(NAF(ctdcqn , ch1 , 2) , &data32 , &qq , &x) ;
CAMAC(NAF(ctdcqn , ch2 , 2) , &data33 , &qq , &x) ;
// CAMAC(NAF(ctdcqn , ch3 , 2) , &data34 , &qq , &x) ; //TDC時間較正

/*-----
 * Update the event counter(i_ev), and send message for
 * every 50 events or there are data for big electron signal.
 *-----*/
// if (data12>100)
if( i_ev%100 == 1 )
// if( i_ev%500 == 1 || data1>50 )
//if( data1>10 ) /* Cut of event not decay to electron */
{ time_t current = time(NULL);
  struct tm *timer = localtime(&current);
  printf("%d/%d/%d %d:%d:%d\n" ,
         timer->tm_year+1900,
         timer->tm_mon+1,
         timer->tm_mday,
         timer->tm_hour ,
         timer->tm_min ,
         timer->tm_sec );
  printf(" Event= %d\n" , i_ev );
  printf(" data21= %d\n" , data21 );
  printf(" data22= %d\n" , data22 );
  printf(" data23= %d\n" , data23 );
  printf(" data24= %d\n" , data24 );
//20181220 add ---
  printf(" data25= %d\n" , data25 );
  printf(" data26= %d\n" , data26 );
//--

```

```

printf(" data1= %d\n", data1);
printf(" data2= %d\n", data2);
printf(" data3= %d\n", data3);
printf(" data4= %d\n", data4);
printf(" data5= %d\n", data5);
printf(" data6= %d\n", data6);
printf(" data7= %d\n", data7);
printf(" data8= %d\n", data8);
printf(" data9= %d\n", data9);
printf(" data10= %d\n", data10);
printf(" data11= %d\n", data11);
printf(" data12= %d\n", data12);
printf(" data31= %d\n", data31);
printf(" data32= %d\n", data32);
printf(" data33= %d\n", data33);
//printf(" data34= %d\n", data34); //TDC時間較正

}

head=-1;

/*-----
 * The read data is written into the file.
 *-----*/

//--
//   if( data1>10 ) /* write events if muon decays to electron */
//
//   {
//
//     time_t current = time(NULL);
//     struct tm *timer = localtime(&current);

fprintf(fp, " %d", head);
fprintf(fp, " %d", i_ev);
fprintf(fp, " %d", data21);
fprintf(fp, " %d", data22);
fprintf(fp, " %d", data23);
fprintf(fp, " %d", data24);
// -20181220 add --
fprintf(fp, " %d", data25);
fprintf(fp, " %d", data26);
//--
fprintf(fp, " %d", data1);
fprintf(fp, " %d", data2);
fprintf(fp, " %d", data3);
fprintf(fp, " %d", data4);
fprintf(fp, " %d", data5);
fprintf(fp, " %d", data6);

```

```

fprintf(fp," %d",data7);
fprintf(fp," %d",data8);
fprintf(fp," %d",data9);
fprintf(fp," %d",data10);
fprintf(fp," %d",data11);
fprintf(fp," %d",data12);
fprintf(fp," %d",data31);
fprintf(fp," %d",data32);
fprintf(fp," %d",data33);
fprintf(fp," %d/%d/%d %d:%d:%d\n" ,
        timer->tm_year+1900,
        timer->tm_mon+1,
        timer->tm_mday,
        timer->tm_hour ,
        timer->tm_min ,
        timer->tm_sec );

//      {

//fprintf(fp," %d\n",data34); //TDC時間較正

        }

/*-----
 * Clear LAM to wait for the next event.
 *   fuc=9 ; LAM clear for usual module
 *   FADC
 *   fuc=9 ; address clear
 *   =10 ; LAM clear
 *-----*/
i_ev++;

/*CAMAC(NAF(4,0,9),&data1,&q,&x);
CAMAC(NAF(lamsrc , lamch ,9),&data1,&q,&x);
CAMAC(NAF(lamsrc , lamch ,10),&data1,&q,&x); */

