

平成27年度 修士論文

ボディアエリアネットワークにおける動的通信チャネルモデルの構築

指導教員 本島 邦行 教授

群馬大学大学院理工学府 理工学専攻
電子情報・数理教育プログラム

笠原 佑作

目次

1. 序論	1
2. 測定システムの構築.....	3
2.1. 測定方法と新システムの設計要件.....	3
2.2. 受信器の設計.....	6
2.2.1. 受信器の設計要件	6
2.2.2. システムの構成（受信器）	6
2.2.3. バッファアンプ.....	7
2.2.4. インピーダンス整合兼フィルタ回路.....	8
2.2.5. 検波回路（AD8307）を用いる	12
2.2.6. 直流バッファアンプ	14
2.2.7. マイクロコントローラを選定とピンアサイン	16
2.2.8. micro SD カードのピンアサイン	17
2.2.9. 受信器のプログラム	19
2.2.10. 受信器の回路図と使用部品.....	20
2.2.11. 受信器の完成と動作スペック	22
2.3. 送信器の設計.....	24
2.3.1. 送信器の設計要件	24
2.3.2. システムの構成（送信器）	24
2.3.3. 送信器の回路図と使用部品	24
2.3.4. 送信器の完成と動作スペック	26
3. 測定の内容と条件.....	28
4. 測定結果	30
4.1. 受信電圧強度の時間変化	30
4.2. 受信電圧強度の確率密度関数	32
5. 解析手法	34
5.1. 複数の正規分布を用いた近似	34
5.2. 最適化と評価関数	36
5.3. 最適化手法	37
5.4. 最適化プログラム	39
6. 解析結果	41
6.1. 解析対象データの平滑化	41
6.2. 最適化結果	42

6.3.	正規分布の数 n に関する依存性.....	44
6.4.	最適化結果からラジオ体操第一の動作を考える.....	46
7.	結論.....	47
8.	今後の課題および方針.....	48
9.	謝辞.....	49
10.	参考文献.....	50
11.	付録.....	51
12.	研究業績.....	56

1. 序論

近年、ウェアラブル機器同士で通信を行う人体通信の技術は、医療、セキュリティあるいはパーソナルサービス等、様々な応用性があるため、研究が盛んになっている。人体通信の方式の一つとして、数メガヘルツ以下の電界を人体周辺に励起させ、それにより通信を行うものがある。このように、複数のウェアラブル機器が人体を介して通信を行う際、人体周辺を一つのネットワークとして見なすことができ、そのネットワークをボディエリアネットワーク (Body Area Network : BAN) と呼ぶ。

この方式の物理チャネルを扱う場合の理論は、1995年に Zimmerman によって考案された容量結合の集中定数回路が有名であり、また、その理論における等価回路を静電界解析に基づいてさらに正確に導出するための研究も行われている [1],[2],[3],[4]。一方、FDTD (Finite-Difference Time-Domain) 法やモーメント法 (Method of Moments : MoM) などの全波動解析手法を用いたチャネル特性の解析も広く行われている。

このように、ボディエリアネットワークの解析手法は様々であるが、現実には、前述した解析手法では定義し難い、様々で複雑な要因が絡むことが想定できる。例えば、人体の複雑な姿勢変化や周辺環境の忠実な電磁界モデルを作成するのは容易ではない。そのため、この方式の物理チャネルのモデルを扱う際には、実人体を用いてチャネルの動的变化を測定し、その実測値に基づいて統計的チャネルモデルを検討するのが不可欠である。

先行研究では、2.4GHz 帯をはじめとした比較的高い周波数における統計的チャネルモデルを検討している例が多いのに対して、数十メガヘルツ程度の比較的低い周波数の統計的チャネルモデルを検討した例はわずかである。Zedong らは、最尤推定法を用いて、45MHz におけるチャネルの動特性を典型的な確率密度関数モデル (ガンマ分布、対数正規分布、仲上・ライス分布、ワイブル分布) に当てはめることを検討している [5]。しかしながら、チャネル特性をより良く表した関数モデルの可能性については検討していない。また、これまでに通信機の接地状態 (バッテリー駆動か商用電源駆動かの違い) がチャネル特性に及ぼす影響について実験的に検討された例は皆無である。

ここで、チャネル特性を測定する場合、ネットワークアナライザ等の測定器を用いることが考えられるが、この場合、送受信アンテナは高周波ケーブル及び電源ケーブルを介して大地に接地されることになる。一方、本研究では複数の接地条件のチャネル特性を検討する必要があるため、アンテナが常に接地されてしまうネットワークアナライザの使用は好ましくない。したがって、バッテリー駆動でアンテナ一体型の測定システムが必要となり、測定データの記録等もスタンドアロンで行う必要がある。

そこで本研究では、まずバッテリー駆動でデータの記録が可能な測定システムを構築し、その測定システムを用いて得た実測値を基に、ボディエリアネットワークの統計的チャネルモデルの構築を考える。本論文の構成は以下の通りである。第 2 章では、測定システムの構築について、述べる。第 3 章で測定内容と条件について述べ、第 4 章でその測定結果を示す。第 5 章では測定結果を解析し、チャネルモデルを構築する手法について述べ、第 6

章でその結果について述べる。第 7 章と第 8 章では、それぞれ結論と今後の課題を述べる。

2. 測定システムの構築

2.1. 測定方法と新システムの設計要件

第 1 章で述べた既存システムの問題点から、本研究ではまず新たな測定システムを構築する。測定は、送受信器を図 2.1.1 のように身体に装着し、送信器から単一周波数の連続波を送信し続け、そのときの受信器における受信電圧を測定する、という方法によって行う。このとき、人体は様々に姿勢を変化させるため、受信器では時間変化を記録できることが望ましい。

また、図 2.1.1 に示すように、送受信器はそれぞれ接地状態と非接地状態の両方で動作できることが求められる。これは、応用目的の中には送受信器のどちらかが据え置きのもがあり、それらが商用交流電源によって動作する、すなわち接地された状態で動作することを再現するためである。そして、本研究においては、広いダイナミックレンジを持つ受信器を開発することによって、それらの接地条件ごとの測定値を、全て同じ測定システムを用いて計測することを目指した。

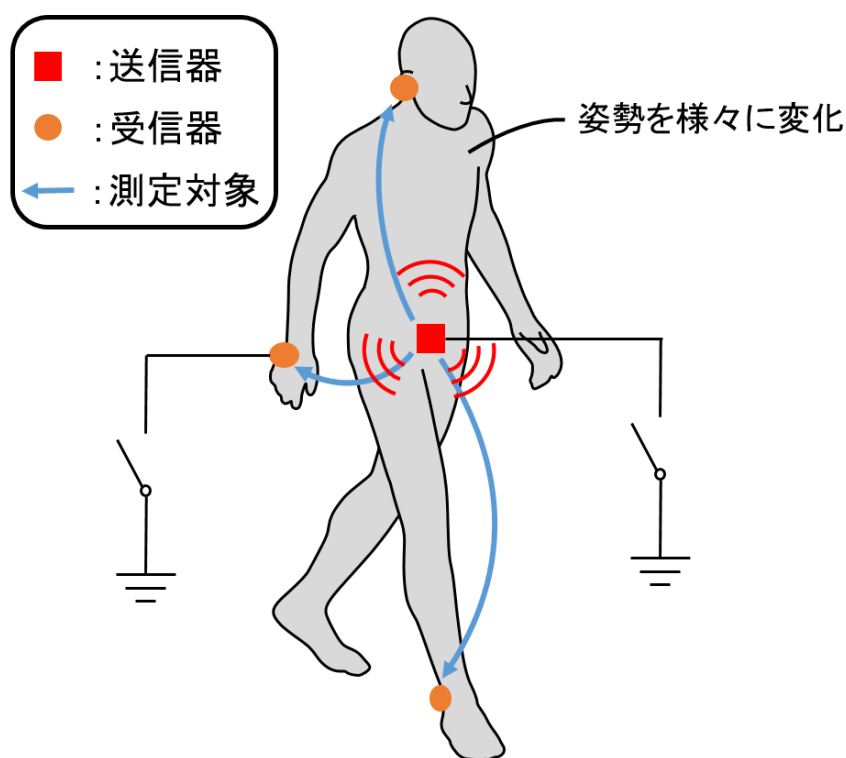


図 2.1.1 測定系イメージ

前述した設計要件をまとめると、送受信器に共通する設計要件は、

- ①身体に装着できるほど小型であること
- ②接地状態と非接地状態の両方で動作できること
- ③アンテナと一体型のシステムであること

という 3 点が挙げられる。送信器については、単一周波数の連続波を放射し続けるだけであるため、これ以上の設計要件は無い。そして受信器に関しては、これらの 3 点の他に、

④広いダイナミックレンジ (80dB 程度) を有すること

⑤時間変化を記録できること

という設計要件が加わる。これら設計要件の具体的な実現方法について検討する。

まず、①の条件は、回路をプリント基板上で設計し、回路素子を表面実装のチップ部品を用いることにより実現した。本研究においては、90mm×60mm のサイズの回路基板上で実現することを目指した。また、②の条件は、測定システムをバッテリー駆動とすることによって非接地状態における動作を実現した。そして、回路基板上に接地用の SMA コネクタを取り付け、そのコネクタから被覆銅線を介して、接地状態を実現する。そして③の条件は、測定システムに平行平板型アンテナを用い、図 2.1.2 のような構造をとることによって実現する。

また、受信器における設計要件④は、検波回路として高性能ログアンプ (Analog Devices, AD8307) を用いることによって実現させる。このログアンプは DC から 500MHz までの広い周波数帯域で、およそ 92dB のダイナミックレンジを有しているため、本研究において採用した[10]。そして条件⑤は、マイクロコントローラを用いて micro SD カードにデータを記録することによって実現させる。

これらの方法により、送受信器の設計要件を満たす。そして図 2.1.2 のようなものをそれぞれ送信器と受信器として一つずつ作製し、これらの送受信器をまとめて一つの測定システムとする。

図 2.1.2 中の「Top electrode」は回路グラウンドを兼ねている電極、「Bottom electrode」は電界入出力電極である。Top electrode 上にある検波回路 (AD8307) と、電界入出力電極である Bottom electrode は、導線によって導通している。また、測定機器を接地状態にする際は、Top electrode の回路グラウンドを、電源コンセントの保安アース端子と導通させる。なお、図中には無いが、実際には Bottom electrode の下に厚さ 5mm の発泡スチロールを貼り付け、装着した人体から電氣的に浮いた状態を実現する。

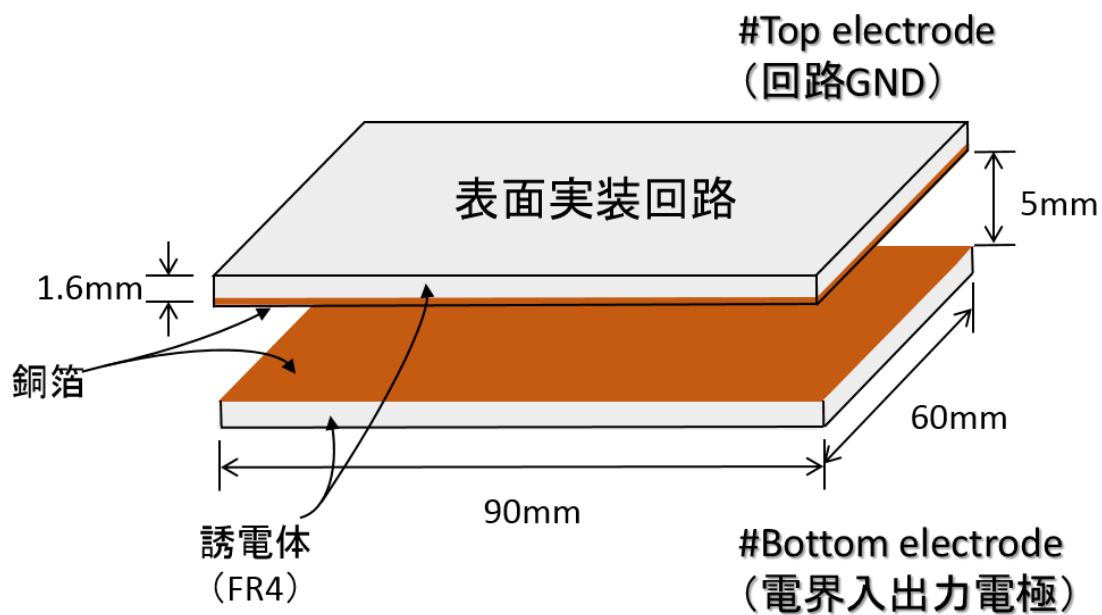


図 2.1.2 測定システムのイメージ

プリント基板の加工には、群馬大学工学部キャンパスにある高度人材育成センターの設備を利用した。また、回路のモデリングには EAGLE という CAD ソフトを用いている。EAGLE によって出力されたガーバーファイルを CircuitCAM というソフトによって変換し、そして基板加工機を動かす Boardmaster というソフトに回路のデータを入力している。この一連の流れは[7]を参考にするとよい。