}/*   end of event loop   */

/*=====
Instructions for termination.
*=====*/
CAMAC(NAF(lamsrc , lamch ,24),&data1,&q,&x); /* F=24 is desable lam.*/
CAMAC(NAF(cadcqn1 , ch1 ,24),&data1,&q,&x); /* F=24 is desable lam.*/
CAMAC(NAF(cadcqn2 , ch1 ,24),&data1,&q,&x); /* F=24 is desable lam.*/
CAMAC(NAF(ctdcqn , ch1 ,24),&data1,&q,&x); /* F=24 is desable lam.*/

CCLOSE();          /* CAMAC close. */
fclose(fp);        /* Close data file.*/

```

```
// remove lock file when finished
remove(f_lockname);

printf(">>>>>>FINISH!<<<<<<\n\n");
return 0;

}
```

C.2 ele_mag.csh

データ収集において日付指定などを行う
 蜂谷先生が書いてくださったプログラム。ele_mag.c の中での手入力であるファイル名指定やイベント数の入力を自動で行えるもの。また、ele_mag.c のなかで作った lockfile の確認も行っている。

```
#!/usr/bin/tcsh -f

if( $#argv != 1 ) then
  echo "Usage : ele_mag.csh <nevent>"
  exit
endif

set command = "/home/2015-b4/ug2020/camdrv2020/ele_mag"
set datadir = "/home/2015-b4/ug2020/data2020"
set lockfile = "/tmp/.ele_mag.lock"
#set datadir = "/home/2015-b4/ug2020/data_tmp"
set datahead = 'date +%Y_%m_%d'

set nevent = $1

#####3
#ls $datadir | grep "$datahead"
set nfile = `ls $datadir | grep "$datahead" | wc -l`
@ fileindex = $nfile
if ($nfile == '0') then
  set datafile = "$datadir/${datahead}_${nevent}.dat"
else
  set datafile = "$datadir/${datahead}_${nevent}_${fileindex}.dat"
endif

#set datafile = "$datadir/${datahead}_${nevent}_${fileindex}.dat"

echo $nfile $fileindex $datafile

# Remove lockfile
if( -e $lockfile ) then
  echo "remove lockfile $lockfile"
  rm -f $lockfile
endif

$command << EOT &
$datafile
$nevent
EOT
```

}

C.3 watch_daq.csh

ele_mag.csh やエントリ一数の指定、lock ファイルと fsize の確認
 蜂谷先生が書いてくださったもの。ele_mag.csh を動かし、filesize や lockfile があるかを確認、表示するもの。
 check の間隔を決めることができ、filesize が変わっていれば 0 となり、lockfile があれば OK と表示される。

```
#!/usr/bin/tcsh -f

set nevent = "2400000"
#set nevent = "1000"
set command = "/home/2015-b4/ug2020/camdrv2020/ele_mag.csh"
set lockfile = "/tmp/.ele_mag_lock"

set outdir = "ug2020/data2020/"
#set outdir = "ug2020/data_tmp/"

#####
# stop all ele_mag process when starting
killall -9 ele_mag

# Remove lockfile
if( -e $lockfile ) then
  echo "remove lockfile $lockfile"
  rm -f $lockfile
endif
#####

@ same_count = 0
set pre_fsize = '0'
while (1)
  sleep 3s

#####
# check if ele_mag is running. if not, start it.
set val = `ps axuw | grep "ele_mag" | wc -l`
echo $val
#if( $val == 2 ) then
# echo "OK"
echo "check lockfile"
if( -e $lockfile ) then
  echo "OK —lock file exist"
else
  echo "exec"
  $command $nevent
endif

#####
# check file size. if size is same for 120s, restart the code
```

```
set fsize = `ls -lrt ${outdir}/*.dat | tail -n 1 | awk -F' ' '{print $5}`
echo "fsize $fsize"

if( $fsize == $pre_fsize ) then
  echo "same $same_count"
  @ same_count += 1
else if( $same_count >= 300 ) then
  ### stop ele_mag first
  killall ele_mag
  sleep 2s
  ### start ele_mag
  echo "exec"
  # ele_mag < 1000000 &
  $command $nevent
  @ same_count = 0
else
  echo "different count = $same_count"
  @ same_count = 0
endif

set pre_fsize = $fsize

end

}
```

C.4 rootfileOutput-mag.cc

CAMAC からのデータを root ファイルへ出力

```

#include "fstream"
#include "TNtuple.h"
#include "TH1.h"
#include "TH2.h"
#include "TCanvas.h"
#include "TFile.h"
#include <string>
#include "TTree.h"
using namespace std;

void rootfileOutput_mag(){

    string    fname="2021_01_06_1200000"; //参照元ファイル名

    string    directory1="/home/2015-b4/ug2020/data2020/";
    string    directory2="/home/2015-b4/ug2020/data2020/";
    string    full_name1 = directory1+ fname+ ".dat"; //データファイルの絶対パス
    string    full_name2 = directory2+ fname+ ".root"; //rootファイルの絶対パス
    cout <<" f-name"<< full_name1<<endl;

    ifstream data(full_name1.c_str()); //ファイルを開く
    float one, ev, mus1, mus2, mut1, mut2, mut3, mut4,
    es1, es2, et1, et2, ev1, ev2, ev3, ev4, ev5, ev6, ev7, ev8,
    tdc0, tdc1, tdc2;
    string yeardate, time;
    //型宣言

    TTree *mu_t=new TTree("mu_t", "title"); //Treeを定義
    mu_t->Branch("one", &one, "one/F"); //Branch(箱)を定義
    mu_t->Branch("ev", &ev, "ev/F");
    mu_t->Branch("mus1", &mus1, "mus1/F");
    mu_t->Branch("mus2", &mus2, "mus2/F");
    mu_t->Branch("mut1", &mut1, "mut1/F");
    mu_t->Branch("mut2", &mut2, "mut2/F");
    mu_t->Branch("mut3", &mut3, "mut3/F"); //20181221-----
    mu_t->Branch("mut4", &mut4, "mut4/F"); //-----
    mu_t->Branch("es1", &es1, "es1/F");
    mu_t->Branch("es2", &es2, "es2/F");
    mu_t->Branch("et1", &et1, "et1/F");
    mu_t->Branch("et2", &et2, "et2/F");
    mu_t->Branch("ev1", &ev1, "ev1/F");
    mu_t->Branch("ev2", &ev2, "ev2/F");
    mu_t->Branch("ev3", &ev3, "ev3/F");
    mu_t->Branch("ev4", &ev4, "ev4/F");

```

```

mu_t->Branch(" ev5", &ev5, " ev5/F");
mu_t->Branch(" ev6", &ev6, " ev6/F");
mu_t->Branch(" ev7", &ev7, " ev7/F");
mu_t->Branch(" ev8", &ev8, " ev8/F");
mu_t->Branch(" tdc0", &tdc0, " tdc0/F");
mu_t->Branch(" tdc1", &tdc1, " tdc1/F");
mu_t->Branch(" tdc2", &tdc2, " tdc2/F");

if (!data){cout<<"wrong file name"<<endl;}
data>>one; //一番上の合計イベント数だけ読む(ループから抜くため)
cout<<" fiest line+="<<one<<endl;
int i=1;

cout <<" before while"<<endl;

while( data >> one >> ev >> mus1 >> mus2 >>
      mut1 >> mut2 >> mut3 >> mut4 >>
      es1 >> es2 >> et1 >> et2 >>
      ev1 >> ev2 >> ev3 >> ev4 >> ev5 >>
      ev6 >> ev7 >> ev8 >> tdc0 >> tdc1 >> tdc2 >>
      yeardate >> time){
cout<< " i="<<i<<" mus1="<<mus1<<endl;
cout << " year-date="<<yeardate<<" time"<<time<<endl;
mus1=mus1-124; //ペDESTALを引く
if (mus1<0){mus1=0;} //ペDESTALを引いて負になる場合は0に揃える
mus2=mus2-104;
if (mus2<0){mus2=0;}
mut1=mut1-97;
if (mut1<0){mut1=0;}
mut2=mut2-93;
if (mut2<0){mut2=0;}
mut3=mut3-20;//20181221---
if (mut3<0){mut3=0;}
mut4=mut4-98;
if (mut4<0){mut4=0;}//-----
es1=es1-60;
if (es1<0){es1=0;}
es2=es2-63;
if (es2<0){es2=0;}
et1=et1-58;
if (et1<0){et1=0;}
et2=et2-63;
if (et2<0){et2=0;}
ev1=ev1-71;
if (ev1<0){ev1=0;}
ev2=ev2-77;
if (ev2<0){ev2=0;}
ev3=ev3-67;
if (ev3<0){ev3=0;}
ev4=ev4-66;

```