2.2. 受信器の設計

2.2.1. 受信器の設計要件

まずは受信器 (Rx) について述べる。受信器の設計要件は、以下の 5 点である。

- ①身体に装着できるほど小型であること
- ②接地状態と非接地状態の両方で動作できること
- ③アンテナと一体型のシステムであること
- ④広いダイナミックレンジ (80dB 程度) を有すること
- ⑤時間変化を記録できること

これらの設計要件を満たすための軸となる部品を表 2.2.1.1 にまとめる。

表 2.2.1.1 受信器の軸となる部品一覧

役割	部品名	提供する会社
検波回路	AD8307	Analog Devices
オペアンプ	LT1490	Linear Technology
記憶媒体への記録	PIC24FJ64GA002	Microchip
記憶媒体	micro SD (2GB)	SanDisk
バッテリー (ニッケル水素電池)	IMPULSE 6P 形 (9V)	TOSHIBA

2.2.2. システムの構成 (受信器)

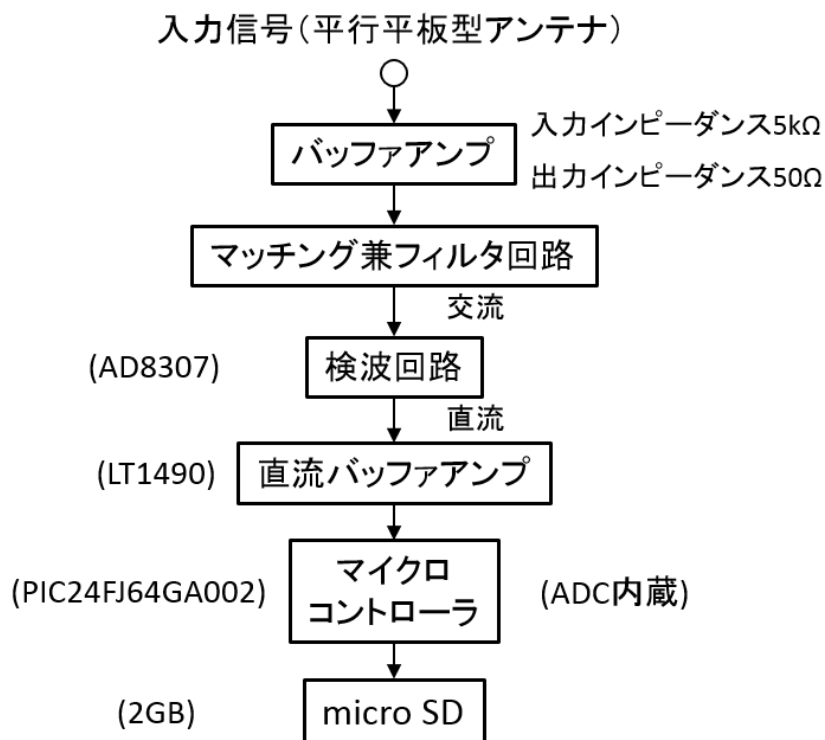


図 2.2.2.1 システムの構成 (受信器)

まず、システムの細かな設計手法の説明の前に、システム全体の機能を表すダイアグラムを図 2.2.2.1 に示した。ここでは、表 2.2.1.1 に示した部品の使用される箇所も併せて示している。

この構成をとることによって、前述した設計要件を概ね満たすことが期待される。本研究では、このシステムを一枚のプリント基板上に構成する。以降、それぞれ順を追って説明する。

2.2.3. バッファアンプ

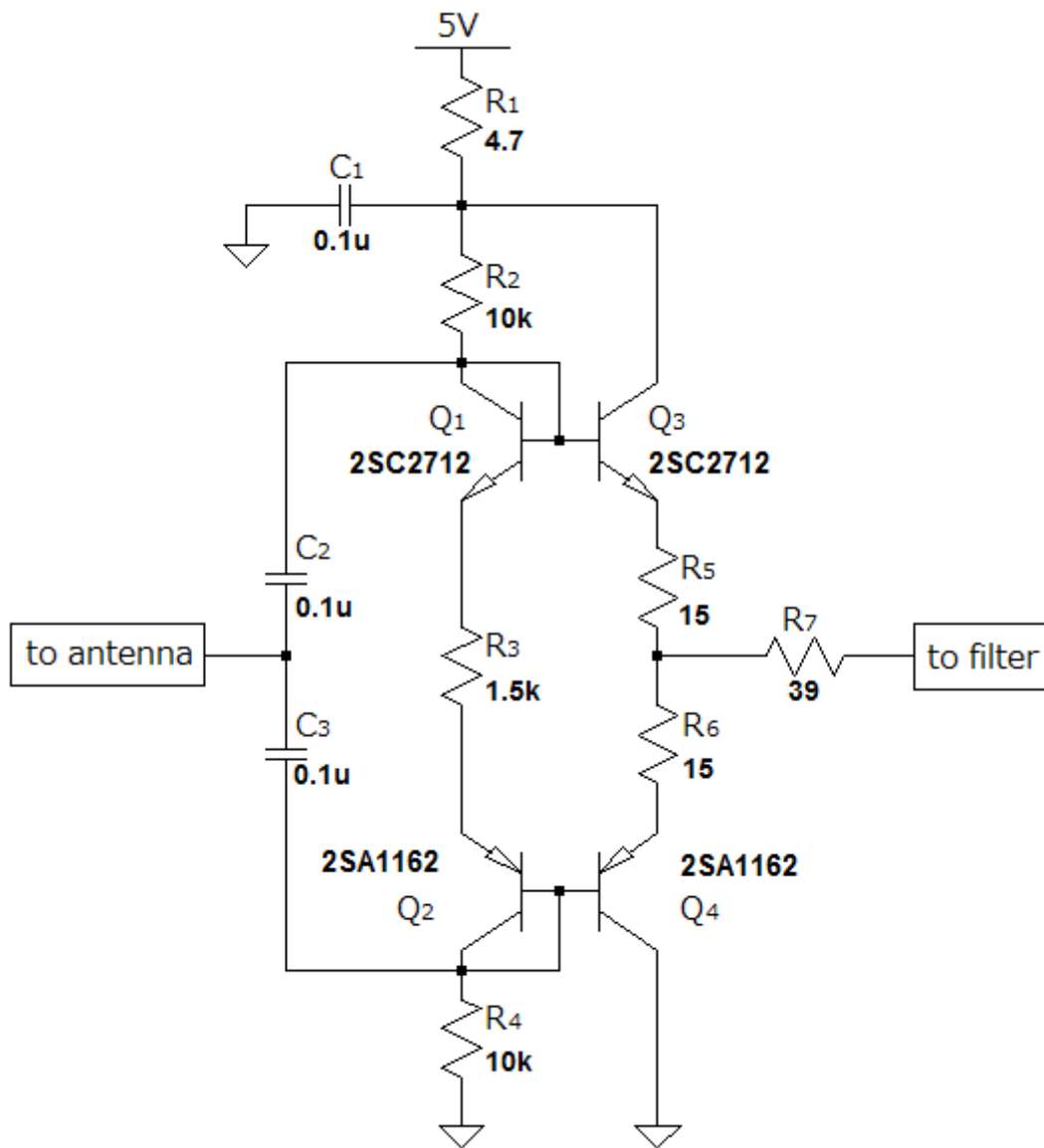


図 2.2.3.1 マッチング回路

図 2.2.3.1 は、作製した受信器に用いているバッファアンプであり、一般にはコレクタ接地増幅回路、あるいはエミッタフォロワと呼ばれるものである。この回路は、平行平板型

アンテナと、後段のフィルタとの橋渡しを目的としており、高いインピーダンスで入力し、低いインピーダンスで出力する。このバッファアンプの入力インピーダンスは R_2 および R_4 の並列接続で定義されるため、値は $5k\Omega$ である。そして、出力インピーダンスは R_5, R_6, R_7 によっておよそ 50Ω となるように設計してある。つまり、ほとんどが容量性であるアンテナの出力インピーダンスが、この回路を通すことによって 50Ω の出力インピーダンスとなる。

また、図 2.2.3.1 のバッファアンプは、 Q_3 と Q_4 が npn と pnp によるプッシュプル構成になっており、 Q_1 と Q_2 はダイオード接続として用いている（カレントミラー回路）。これによって Q_1 と Q_2 のバイアス電流値を定めている。このバイアス電流は、以下の近似計算により、大まかに見積もることができる。

R_1 による電圧降下と Q_3, Q_4 のベース電流を無視し、ダイオード接続された Q_1, Q_2 の電圧をそれぞれ $0.6V$ とすると、 R_2, R_3, R_4 に流れる電流は、 $(5 - 0.6 - 0.6)/(10k + 1.5k + 10k) \approx 0.28mA$ となる。 R_3 にかかる電圧と R_5, R_6 にかかる電圧が等しいと近似すると、 R_5, R_6 に流れる電流は $0.28mA \times 1.5k/(15 + 15) = 14mA$ となる。

2.2.4. インピーダンス整合兼フィルタ回路

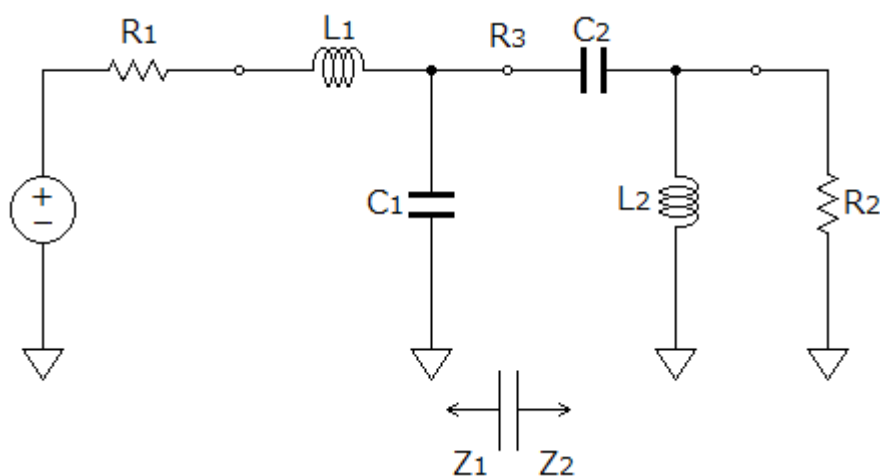


図 2.2.4.1 Low-High フィルタ回路

前述のように、バッファアンプの出力インピーダンスは 50Ω である。一方、後述するように、検波回路の入力インピーダンスは $1.1k\Omega$ であるので、受信感度の観点から、その間にインピーダンス整合回路を設けるのが望ましい。また、耐ノイズ性の観点から、使用周波数である $10MHz$ のみを通過させるようなバンドパスフィルタとしての機能も持ち合わせることを望ましい。

まず、図 2.2.4.1 のように、バンドパスフィルタとして、ローパスフィルタとハイパスフィルタを組み合わせた回路を考える。ここで、 R_1 は前述したバッファアンプの出力インピーダンス 50Ω であり、 R_2 は次段の検波回路の入力インピーダンスである。 L_1 と C_1 により形

成されるローパスフィルタは、 R_1 と仮想的な中間インピーダンス R_3 を整合させる役割を担っている。一方、 L_2 と C_2 により形成されるハイパスフィルタは、 R_3 と R_2 を整合させる役割を担っている。インピーダンス整合が達成されている状態では、ローパスフィルタとハイパスフィルタの接続点から左側を見たインピーダンス Z_1 と、右側を見たインピーダンス Z_2 は、共に中間インピーダンス R_3 と等しくなるはずである。この条件を数式で表すと、それぞれ以下のようなになる。

$$Z_1 = \frac{1}{j\omega C_1 + \frac{1}{j\omega L_1 + R_1}} = \frac{j\omega L_1 + R_1}{1 + j\omega C_1 R_1 - \omega^2 L_1 C_1} = R_3 \quad (2.2.4.1)$$

$$Z_2 = \frac{1}{j\omega C_2} + \frac{1}{\frac{1}{j\omega L_2} + \frac{1}{R_2}} = \frac{R_2 + j\omega L_2 - \omega^2 L_2 C_2 R_2}{j\omega C_2 R_2 - \omega^2 L_2 C_2} = R_3 \quad (2.2.4.2)$$

となる条件式が成り立つ。まず、式(2.2.4.1)について整理すると、

$$R_3 - \omega^2 L_1 C_1 R_3 - R_1 + j(\omega C_1 R_1 R_3 - \omega L_1) = 0$$

となる。これが恒等的に成り立つため、

$$\begin{cases} R_3 - \omega^2 L_1 C_1 R_3 - R_1 = 0 & (2.2.4.3) \\ \omega C_1 R_1 R_3 - \omega L_1 = 0 & (2.2.4.4) \end{cases}$$

という条件式が導かれる。これを L_1 および C_1 について解くと、

$$L_1 = \frac{\sqrt{R_1(R_3 - R_1)}}{\omega} \quad (2.2.4.5)$$

$$C_1 = \frac{1}{\omega R_3} \sqrt{\frac{R_3 - R_1}{R_1}} \quad (2.2.4.6)$$

となる。

同様に、式(2.2.4.2)について整理すると、

$$R_2 - \omega^2 L_2 C_2 R_2 + \omega^2 L_2 C_2 R_3 + j(\omega L_2 - \omega C_2 R_2 R_3) = 0$$

となり、恒等的に0となるので、

$$\begin{cases} R_2 - \omega^2 L_2 C_2 R_2 + \omega^2 L_2 C_2 R_3 = 0 & (2.2.4.7) \\ \omega L_2 - \omega C_2 R_2 R_3 = 0 & (2.2.4.8) \end{cases}$$

という式が得られる。これを L_2 および C_2 について解くと、

$$L_2 = \frac{R_2}{\omega} \sqrt{\frac{R_3}{R_2 - R_3}} \quad (2.2.4.9)$$

$$C_2 = \frac{1}{\omega \sqrt{R_3(R_2 - R_3)}} \quad (2.2.4.10)$$

と求めることができる。よって、目標とする各周波数 ω と中間インピーダンス R_3 の値さえ決

定してしまえば、これらの式(2.2.4.5), (2.2.4.6), (2.2.4.9), (2.2.4.10)によって、 L_1, C_1 および L_2, C_2 が一意に定まる。

R_3 は前述した通り、ローパスフィルタとハイパスフィルタを設計する上で導入された中間のインピーダンスである。この値を、例えば R_1 と R_2 の相乗平均 $\sqrt{R_1 R_2}$ とすれば、ローパスフィルタとハイパスフィルタの効き目がほぼ同じとなるようなバンドパスフィルタとなる。この相乗平均値を軸に、LやCの実際に存在する値（E6系列）を考慮に入れつつ、一番良い特性を持つ組み合わせを選んだ。

ここまでで述べた手順によって選んだ素子値によって得られる出力電圧の周波数特性は、次頁の図 2.2.4.2 の通りである。なお、解析にはLTspiceを使用しており、入力電圧は、振幅 1V の正弦波としている。図 2.2.4.2 の実線は出力電圧の振幅を表し、破線がその位相を表している。

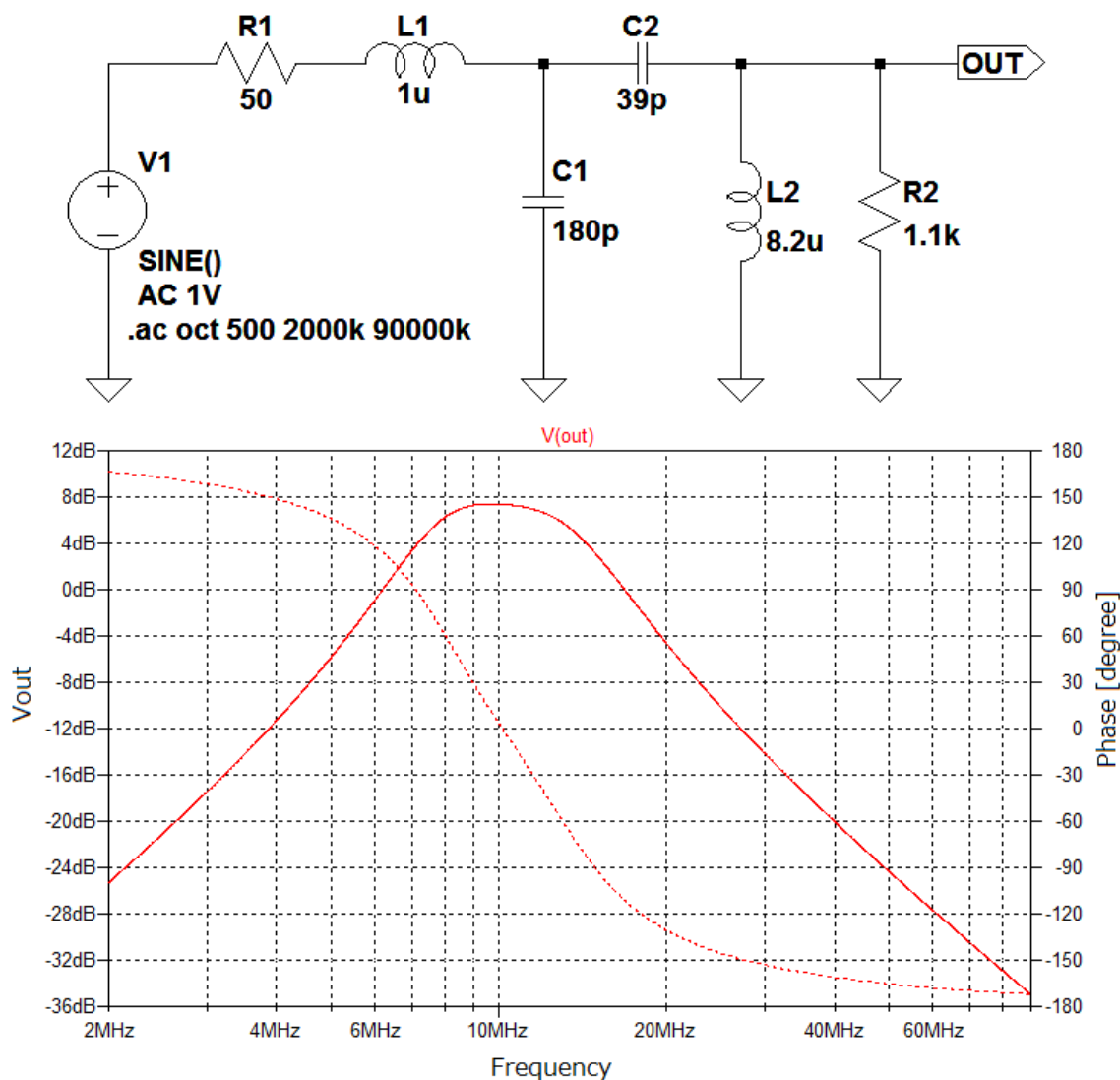


図 2.2.4.2 Low-High フィルタ回路とその周波数特性

図 2.2.4.2 を見ると分かるように、目標周波数である 10MHz 付近では、インピーダンス整合により出力電圧が最大となっている。しかしながら、10MHz 付近の周波数変化は比較的広帯域な特性となっているため、不要な周波数の信号まで拾ってしまいかねない。そこで、この回路に新たに狭帯域な並列回路を付加することを検討する。

共振回路を付加する部分としては色々考えられるが、ここでは図 2.2.4.3 のように、出力部の R_2 に対して並列に、 L_3, C_3 による並列共振回路を付加する案を採用する。この並列共振回路の帯域が L_1, L_2, C_1, C_2 により構成されるバンドパスフィルタのそれと比較して非常に狭いとすると、回路の Q 値は R_2, L_3, C_3 のみによって次式で近似できる。

$$Q = R_2 \sqrt{\frac{C_3}{L_3}} \quad (2.2.4.11)$$

と表すことができる。また、 $\omega_0 = 10\text{MHz}$ で共振させるためには

$$\omega_0^2 = \frac{1}{L_3 C_3} \quad (2.2.4.12)$$

という条件が成り立つ。ここで、 L_3 を優先的に決めたいので、式変形をすると、

$$C_3 = \frac{1}{\omega_0^2 L_3} = \frac{1}{(2\pi \times 10^7)^2 L_3} \quad (2.2.4.13)$$

となる。

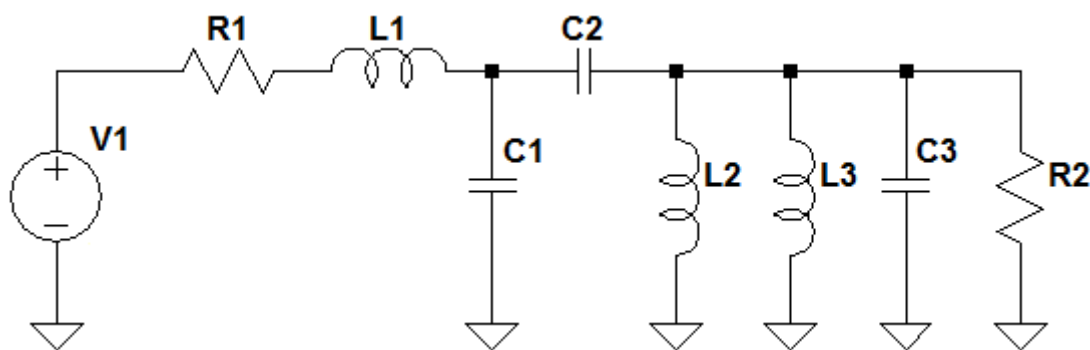


図 2.2.4.3 Low-High フィルタ+並列共振回路

この図 2.2.4.3 において、式(2.2.4.11)より、 $L_3 \rightarrow$ 小、 $C_3 \rightarrow$ 大とすれば、この回路の Q 値は高くなると予想できる。また、図 2.2.4.3 において L_2 と L_3 は並列の並びになっており、合成インピーダンス L_4 として表すことができるため、実際に新たに増える部品は C_3 のみである。値を決定した後の回路図およびその周波数特性を次に示す。

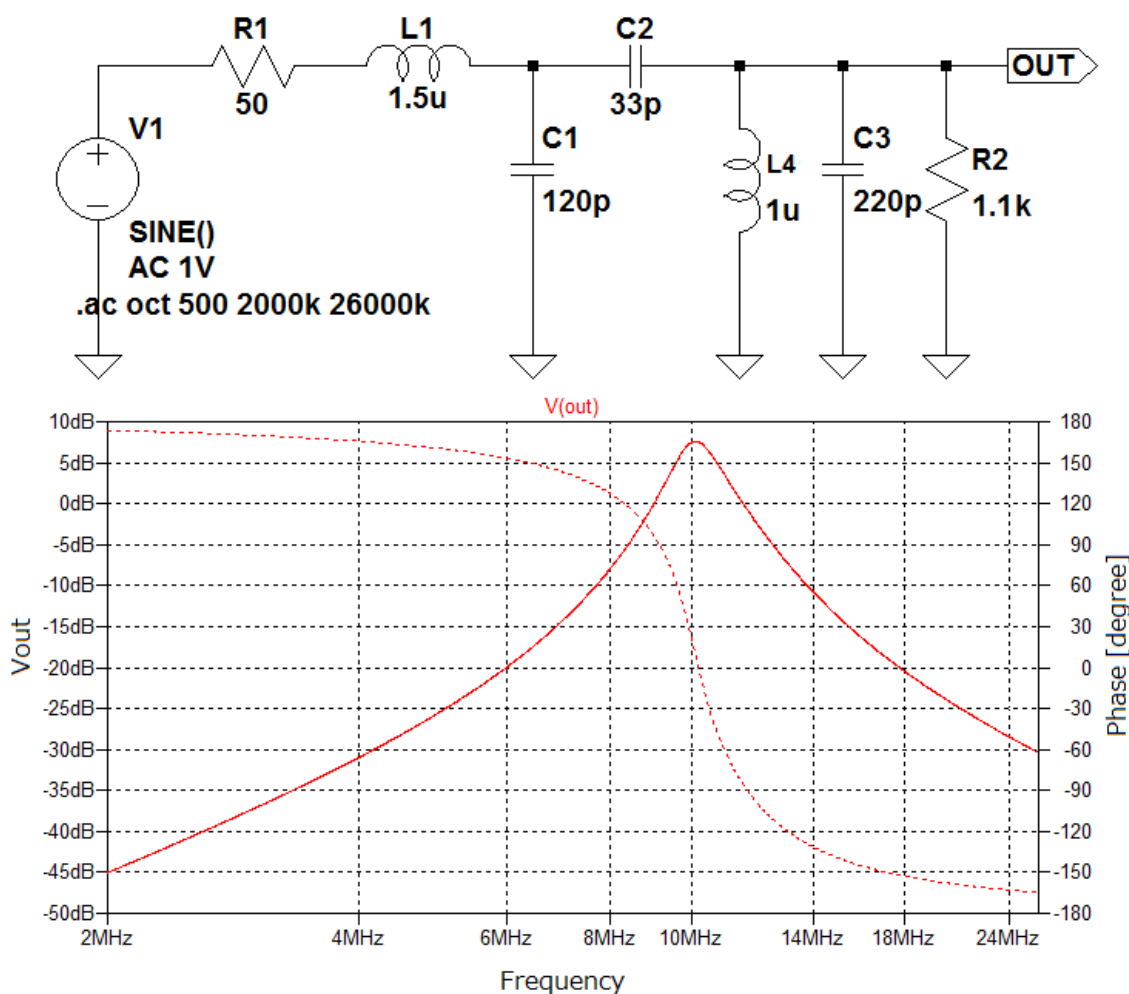


図 2.2.4.4 受信器に使用するバンドパスフィルタとその周波数特性

図 2.2.4.4 の周波数特性を見ると、図 2.2.4.2 の回路構成のときより Q 値が高くなっていることが分かる。よって、本研究における受信器では、この回路構成を用いる。