```

    if (ev4 < 0){ev4=0;}
ev5=ev5-68;
    if (ev5 < 0){ev5=0;}
ev6=ev6-64;
    if (ev6 < 0){ev6=0;}
ev7=ev7-69;
    if (ev7 < 0){ev7=0;}
ev8=ev8-81;
    if (ev8 < 0){ev8=0;}

if ( i%100 ==1){
cout <<" i="<<i<<" one="<<one<<" ev="<<ev<<"
mus1="<<mus1<<" mus2="<<mus2<<endl; // 出力
}
mu_t->Fill ();
i++;
};

data.close (); // ファイル閉じる

TFile *rtfilemt = new TFile(full_name2.c_str(),"RECREATE"); // rootファイルを作成
mu_t->Write ();
rtfilemt->Close (); // rootファイルを閉じる

TFile *f= new TFile(full_name2.c_str(),""); // rootファイルを開く
f->ls ();

return ;
}

```

C.5 tdc_2020.cc

TDC calibration に用いる値の出力

```

#include "fstream"
#include "TNtuple.h"
#include "TH1.h"
#include "TH2.h"
#include "TCanvas.h"
#include "TFile.h"
#include "TTree.h"
#include "TChain.h"
#include "TChain.h"
#include "iomani"
using namespace std;

void tdc_2020(){
  gStyle->SetOptStat(1000002211);
  gStyle->SetOptFit(1);

  TCanvas *c1=new TCanvas("c1","c1title",0,0,400,300);
  //c1->Divide(3,2);
  //c1->SetLogy(1);
  //c1->SetLogy(false);
  TChain *mu_t=new TChain("mu_t","mu_title");
  mu_t->Add("/home/2015-b4/ug2020/data2020/TDC/TDC_10_13_5500.root");

  // c1->cd(1);
  // TH1F *h1=new TH1F("h1","mut1",500,0,1500);
  /* cut*/
  // TCut positiveCut="mut3>100";
  TCut e_decay="120<es1 && tdc2<4000";
  TCut mag ="mut3>50 || mut4>225";
  //c1->SetLogy();
  //c1->cd(1);
  //TH1F *h1=new TH1F("h1","mut1",500,0,1500);
  //mu_t->Draw("mut1>>h1",e_decay,"");
  //h1->Draw();

  //c1->cd(12);
  //
  TH1F *h12=new TH1F("h12","tdc2",6000,0,6000);
  // gStyle->SetOptFit(2211);
  mu_t->Draw("tdc2>>h12","");
  float mean = h12->GetMean();
  float emean = h12->GetMeanError();
  float rms = h12->GetRMS();
  float erms = h12->GetRMSError();

```

```
cout<<setw(10)<<setprecision(7)<<mean<<" ";  
cout<<setw(10)<<setprecision(7)<<emean<<" ";  
cout<<setw(10)<<setprecision(7)<<rms<<" ";  
cout<<setw(10)<<setprecision(7)<<erms<<" ";  
cout<<endl;
```