2.2.5. 検波回路 (AD8307) を用いる

本研究で用いた検波回路は、Analog Devices Components から提供されている AD8307 という 8 ピンの IC である。この AD8307 は復調型ログアンプ (包絡線検波) で、およそ 90dB のダイナミックレンジが、100MHz 以下の任意の周波数で実現できる。そして 1dB 当たり 25mV の電圧が出力される。入力は完全な差動型で、入力インピーダンスが 1.1k Ω 、出力インピーダンスが 12.5k Ω 、確度は ± 1 dB といった構造である。

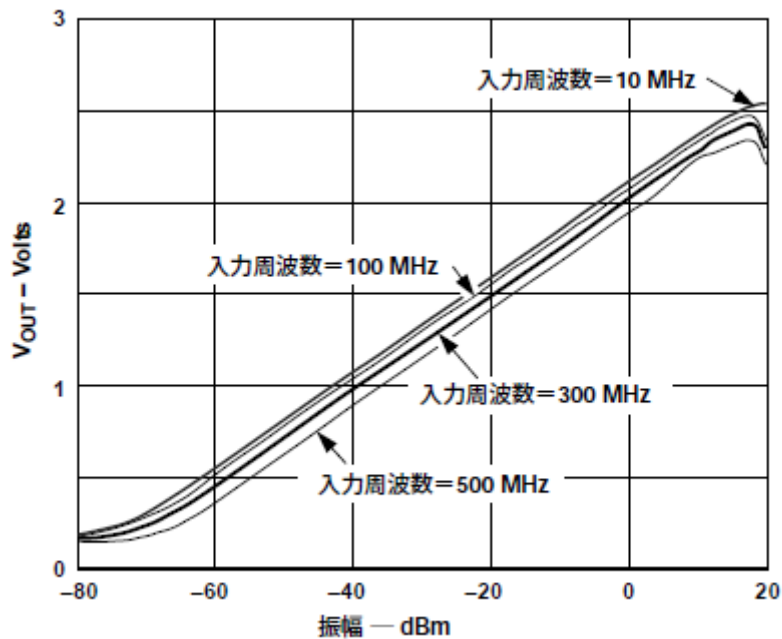


図 2.2.5.1 各周波数における V_{OUT} と入力レベルの関係[10]

図 2.2.5.1 はデータシートに掲載されている出力特性の一部である。この図を見ると分かるように、AD8307 は $0\sim 2.5V$ の出力範囲を持っている。また、AD8307 は外付けの回路をあまり必要としない特徴があるが、あえて回路を外付けすることによって、図 2.2.5.1 における直線の傾きを調整したり、オフセットを調整したりすることができる。

本研究においては、前述したパラメータの調整による誤差が生じる恐れがあるので、特別に外付けの回路を用意はしない。しかしながら、これは後述するが、PIC 内蔵の ADC のリファレンス電圧を $3.3V$ とするため、AD8307 の出力を $0\sim 2.5V$ から $0\sim 3.3V$ に増幅する回路を次の段で用意する。

AD8307 を用いる際の配線は、図 2.2.5.1 のようである。1 ピンが反転入力、8 ピンが非反転入力となっているが、本研究ではシングルエンドの動作をさせるために、反転入力はバイパスコンデンサを介して交流的にグラウンドに落とす構成としている。なお、この抵抗 $R1$ と並列に挿入されているコンデンサはローパスフィルタを形成しており、電源ラインのノイズを除去するための構成として一般的である。

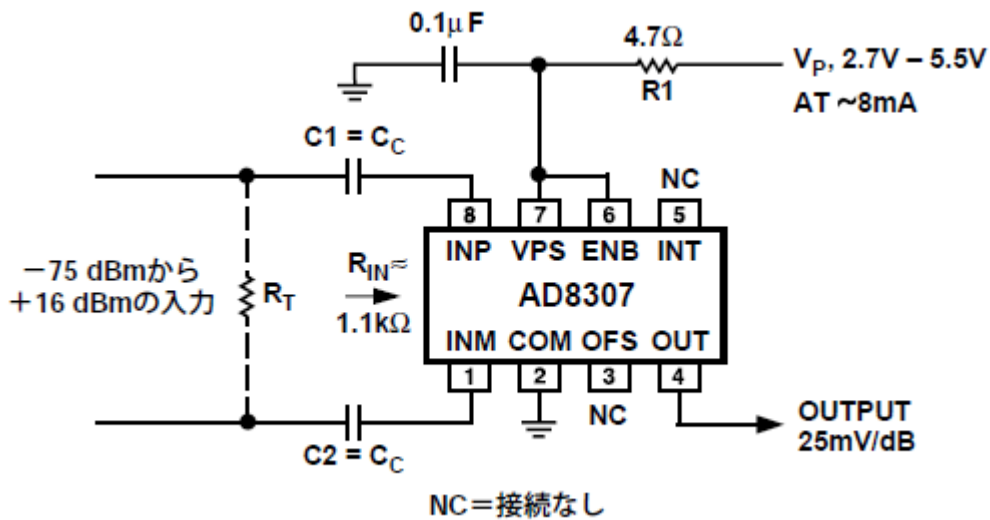


図 2.2.5.1 基本的な接続 (データシート[10]より)

2.2.6. 直流バッファアンプ

検波回路である AD8307 の出力インピーダンスは $12.5\text{k}\Omega$ と比較的高いため、出力信号をマイクロコントローラの AD コンバータに入力する際に、一度バッファアンプを通すことが必要となる。また、検波回路が出力する直流電圧の範囲が概ね $0\sim 2.5\text{V}$ であるのに対して、本研究では AD コンバータの入力電圧範囲を $0\sim 3.3\text{V}$ としているのので、それに合わせて信号を増幅することが望ましい。そこで、本研究ではオペアンプにより正相増幅器をバッファアンプとして用いた。

本研究で用いたオペアンプは、Linear Technology Devices から提供されている LT1490 という 8 ピンの IC である。LT1490 は $2\sim 44\text{V}$ の単一電源および両電源で動作し、静止電流は 1 アンプあたりわずか $40\mu\text{A}$ であるという特徴を持った、マイクロパワー・レール・トゥ・レールオペアンプである。内蔵のオペアンプは 2 つで、ピンアサインは図 2.2.6.1 のようである。

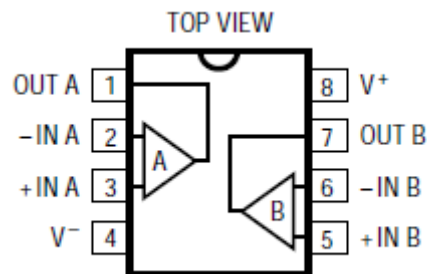


図 2.2.6.1 LT1490 のピンアサイン (データシート[11]より)

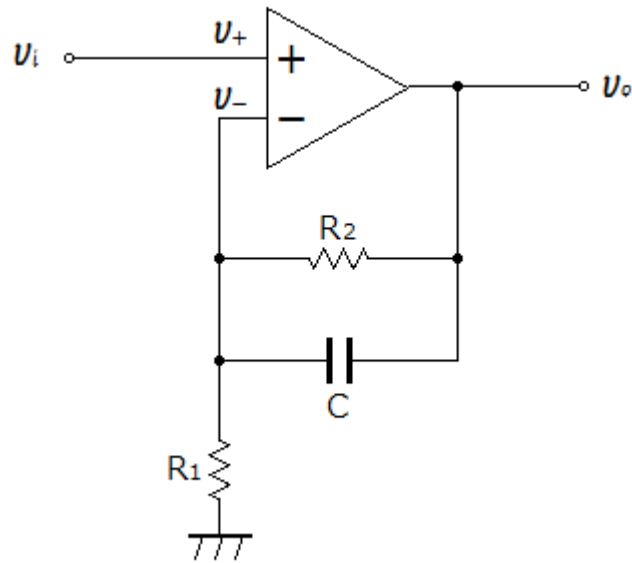


図 2.2.6.2 正相増幅器

本研究では、オペアンプを正相増幅器として用いるため、図 2.2.6.2 のように回路を組んだ。このときキャパシタ C はオペアンプの発振を防ぐために挿入している。利得は R_1 と R_2 によって以下のように求めることができる。

$$\text{利得} = \frac{v_o}{v_i} = 1 + \frac{R_2}{R_1} \quad (2.2.6.1)$$

本研究では、AD8307 の出力範囲 $0 \sim 2.5V$ を、 $0 \sim 3.3V$ の出力範囲へ広げる目的で設計するので、このことから利得は以下のように求められる。

$$\text{利得} = \frac{3.3}{2.5} = 1.32 \quad (2.2.6.2)$$

よって、 R_1 と R_2 は次のような条件にする必要がある。

$$\frac{R_2}{R_1} = 0.32 \quad (2.2.6.3)$$

この条件を一番良く満たす実際の抵抗値の組み合わせとして、 $R_1 = 47k\Omega$, $R_2 = 15k\Omega$ を選んだ。

2.2.7. マイクロコントローラの選定とピンアサイン

本研究では、SD カードへデータを記録する際、マイクロコントローラに PIC (Peripheral Interface Controller) マイコンと呼ばれる種類のものを使用した。

PIC には様々な種類があり、それぞれの種類を大きくわけてファミリーと呼ぶ。PIC は Microchip 社が随時研究開発しており、大きなファミリーのものほど多くの周辺機能を内蔵している。ファミリーには、ベースラインファミリー (PIC10F, PIC12F)、ミッドレンジファミリー (PIC16F)、ハイエンドファミリー (PIC18F)、MCU ファミリー (PIC24F, PIC24H)、DSC ファミリー (dsPIC30F, dsPIC33F)、PIC32MX ファミリー (PIC32MX) という種類がある。本研究ではこの中の MCU ファミリーである PIC24F シリーズを用いた。

PIC24F シリーズは、コストパフォーマンスが高い 16 ビットマイコンである。16 ビットマイコンの中では下位クラスであるが、本研究のように、本格的な用途で用いない場合においては十分な性能および機能を保有し、コストパフォーマンスの高さを最大限に発揮できる。

この PIC24F シリーズの中から、本研究では PIC24FJ64GA002 という型番のものを使用した。これは PIC24 シリーズでも割とオーソドックスな型の「PIC24FJxxGA002」シリーズの内の一つである。内蔵している機能や構成の仕様を表 2.2.7.1 に示す。

表 2.2.7.1 PIC24FJ64GA002 の主な仕様

項目	仕様
RAM 容量	8kB
ピン数	28 ピン
プログラムメモリ容量	64kB
プログラムメモリタイプ	フラッシュ
データバス幅	16 ビット
ADC チャンネル数	10 チャンネル
ADC 分解能	10 ビット
ADC ユニット数	1 つ
USB ユニット数	なし
タイマ数	5 つ
最大システムクロック周波数	32MHz
標準動作供給電圧	2~3.6V
最大 SPI チャンネル数	2 チャンネル

本研究のように、PIC マイコンが SD カードと通信を行う際、PIC マイコンに書き込むプログラムがやや長くなるため、RAM 容量は 4kB 以上であるのが好ましい。本研究では余裕を持って 8kB であるものを選んだ。

また、PIC24Fシリーズに内蔵されているADコンバータは、逐次比較近似タイプであり、分解能は10ビット（0～1023）である。このとき、AD変換のためのリファレンス電圧は、3.3Vのシリアズレギュレータによって供給されている。この3.3Vという電圧は、SDカードおよびPIC24FJ64GA002を動作させる電源電圧にも利用している。

タイマ機能は、ADコンバータでAD変換をする際のサンプリング間隔を決めるときに用いる。

また、SDカードとの通信は、SPI（Serial Peripheral Interface）通信という通信方式を用いる。

PICを動作させる際のシステムクロックの周波数は、PICに内蔵されている8MHzの振動子を源振として、2MHzを作り出し、それをシステムクロック周波数として用いた。また、SDカードと通信を行う際もクロックが必要になるのだが、その周波数は1MHz（システムクロック周波数の半分）としてある。