```
return;
```

```
}
```

```
}
```

C.6 tdc-calibration2020

ROOT による TDC 時間較正

```

#include "fstream"
#include "TH1.h"
#include "TCanvas.h"
#include "TFile.h"
#include "TTree.h"
#include "TChain.h"
#include "TGraph.h"
#include "TMath.h"
#include "iomani"

void tdc_calibration2020 () {
    TCanvas *c1 = new TCanvas("c1","A Simple Graph with error bars",
        200,10,700,500);
    c1->Divide(2,1);
    //c1->SetFillColor(42);
    //c1->SetGrid();
    //c1->GetFrame()->SetFillColor(21);
    //c1->GetFrame()->SetBorderSize(12);
    const Int_t n=17;
    Double_t y[n]={97.752, 198.220,299.609 ,398.450 ,497.889 ,
        600.421, 700.063,799.606 ,898.942 ,998.872 ,
        1098.771,1198.591,1299.958,1398.594,1498.123 ,
        1598.454,1698.211};
    Double_t x[n]={500,1000,1500,2000,2500,
        3000,3500,4000,4500,5000,
        5500,6000,6500,7000,7500,
        8000,8500};
    Double_t ey[n]={2.,4.,4.,4.,10.,
        10.,10.,10.,10.,10.,
        20.,20.,20.,20.,20.,
        20.,20.};
    Double_t ex[n]={10,20,20,20,50,
        50,50,50,50,50,
        100,100,100,100,100,
        100,100};

    c1->cd(1);
    TGraphErrors *gr = new TGraphErrors(n,x,y,ex,ey);

    TF1 *func = new TF1("func","pol1");
    func->SetParameter(0,2);
    func->SetParameter(1,5);
    gr->Fit("func");
    gr->SetMarkerStyle(8);

```

```

gr->SetMarkerColor(2);
//gr->SetMarkerSize(1.5);
gr->GetYaxis()->SetTitle("TDC [count]");
gr->GetXaxis()->SetTitle("Delay Time [ns]");
gr->GetYaxis()->SetTitleOffset(1.3);
gr->GetXaxis()->SetTitleOffset(1.3);
gr->SetTitle("TDC Time Calibration");
gr->Draw("AP");
gStyle->SetOptFit(1);
Double_t Y[n], Pull[n];
Double_t p0 = func->GetParameter(0);
Double_t p1 = func->GetParameter(1);
//Double_t yy[n]={TMath::Abs((Double_t)p0 +
                    (Double_t)p1*(Double_t)x[n] - (Double_t)y[n])};
//cout << "p0=" << p0 << "p1=" << p1 << endl;
for (Int_t i = 0; i<n; i++){
    ey[i] = sqrt( (ex[i]* p1) * (ex[i]*p1) + (ey[i]*ey[i]) );
    Y[i] = p0 + p1*x[i];
    //Pull[i] = (y[i]-Y[i])/(TMath::Power(y[i],0.5));
    Pull[i] = (y[i]-Y[i])/ ey[i] ;
    //-- cout<<i<<" "<<x[i]<<" "<<y[i]<<" "<<Y[i]<<" "<<ey[i]<<endl;

};
cout<<setw(10)<<setprecision(7)<<p0<<" ";
cout<<setw(10)<<setprecision(7)<<p1<<" ";
c1->cd(2);
TH1F* frm= gPad->DrawFrame(0, -3, 9000, 3);
TGraph* Gr = new TGraph(n,x, Pull);
Gr->Draw("P");
//-- Gr->Print();
Gr->SetMarkerStyle(5);
Gr->SetMarkerSize(1.5);
frm->SetTitle(" Pull");
frm->GetYaxis()->SetTitle("( Counts-Fit )/1");
frm->GetXaxis()->SetTitle("Delay Time [ns]");

return;
}

```

C.7 histogramOutput2020

ROOT によるヒストグラム作成

```
#include "fstream"
#include "TNtuple.h"
#include "TH1.h"
#include "TH2.h"
#include "TCanvas.h"
#include "TFile.h"
#include "TTree.h"
#include "TChain.h"
using namespace std;

void histogramOutput2020 () {
    gStyle->SetOptFit(1);

    TCanvas *c1=new TCanvas("c1","c1title",0,0,400,300);
    c1->Divide(3,2);
    TChain *mu_t=new TChain("mu_t","mu_title");
    mu_t->Add("/home/2015-b4/ug2020/data2020/*.root");

    /* cut */
    TCut e_decay="120<es1 && tdc2<4000";
    TCut mag ="mut3>200 || mut4>200";

    c1->cd(1);
    TH1F *h1=new TH1F("h1","mut1",100,0,1500);
    mu_t->Draw("mut1>>h1",e_decay,"");

    c1->cd(2);
    TH1F *h2=new TH1F("h2","mut3",100,0,2500);
    mu_t->Draw("mut3>>h2",e_decay,"");

    c1->cd(3);
    TH1F *h3=new TH1F("h3","mut4",100,0,2500);
    mu_t->Draw("mut4>>h3",e_decay,"");

    c1->cd(4);
    TH1F *h4=new TH1F("h4","es1",100,0,1500);
    mu_t->Draw("es1>>h4",e_decay,"");

    c1->cd(5);
    TH1F *h5=new TH1F("h5","mus1",100,0,1500);
    mu_t->Draw("mus1>>h5",e_decay,"");

    c1->cd(6);
    TH1F *h6=new TH1F("h6","et1",100,0,1500);
    mu_t->Draw("et1>>h6",e_decay,"");
}
```

```
    return;  
}  
  
}
```

C.8 histlifetime2020.cc

ROOT による μ 粒子の寿命解析

```
#include "fstream"
#include "TNtuple.h"
#include "TH1.h"
#include "TH2.h"
#include "TCanvas.h"
#include "TFile.h"
#include "TTree.h"
#include "TChain.h"

using namespace std;

void hist_lifetime2020 () {

    TCanvas *c1 = new TCanvas("c1", "ctitle", 0, 0, 900, 500);
    // 新しいキャンバスを作成
    c1->Divide(2, 2); // キャンバスを分割

    TChain *mu_ch=new TChain("mu_t", "mu_title"); // Chainを作成
    mu_ch->Add("/home/2015-b4/ug2020/data2020/*.root"); // Chainに入れる

    TCut e_decay="120<es1 && tdc2<4000";
    TCut mag="120<es1 && tdc2<4000 && (50<mut3 || 225<mut4)";
    TCut nonmag=" 120<es1 && tdc2<4000 ";

    /*  $\mu$  粒子寿命測定*/
    c1->cd(1); // Padを移動
    TH1F *h1= new TH1F("h1", "mu_lifetime", 500, 0, 2000);
    // 新しいヒストグラムを作成
    mu_ch->Draw("tdc2>>h1", nonmag, ""); // ヒストグラムを出力
    h1->GetXaxis()->SetTitle("TDC");
    h1->GetYaxis()->SetTitle("Counts");
    h1->SetTitle("mu_lifetime");

    /* Fit */
    //c1->cd(1);
    TF1 *f_exp_y=new TF1("f_exp_y", "[0]*exp(-x/[1])", 0, 2000);
    //f_exp_y=p0+exp(-x/p1)
    f_exp_y->SetParameter(0, 665);
    // p2無しパラメータの初期値を設定(i番目, 初期値)
    f_exp_y->SetParameter(1, 435);
    GetBinCenter
    h1->Fit("f_exp_y", "", "", 60, 2000); // f_exp_yでFit
```

```

    gStyle->SetOptFit(1); //キャンバスにパラメータを表示
    Double_t p0_y = f_exp_y->GetParameter(0); //p0を取得
    Double_t p1_y = f_exp_y->GetParameter(1); //p1を取得

    cout << " p0_y=" << p0_y << " p1_y=" << p1_y << endl;

    const Int_t n=500;
    Double_t x[n],y[n],Y[n],y-Y[n],X[n],pull[n];
    for (Int_t i=1 ; i<n ; i++){
        x[i]= h1->GetBinCenter(i);
        y[i]= h1->GetBinContent(i);
        Y[i]= p0_y*exp(-x[i]/p1_y);
        pull[i]=(y[i]-Y[i])/(TMath::Power(y[i],0.5));
    };

    c1->cd(2);
    TGraph* g=new TGraph(n,x,pull);
    g->GetYaxis()->SetRangeUser(-5,5);
    g->GetXaxis()->SetRangeUser(0,2000);

    g->Draw("AP");

    g->SetMarkerStyle(7);
    g->SetTitle("Pull");

    return ;
}

```

C.9 histlifetime2020.cc

ROOT による μ 粒子の寿命解析

```
#include "fstream"
#include "TNtuple.h"
#include "TH1.h"
#include "TH2.h"
#include "TCanvas.h"
#include "TFile.h"
#include "TTree.h"
#include "TChain.h"

using namespace std;

void histlifetime2020 () {

    TCanvas *c1 = new TCanvas("c1", "ctitle", 0, 0, 900, 400); //新しいキャンバスを作成
    c1->Divide(2,1); //キャンバスを分割