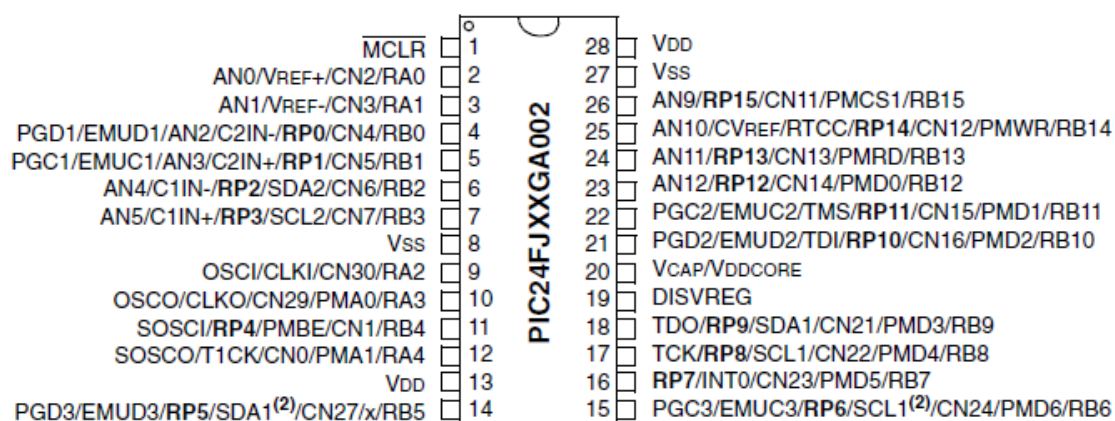


図 2.2.7.1 PIC24FJxxGA002 シリーズのピンアサイン

PIC24FJxxGA002 シリーズのピンアサインは図 2.2.7.1 の通りである。

また、筆者は統合開発環境として MPLAB X (ver 1.85) を使用しており、PC と PIC との接続には Pickit3 を使用した。PIC24FJ64GA002 は DIP (Dual In-line Package) のものと表面実装のものと両方あり、本研究では表面実装のものを使用している。そのため、プログラムを Pickit3 で書き込む際は、表面実装部品を DIP 基板にさせるようにする変換器を別途で使用している。

2.2.8. micro SD カードのピンアサイン

本研究では、測定データを micro SD に蓄積することによって、時間的な記録を残す。このとき、マイコンと micro SD との通信は、SPI 通信を用いて行う。SPI 通信とは、Serial Peripheral Interface の略で、3本の少ない配線で数 Mbps の通信を可能にする通信方式である。SD カードには SD カード独自の通信方式のほかに SPI 通信方式にも対応しており、

また、PIC も SPI 通信のモジュールが搭載されている。

micro SD カード自体は 8 ピンで構成されており、micro SD カードスロットに CD (Card Detect) ピンが 2 ピン搭載されていて、図 2.2.8.1 のように計 10 ピンが見えている。このとき、CD ピンとは⑨ピンと⑩ピンの「カード検出スイッチ A, B」のことであり、②の CD とは別の存在である。本研究ではカード検出スイッチ B を GND に落とし、カード検出スイッチ A を CD ピンとして扱う。

なお、図 2.2.8.1 は micro SD カードを DIP 基板で使用できるように変換したものであり、本研究ではこの図の DIP に変換していない素の状態のものを使用した。ピンアサインは表面実装のものも、DIP に変換したものも等価である。

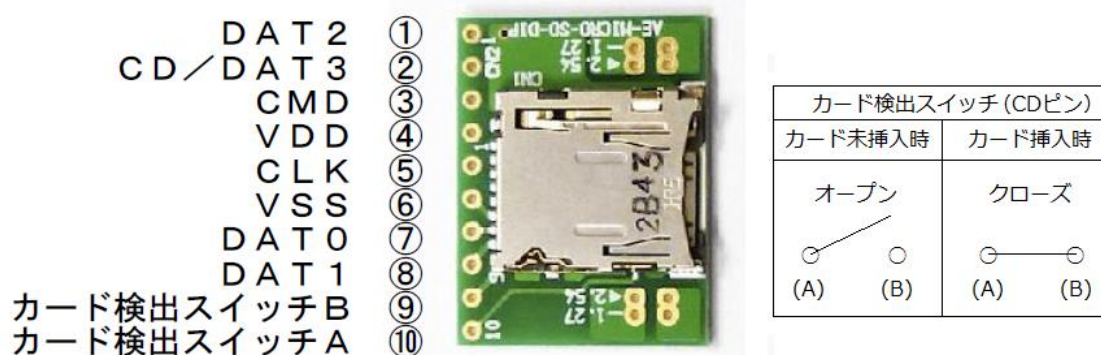


図 2.2.8.1 micro SD カードのピンアサイン

図 2.2.8.1 は micro SD カードのピンアサインであるが、これらのピンアサインは SD カード独自の通信方式 (SD モードと呼ぶ) を用いる際のピン名称で、SPI 通信を使うときはピンの名称および使用するピンが変わってくる。その対応を表 2.2.8.1 に示す。

表 2.2.8.1 micro SD カードのピンアサイン対応表

ピン番号	SD モード	SPI モード
①	DAT2	使わない
②	CD / DAT3	CS
③	CMD	SDI
④	VDD	VDD
⑤	CLK	SCK
⑥	VSS	VSS
⑦	DAT0	SDO
⑧	DAT1	使わない
⑨	カード検出スイッチ B	GND
⑩	カード検出スイッチ A	CD

SPI 通信方式において、重要な役割を果たすのはこの中の SDI , SDO , SCK の 3 つのピンである。また、SD カード全般の電源電圧は 3.3V であるので、VDD は 3.3V としている。

2.2.9. 受信器のプログラム

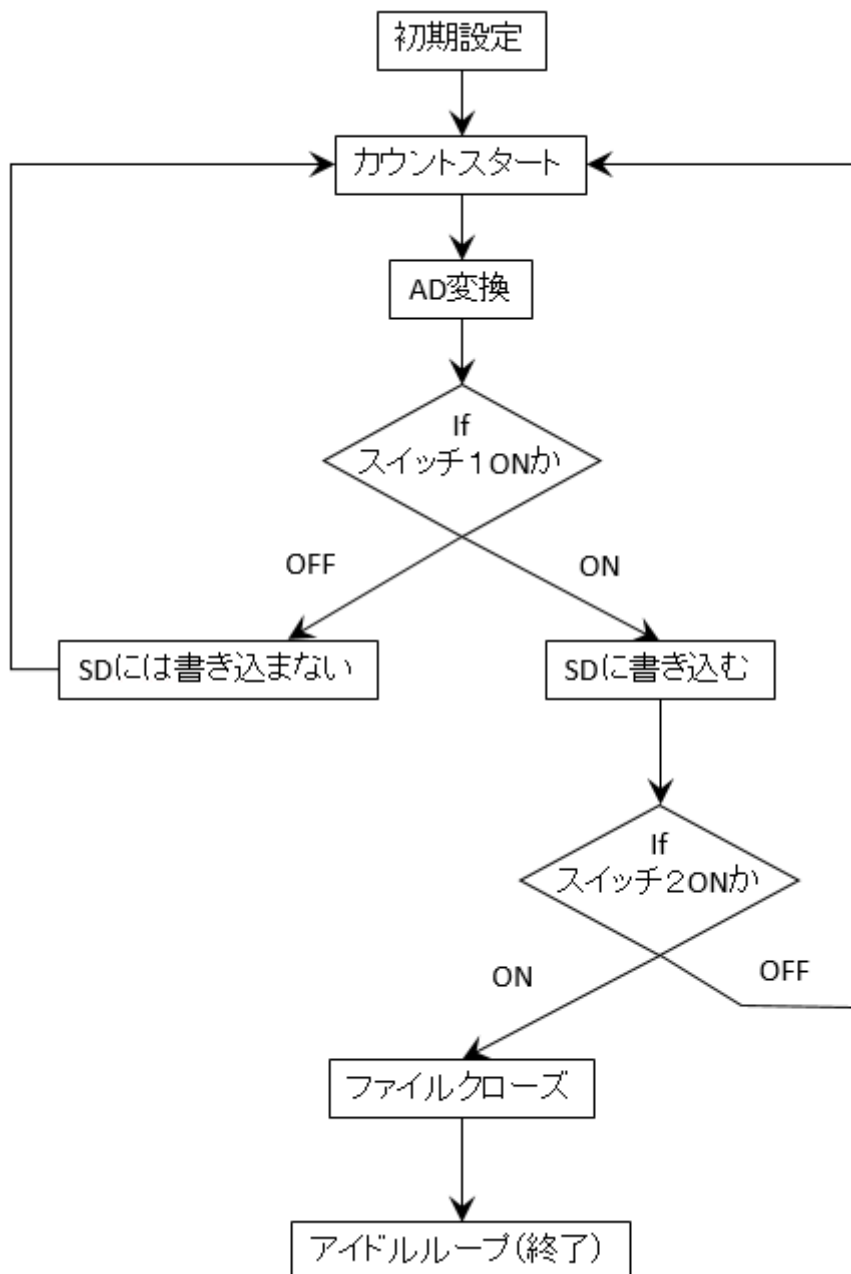


図 2.2.9.1 受信器プログラムフローチャート

受信器の測定プログラムは図 2.2.9.1 のようである。このプログラムを実現するために使用する PIC のモジュールは、タイマ、AD コンバータ、プルアップ抵抗、SPI 通信である。これらについては、[6]や[8]を参照するとよい。

また、PIC 内蔵のモジュール以外に、micro SD への書き込み開始スイッチおよび書き込み停止スイッチを 2 つ用意する。そして、これらの動作を外から分かるように LED を 1 つ設けている。

micro SD と PIC を通信させるためには、別途ライブラリが必要であり、その中身を改編する必要がある。この作業については[6]を参照すると良い。このことを踏まえ、実際に筆者が受信器に使用しているプログラムのメインファイルの内容を末尾に付録として添付する。

2.2.10. 受信器の回路図と使用部品

次頁図 2.2.10.1 に受信器の完成回路図を掲載する。また、この回路図において使用した部品を以下の表 2.2.10.1 にまとめる。

表 2.2.10.1 受信器回路に使用した部品一覧

部品	型番
IC1	PIC24FJ64GA002
IC2	AD8307
IC3	LT1490
SD Card socket 1	DM3AT-SF-PEJM5
TR1, TR3	2SC2712-Y(F)
TR2, TR4	2SA1162-Y(F)
REGULATOR1	L78L05ABUTR
REGULATOR2	L78L33ABUTR
V1	BH-9V-1P
SW1~3	SS12SDH2
D1, D2	DA2J10700L
LED1	LNJ237W82RA
L1	B82498B1152J
L2	B82498B1102J
C14, C16, C19, C22	F931A106MA
C18, C20	GRM188B11C334KA01D
その他 C	THN-GRM18RY
R 全般	THN-RK1608J

また、SMA コネクタが校正と接地用に 1 つずつ取り付けられている。

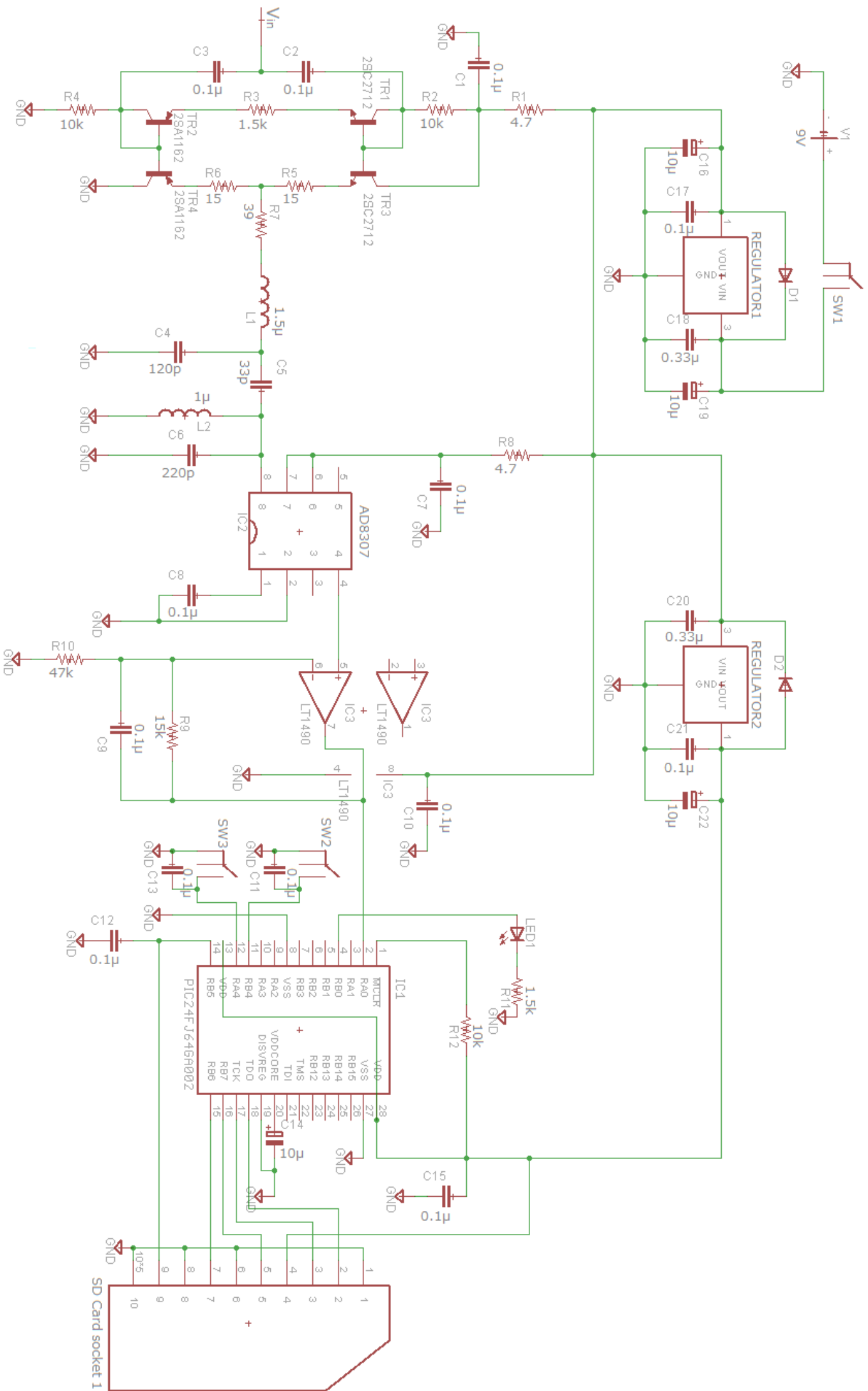


图 2.2.10.1 受信器回路图

2.2.11. 受信器の完成と動作スペック

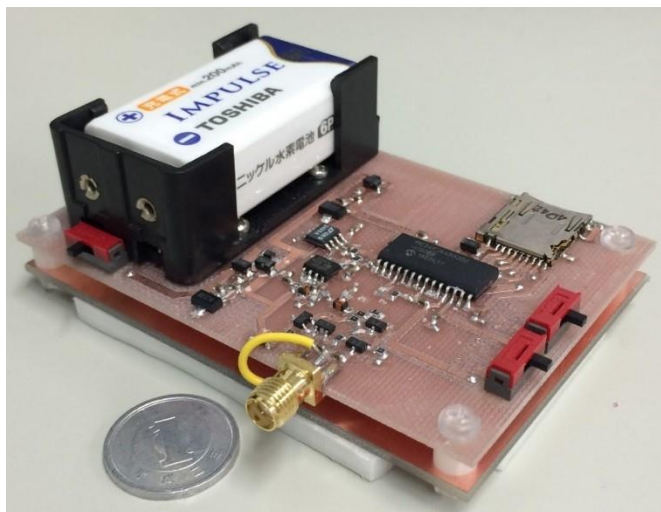


図 2.2.11.1 受信器外観

図 2.2.11.1 は完成した受信器の外観である。以下にこの受信器の動作スペックを示す。また、稼働中の LED は表 2.2.11.2 のように点灯と消灯する。

表 2.2.11.1 受信器の動作スペック

項目	動作スペック
測定上限	約 -3dBV
測定下限 (ノイズフロア)	約 -90dBV
ダイナミックレンジ	約 87dB
刻み幅	0.095dB
確度	$\pm 1\text{dB}$
サンプリングレート	50ms (1 秒間に 20 データ)
消費電流	約 20mA
連続稼働可能時間	約 10 時間
温度依存性	ほぼなし
入力インピーダンス	$5\text{k}\Omega$

表 2.2.11.2 LED の点灯、消灯プログラム

動作	LED の状態
電源が入り PIC の初期化完了、 micro SD の挿入待ち	点灯
micro SD 挿入し初期化中	消灯
micro SD 初期化完了、測定開始待ち	点灯
START スイッチ(SW2)ON、測定中	消灯
STOP スイッチ(SW3)ON、測定終了	点滅を繰り返す

2.3. 送信器の設計

2.3.1. 送信器の設計要件

次に送信器 (Tx) について述べる。送信器の設計要件は、以下の 3 点である。

- ① 身体に装着できるほど小型であること
- ② 接地状態と非接地状態の両方で動作できること
- ③ アンテナと一体型のシステムであること

これらの設計要件は受信器と同様であり 2.4 節において説明した。送信器に関する設計要件はこの他に、「周波数 10MHz の正弦波を発振し続けること」ということさえ満たしていれば良い。この設計要件は、10MHz の水晶振動子を用いた正弦波を生成する発振回路を組むことによって実現した。本研究では、この発振回路における正弦波の振幅が、8V_{p-p} となるように設計した。

2.3.2. システムの構成 (送信器)

送信器は受信器ほど複雑ではない。10MHz の水晶振動子を利用した正弦波発振回路を組み、それを受信器でも使用した 2.2.3 項のバッファアンプを通して、平行平板アンテナから放射される、という構成である。

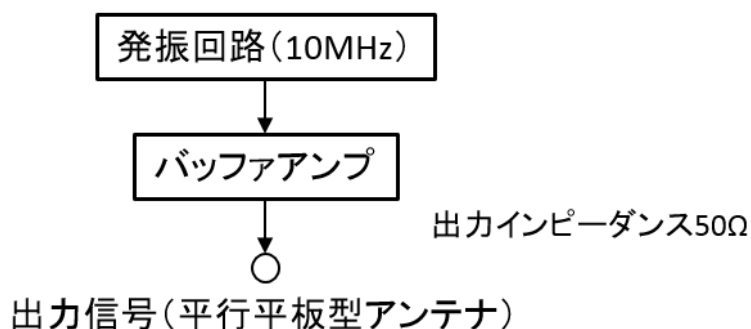


図 2.3.2.1 システム構成 (送信器)

2.3.3. 送信器の回路図と使用部品

送信器の回路は、あえて細かく分けて説明する必要性もないため、先に全体の回路図を次頁図 2.3.3.1 に示す。その後、使用した部品を表 2.3.3.1 にまとめる。

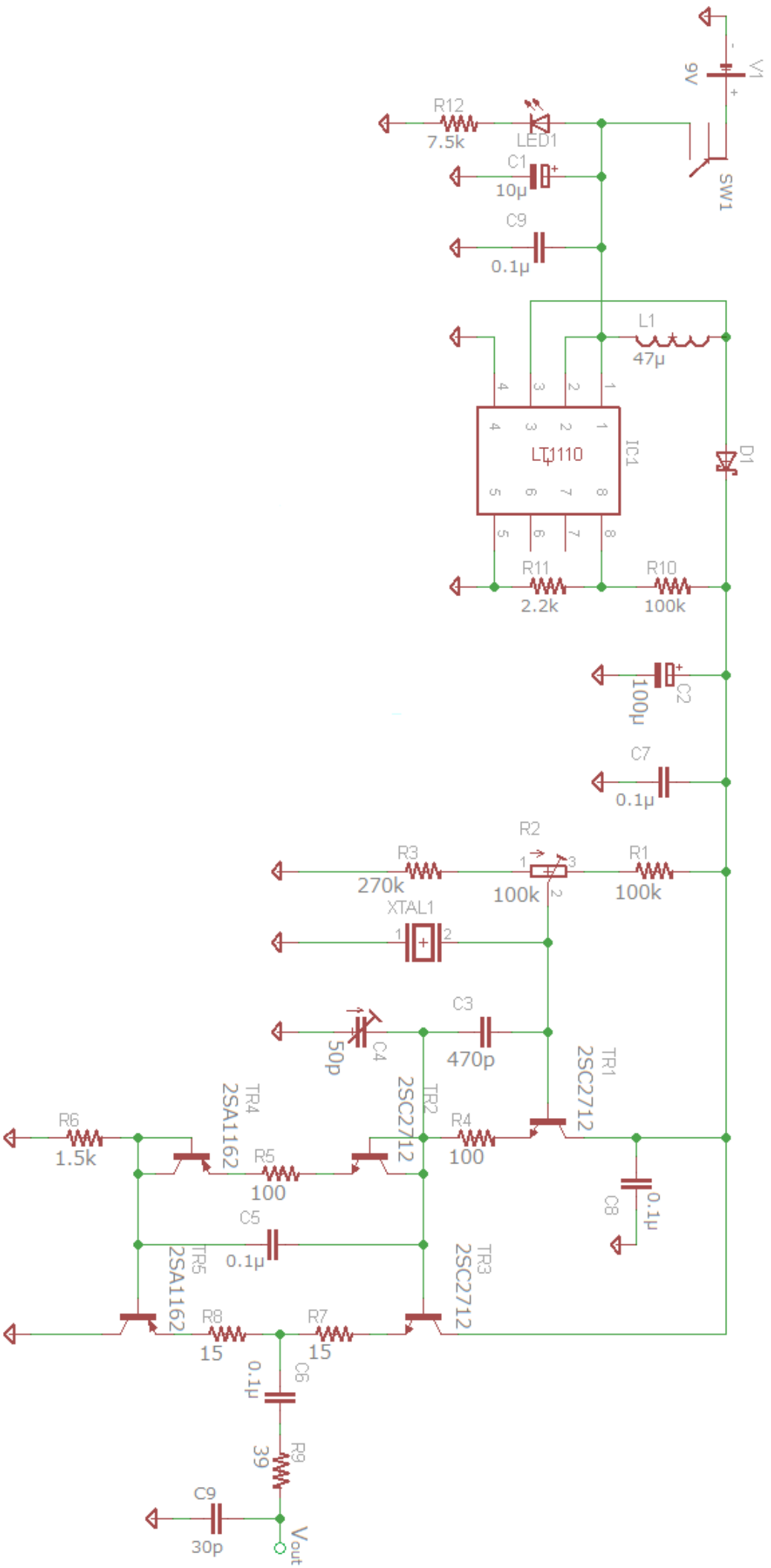


图 2.3.3.1 送信器回路图

表 2.3.3.1 送信器回路に使用した部品一覧

部品	型番
IC1	LT1110
TR1, TR2, TR3	2SC2712-Y(F)
TR4, TR5	2SA1162-Y(F)
XTAL1	FOXSDLF/100-20
V1	BH-9V-1P
SW1	SS12SDH2
D1	DB2X20700L
LED1	LNJ237W82RA
L1	SRR7045-470M
C1	F931A106MA
C2	16SVPC100M
C4	TZB4R500BB10R00
その他 C	THN-GRM18RY
R2	SM-42X 100k Ohm
その他 R	THN-RK1608J

また、校正用と接地ケーブル接続用に SMA コネクタが一つずつ取り付けられている。**R2** によって発振回路の動作点の調節を行い、**C4** によって発振する正弦波波形の振幅を調整することができる。**IC1** の「LT1110」は DC-DC コンバータであり、発振回路の電源電圧を生成している。この出力は外付け抵抗により約 10V となるようにしてある。

2.3.4. 送信器の完成と動作スペック



図 2.3.4.1 送信器外観

表 2.3.4.1 送信器の動作スペック

項目	動作スペック
消費電流	約 14mA
連続稼働可能時間	約 14 時間
温度耐性	±3%ほど
発振振幅調整可能範囲	4Vp-p 後半～8.08Vp-p
第二次高調波	-40dB 以下
振動子	水晶
出力インピーダンス	50Ω

3. 測定の内容と条件

測定は実験の再現性を考慮し、電氣的ノイズが限りなくゼロに近いシールドルーム内にて実施した。このシールドルームの大きさは、10m×5.5m×3mである。

また、ボディエリアネットワークのチャネルの振る舞いを検討するための動作の一例として、NHKから放送されているラジオ体操第一を選んだ。この理由は、ラジオ体操第一は姿勢変化のバリエーションがある程度豊富であり、なおかつ、姿勢を変化させる動作を誰でも再現できると考えられるためである。より詳細な検討を行うためには、より多くの姿勢のバリエーションが必要と考えられるが、本研究では測定結果の統計解析手法の確立に重きを置いており、詳細な検討は今後の課題とする。

ラジオ体操第一はおよそ180秒あり、動作ルーチンは、前奏で待機するときの直立静止状態を除けば13個の動作ルーチンが存在する（表3.1参照）。

表 3.1 ラジオ体操第一の動作

動作ルーチンの順番	動作ルーチンの内容
前奏	直立状態
①	背伸びの運動
②	腕と足の曲げ伸ばし運動
③	肩の運動
④	胸の運動
⑤	横曲げの運動
⑥	前後の運動
⑦	ねじりの運動
⑧	腕と足の運動
⑨	斜め下への運動
⑩	大きく回す運動
⑪	両足跳びの運動
⑫	②と同じ
⑬	深呼吸（①と同じ）

このとき、作製した測定機器を装着し、このラジオ体操第一を行ったのは筆者である（身長185cm、体重75kg）。なお、ラジオ体操第一を行う際は、厚さ10cmの発泡スチロールの上に乗って実施しており、電氣的に浮いている。また、送信器を腹部に、受信器を左腕に装着して行っている。