    TChain *mu_ch=new TChain("mu_t", "mu_title"); //Chainを作成
    //mu_ch->Add("/home/2015-b4/ug2020/data2020/*.root"); //Chainに入れる
    mu_ch->Add("/export/home/minugai/ug2020_data/data2020/*.root");
    mu_ch->Add("/export/home/minugai/ug2020_data/data2019/*.root");
    mu_ch->Add("/export/home/minugai/ug2020_data/data2018/mu-/*.root");
    //mu_ch->Add("/home/2015-b4/ug2018/data2018/mu-/*.root"); //Chainに入れる
    //mu_ch->Add("/home/2015-b4/ug2019/data2019/mu-/*.root");

    TCut e_decay="mut1>500 && 100<es1 && tdc2<4000";
    TCut mag="mut1>500 && 100<es1 && tdc2<4000 && (300<mut3 || 200<mut4)";
    TCut nonmag=" 100<es1 && tdc2<4000 ";

    /*  $\mu$  粒子寿命測定*/
    c1->cd(1); //Padを移動
    TH1F *h1= new TH1F("h1", "mu_lifetime",150,0,2000); //新しいヒストグラムを作成
    //gPad->SetLogy()
    //h1->SetMinimum(3.);
    mu_ch->Draw("tdc2>>h1",mag,""); //ヒストグラムを出力
    h1->GetXaxis()->SetTitle("TDC");
    h1->GetYaxis()->SetTitle("Counts");
    h1->SetTitle("mu_lifetime");

    /* Fit */
    //c1->cd(1);
    TF1 *f_exp_y=new TF1("f_exp_y", "[0]*exp(-x/[1])",0,2000); //f_exp_y=p0+exp(-x/p1)
    f_exp_y->SetParameter(0,665); //p2無しパラメータの初期値を設定(i番目,初期値)
    f_exp_y->SetParameter(1,435);
```

```

h1->Fit("f_exp-y","",",",60,2000); // f_exp-y で Fit

gStyle->SetOptFit(1); // キャンパスにパラメータを表示

//TH1F *f=new TH1F("f","f_exp-y",500,0,2000);
//f->FillRandom("f_exp-y");
//TH1F *h1minusf = new TH1F("h1" - "f");
//c1->cd(2);
//f->Draw();

//c1->cd(2);
// TF1 *f_exp_n=new TF1("f_exp_n", "[0]*(exp(-x/[1])+exp(-x/431.8))"); // f_exp_n=p0exp(-x/p1)+p2
//f_exp_n->SetParameter(0,665); // p2ありパラメータの初期値を設定(i番目,初期値)
//f_exp_n->SetParameter(1,435);
//f_exp_n->SetParameter(2,650);
//h2->Fit("f_exp_n","",",",60,2000); // f_exp_n で Fit
//gStyle->SetOptFit(1); // キャンパスにパラメータを表示

Double_t p0_y = f_exp_y->GetParameter(0); // p0を取得
Double_t p1_y = f_exp_y->GetParameter(1); // p1を取得

//Double_t p0_n = f_exp_n->GetParameter(0); // p0を取得
//Double_t p1_n = f_exp_n->GetParameter(1); // p1を取得
//Double_t p2_n = f_exp_n->GetParameter(2); // p2を取得

cout << " p0_y=" << p0_y << " p1_y=" << p1_y << endl;
//cout << " p0_n=" << p0_n << " p1_n=" << p1_n << endl; // パラメータを表示 p2無し
//cout << " p0_n=" << p0_n << " p1_n=" << p1_n << " p2_n=" << p2_n << endl; // パラメータを表示