さらに、送受信器の接地状態によってボディエリアネットワークの振る舞いに変化することが知られている[1],[2]。そのため、送受信器がそれぞれ接地された状態の測定も行うのだが、この状態を実現するために、およそ2.5mの被覆銅線を用いた。送信器や受信器を接地状態にする際は、これらの回路グラウンド（図2.1.2における「Top electrode」）と、シー

ルドルームの床を繋ぎ、同電位としている。被覆銅線を基板と繋ぐ際は SMA コネクタによって堅固な状態を実現している。また、シールドルームの床と被覆銅線は、圧着端子をネジ止めすることによって堅固な状態を実現している。

この送受信器の接地状態に関する条件は 4 つ存在し、本研究においては簡単に次のように表記する。

No GND : 送受信器共に接地していない状態。共に非接地状態。

GND Rx : 受信器のみを接地した状態。

GND Tx : 送信器のみを接地した状態。

GND Tx & Rx : 送受信器を共に接地した状態。

測定風景の一例を図 3.1 に示す。

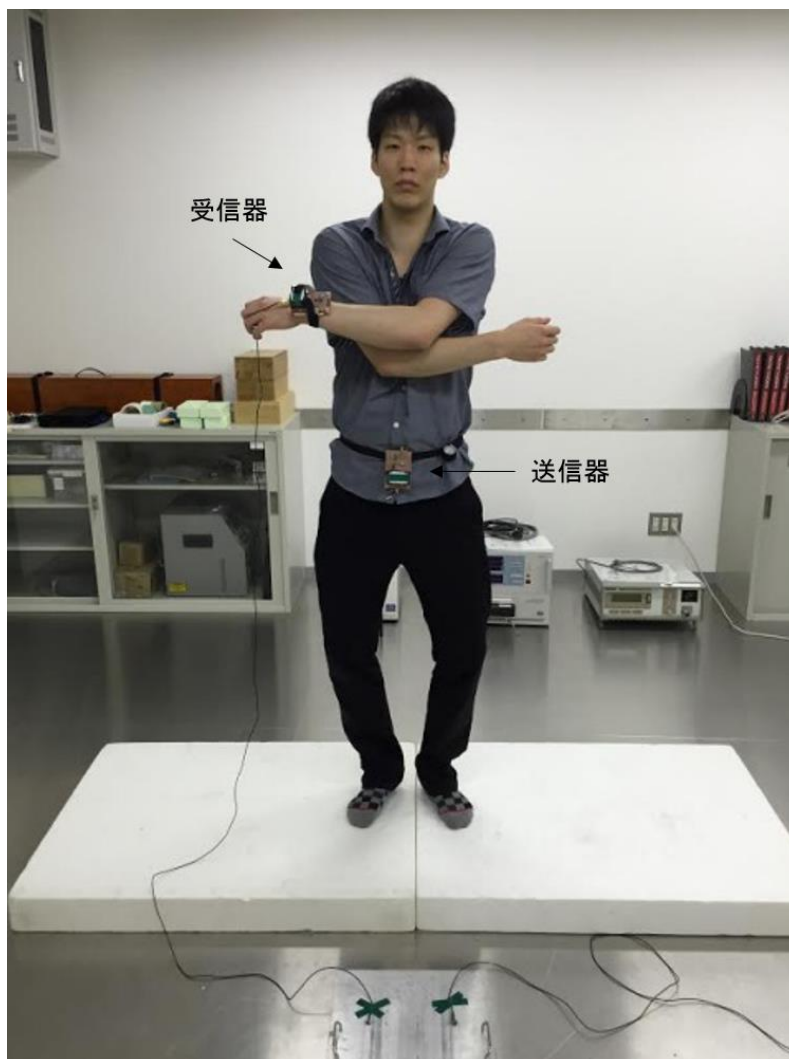


図 3.1 測定風景

4. 測定結果

4.1. 受信電圧強度の時間変化

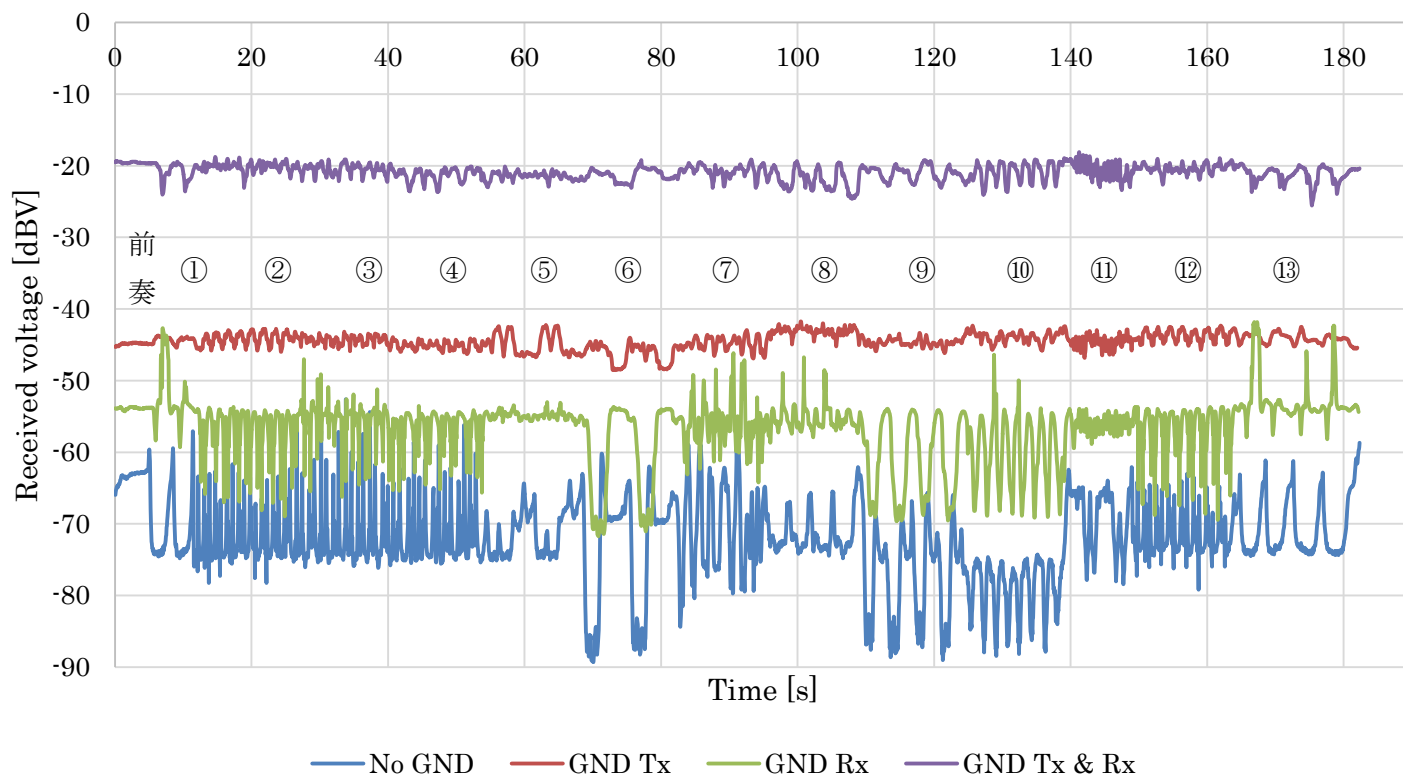


図 4.1.1 ラジオ体操第一における BAN の測定結果（時間領域）

図 4.1.1 は、3 章において述べた測定条件によって得られた受信電圧強度の時間変化である。図中に示されている数字は、ラジオ体操第一において動作ルーチンが行われているおおよその区間に割り振られており、数字は 3 章の表 3.1 と対応している。

図 4.1.1 を見ると分かるように、送受信器の接地状態によって測定結果の傾向が異なることが分かる。「GND Tx & Rx」や「GND Tx」においてはあまり動作に依存せず、他の 2 つのパターンと比べて受信電圧レベルが高く、また安定している。一方、「GND Rx」や「No GND」においては、受信電圧にばらつきが多い傾向がある。特に、「No GND」の動作⑥や⑨では、受信電圧レベルがほとんどノイズフロアレベルまで低下する瞬間もある。

この動作⑥や⑨は、前屈をするような運動であり、腹部にある送信器と左腕にある受信器とを繋ぐチャンネルが、前屈をした際に上体によって遮られるため、ほぼ無信号状態になると考えられる（図 4.1.2 参照）。

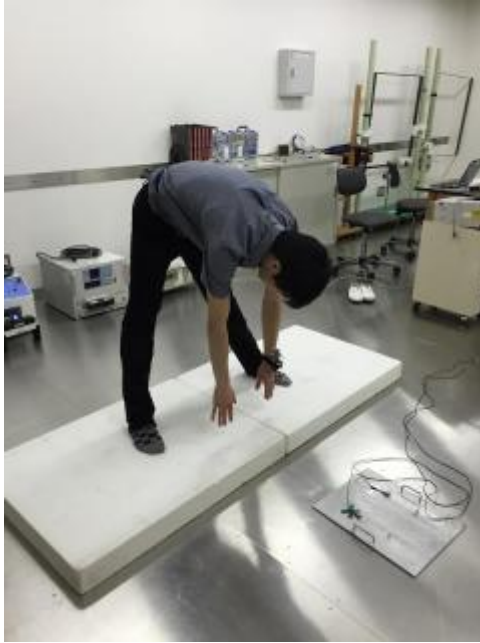


図 4.1.2 前屈運動

4.2. 受信電圧強度の確率密度関数

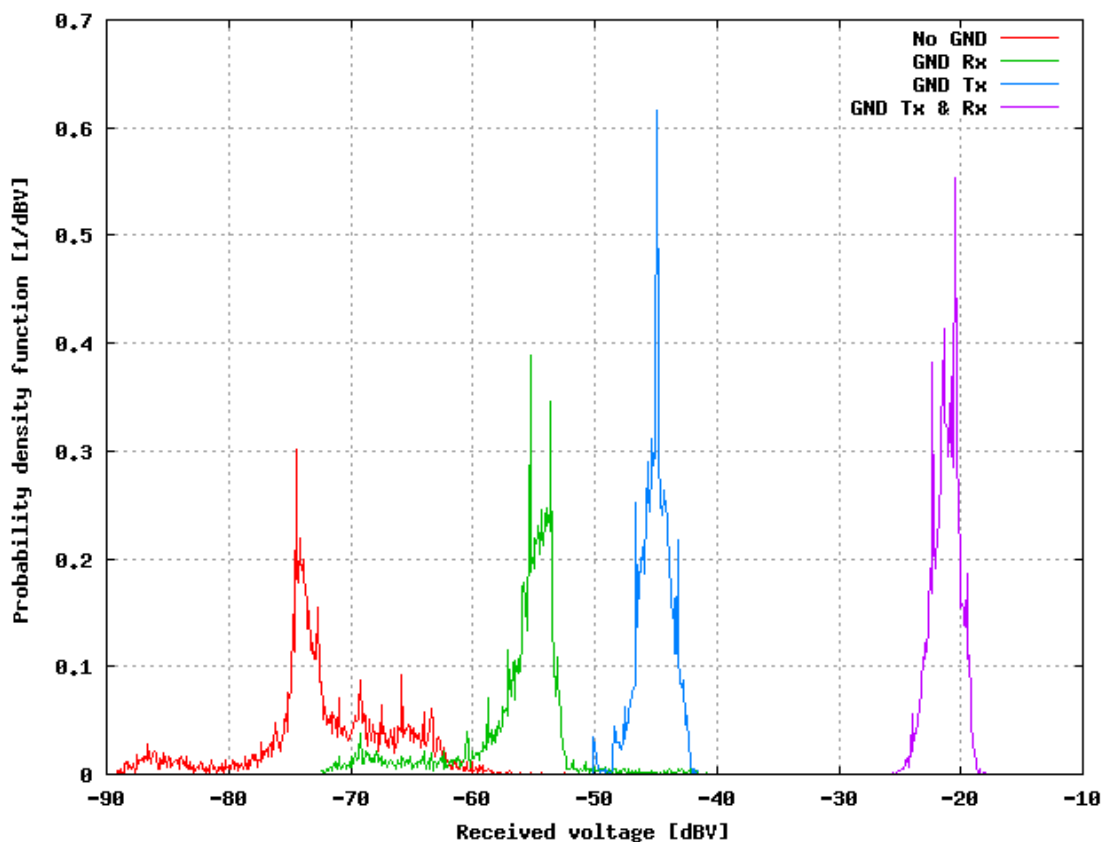


図 4.2.1 確率密度関数による評価

表 4.2.1 各接地状態における受信電圧の平均値と標準偏差

接地状態	平均値 [dBV]	標準偏差 [dBV]
No GND	-72.19	5.651
GND Rx	-56.52	4.428
GND Tx	-45.08	1.342
GND Tx & Rx	-21.27	1.125