const Int_t n=150;
Double_t x[n],y[n],Y[n],y-Y[n],X[n],pull[n];
for (Int_t i=1 ; i<n ; i++){
    x[i]= h1->GetBinCenter(i);
    y[i]= h1->GetBinContent(i);
    Y[i]= p0_y*exp(-x[i]/p1_y);
    pull[i]=(y[i]-Y[i])/(TMath::Sqrt(y[i]));
    //cout << " x=" << x[i] << " y=" << y[i] << " Y=" << Y[i] << "
y-Y=" <<y-Y[i]<< endl ;
};

c1->cd(2);
TGraph* g=new TGraph(n,x,pull);
g->GetYaxis()->SetRangeUser(-5,5);

```

```

g->GetXaxis()->SetRangeUser(0,2000);

g->Draw("AP");

g->SetMarkerStyle(7);
g->SetTitle("Pull");
g->GetXaxis()->SetTitle("TDC");
g->GetYaxis()->SetTitle("(Counts-Fit)/Counts^1/2");

TLine *line = new TLine(0,0,2000,0);
line->SetLineColor(1);
line->SetLineWidth(1);
//line->SetLineStyle(1);
line->Draw();

/*  $\mu$  粒子寿命測定*/
// c1->cd(3); // Padを移動
//TH1F *h2= new TH1F("h2", "mu_lifetime",500,0,2000); //新しいヒストグラムを作成
// mu_ch->Draw("tdc2>>h2",nonmag,""); //ヒストグラムを出力
// h2->GetXaxis()->SetTitle("TDC");
// h2->GetYaxis()->SetTitle("Counts");
// h2->SetTitle("mu_lifetime");

/* Fit */
//c1->cd(2);
// TF1 *f_exp_n=new TF1("f_exp_n", "[0]*(exp(-x/[1])+exp(-x/[2]
))"); //f_exp_n=p0exp(-x/p1)+p2exp*-x/431.8)
// f_exp_n->SetParameter(0,665); // p2ありパラメータの初期値を設定(i番目,初期値)
// f_exp_n->SetParameter(1,435);
//f_exp_n->SetParameter(2,650);
//h2->Fit("f_exp_n", "", "", 60,2000); //f_exp_nでFit
//gStyle->SetOptFit(1); //キャンバスにパラメータを表示

//Double_t p0_n = f_exp_n->GetParameter(0); //p0を取得
//Double_t p1_n = f_exp_n->GetParameter(1); //p1を取得
//Double_t p2_n = f_exp_n->GetParameter(2); //p2を取得

//cout << " p0_n=" << p0_n << " p1_n=" << p1_n << " p2_n=" << p2_n << endl; //パラ

//const Int_t n=500;
//Double_t x[n],y[n],Y[n],y-Y[n],X[n],pull[n];
//Double_t YY[n],pull2[n];
//for (Int_t i=1 ; i<n ; i++){
// x[i]= h2->GetBinCenter(i);
// y[i]= h2->GetBinContent(i);
// YY[i]= p0_n*(exp(-x[i]/p1_n)+exp(-x[i]/p2_n));

```

```

// pull2 [ i ]=(y [ i ]-YY [ i ])/(TMath::Power(y [ i ],0.5));
//cout << " x=" << x [ i ] << " y=" << y [ i ] << " Y=" << Y [ i ] << "
y-Y=" <<y-Y [ i]<< endl ;
//};

//c1->cd (4);
//TGraph* gg=new TGraph(n,x,pull2);
//gg->GetYaxis()->SetRangeUser(-5,5);
//gg->GetXaxis()->SetRangeUser(0,2000);

// gg->Draw("AP");

//gg->SetMarkerStyle(7);
//gg->SetTitle("Pull2");
//gg->GetXaxis()->SetTitle("TDC");
//gg->GetYaxis()->SetTitle("(Counts-Fit)/Counts^1/2");

//TLine *line2 = new TLine(0,0,2000,0);
//line2->SetLineColor(1);
// line2->SetLineWidth(1);
//line2->SetLineStyle(1);
//line2->Draw();

//c1->Print("./pdf_file/2000_noconstL1000'.pdf");
return ;
}

```