次に、受信電圧強度の確率密度関数を評価する。前項の図 4.1.1 では、各接地状態における測定が 1 回分しか示されていないが、実際は各接地状態において 2 回ずつ測定を行っている。この各接地状態の 2 回分の測定結果をひとまとまりとし、確率密度関数として評価した。評価結果は図 4.2.1 の通りであり、このときの各接地状態における平均値と標準偏差は表 4.2.1 の通りである。なお、横軸の刻み幅は 0.1dBV としている。

受信器は 1 秒間に 20 個の測定値を記録するため、およそ 180 秒あるラジオ体操第一の 1 回あたりのデータ数は約 3600 個である。これを 2 回分まとめて解析するため、一つの接地状態におけるデータ数は約 7200 個である。

本研究では、これらの各接地状態における確率密度関数の近似曲線を求め、数式化することを目指している。次章より、この手法を考える。

5. 解析手法

5.1. 複数の正規分布を用いた近似

4 章図 4.2.1 の確率密度関数を、近似曲線によって表す手法について検討する。筆者が検討した近似手法は、各接地状態の確率密度関数の分布を、複数の正規分布を足し合わせることによって近似する、という手法である。これは、送受信器を装着した人体が静止状態であるときは、受信電圧の確率密度が正規分布に従うという検証結果に基づいた手法である。

今、静止した 3 つの姿勢 pose1~3 を以下のように定義する。

pose1 : 直立状態。両腕を下ろし、足を閉じて立っている。

pose2 : 十字状態。両腕を伸ばしたまま左右に肩の高さまで上げ、足を閉じて立っている。

pose3 : 前屈状態。足を肩幅強に広げ、両足を伸ばしたまま両腕を地面に届かせるようにかがむ。

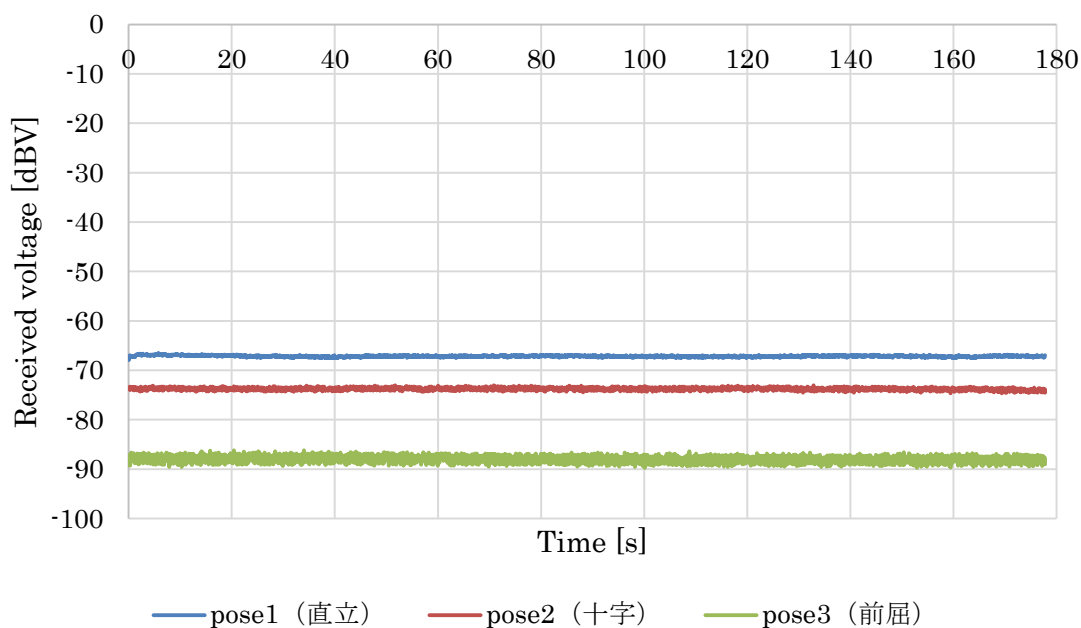


図 5.1.1 3 つの姿勢の静止状態における受信電圧 (No GND)

これらの各姿勢で 180 秒静止した際の測定結果は図 5.1.1 の通りである。送受信器の接地状態は「No GND」である。この測定結果からも明らかであるように、受信電圧はほとんど一定である。そして、これらの 3 つの姿勢は、ラジオ体操第一の中でも何度か登場する姿勢であり、このことは図 4.1.1 の測定結果の受信電圧と見比べてもおおよそ一致する。つまり、ラジオ体操第一に登場する全ての姿勢における受信電圧を調べ、それらを適当な比率

で組み合わせれば、その接地状態における確率密度関数は再現できるはずである。もちろん、ラジオ体操第一の全ての姿勢に対する受信電圧を調べることは困難であるため、本研究では1~4つの正規分布を用いて近似することを目指した。

$$\text{Norm}(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x-\mu)^2}{2\pi\sigma^2}\right) \quad (5.1.1)$$

正規分布 $\text{Norm}(x)$ は式(5.1.1)のように表すことができ、この正規分布の形は平均値 μ と標準偏差 σ によってのみ決まる。本研究で用いる近似手法では、複数の正規分布の各平均値 μ と標準偏差 σ を適当な値に変化させ、最適な近似曲線を得る。また、複数の正規分布に重み付け係数 w を掛け、全ての正規分布の積分値が1となるようにする。つまり、本研究における近似曲線 $\text{PDF}_{\text{optim}}(x)$ は、以下のように定義できる。

$$\text{PDF}_{\text{optim}}(x) = \sum_{i=1}^n \frac{w_i}{\sqrt{2\pi}\sigma_i} \exp\left(-\frac{(x-\mu_i)^2}{2\pi\sigma_i^2}\right) \quad (5.1.2)$$

ただし、 σ_i は標準偏差、 w_i は重み付け係数、 μ_i は平均値、 n は用いる正規分布の数である。また、 x は受信電圧に対応している。そして、 $\text{PDF}_{\text{optim}}(x)$ の積分値が1となるためには、次の式(5.1.3)が成り立つ必要がある。

$$\sum_{i=1}^n w_i = 1 \quad (5.1.3)$$

この式(5.1.3)を達成するために、実際には

$$w_n = 1 - \sum_{i=1}^{n-1} w_i \quad (5.1.4)$$

というようにする。例えば $n=3$ のとき、 $w_3 = 1 - (w_1 + w_2)$ となる。また、 w_i は必ず正となることも追加条件として考慮しなければならない。つまり

$$0 \leq w_i \leq 1 \quad (5.1.5)$$

となる。本研究における最適化においては、この条件を w_i に対して絶対値を取り、 $|w_i|$ として考えることによって対処した。

5.2. 最適化と評価関数

近似をするためのアプローチは前節にて説明したが、この手法を実行するために、本研究では最適化を用いる。最適化は、実測値と近似曲線との誤差の二乗和（詳細は後述）が最小となるように行い、式(5.1.2)における標準偏差 σ_i 、重み付け係数 w_i 、平均値 μ_i （以降、パラメータと称する）を次々と変化させ、誤差の二乗和が最小となる時のパラメータの組み合わせを求める。つまり、これらのパラメータを引数として受け取り、誤差の二乗和を返す関数を最小化する。そのため、そのような関数を用意する必要があり、この関数が最適化の評価関数となる。

今、実測値から得られた確率密度関数を $\text{PDF}_{\text{measured}}(x)$ 、その確率密度関数の近似曲線を $\text{PDF}_{\text{optim}}(x)$ とする。前述した式(5.1.2)におけるパラメータを 1 つのベクトル $\vec{x} = (\sigma_1, \dots, \sigma_n, w_1, \dots, w_{n-1}, \mu_1, \dots, \mu_n)$ とすれば、最適化の評価関数 $f(\vec{x})$ は以下のように表せる。

$$f(\vec{x}) = \sum \{ \text{PDF}_{\text{measured}}(x) - \text{PDF}_{\text{optim}}(x) \}^2 \quad (5.2.1)$$

よって式(5.1.2)より、

$$f(\vec{x}) = \sum \left\{ \text{PDF}_{\text{measured}}(x) - \sum_{i=1}^n \frac{w_i}{\sqrt{2\pi}\sigma_i} \exp\left(-\frac{(x-\mu_i)^2}{2\pi\sigma_i^2}\right) \right\}^2 \quad (5.2.1)$$

と定義することができる。 x は受信電圧対応しており、 $\text{PDF}_{\text{measured}}(x)$ の刻み幅が 0.1dBV であるため、この式(5.2.1)も同じ刻み幅で計算している。なお、本研究においては、最適化に使用する正規分布の数 n は、固定値（プログラマ側で決める）としており、評価関数 $f(\vec{x})$ の引数には含まれない。

5.3. 最適化手法

本研究に使用した最適化アルゴリズムは、遺伝的アルゴリズム (GA: Genetic Algorithm) と滑降シンプレックス法 (Nelder-Mead 法) である[13],[14]。これら 2 つのアルゴリズムを組み合わせ、前節で説明した誤差の二乗和を最小化する。

これら 2 つのアルゴリズムは、無料統計解析ソフト R を介して使用した[15]。遺伝的アルゴリズムは R の別途パッケージ「genalg」をミラーサイト「Japan」からインストールすると、「rbga」という関数によって使用することができる。また、滑降シンプレックス法は、R の標準最適化関数である「optim」という関数によって使用することができる。

最適化の大まかな手順は、まずグローバルな最適化を遺伝的アルゴリズムによって行い、その後、ローカルな最適化を滑降シンプレックス法によって行う、という流れである。遺伝的アルゴリズムはローカルミニマムに陥る危険度が低い値の収束が遅い。一方、滑降シンプレックス法は値の収束が速く精度も高い。これらの長所と短所を上手く組み合わせることを考える。

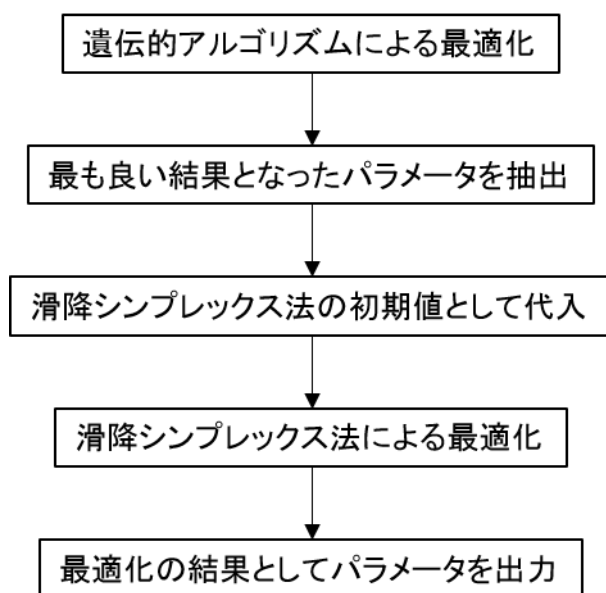


図 5.3.1 最適化手順

本研究における最適化の手法は、図 5.3.1 のような流れで行う。この手順に沿って R でプログラムを作成した。R における遺伝的アルゴリズムを行う関数「rbga」では、設定項目が 11 個ある[15]。本研究においては、主要な 5 項目をプログラマが設定し、その他の項目は未設定および標準の仕様に任せている。設定した項目は、次の 5 つである。

- ①各パラメータの取り得る最小値 (stringMin)
- ②各パラメータの取り得る最大値 (stringMax)
- ③最適化の試行回数 (popSize)

④評価する関数 (evalFunc)

⑤突然変異確率 (mutationChance)

まず、①と②に関して、標準偏差 σ_i が取り得る値は 0~100dBV, 重み付け係数 w_i が取り得る値は 0~1, 平均値 μ_i が取り得る値は-100~0dBV とした。③に関しては、近似に用いる標準偏差の数 n によって変えており、 $n = 1$ または 2 ならば popSize=200 回、 $n = 3$ ならば popSize=500 回、 $n = 4$ ならば popSize=1000 回としている ($n = 5$ 以上は解析時間が膨大になるため未実施)。④に関しては、式(5.2.1)をプログラム上で表現した関数を用いる。そして⑤は、[15]のサンプルコードで 0.01%がよく用いられていたため、本研究においても 0.01%と設定した。

遺伝的アルゴリズムにおいては、ローカルミニマムに陥る危険を回避するために突然変異が存在する。しかしそのため解の収束を判断する条件がやや曖昧となる。そこで、「ある一定回数の試行結果の中から最も良いものを抽出」というプログラムを組んでいる。このとき、本研究における 1 回の試行結果というのは、誤差の二乗和が小さくなるようなパラメータの組み合わせを 1 組選出することを指している。

滑降シンプレックス法は突然変異が存在しないため、ローカルミニマムの回避は難しいが、本手法ではすでに遺伝的アルゴリズムによってローカルミニマムは回避していることを前提としている。そのため、滑降シンプレックス法を用いる際は最適化を終了させるための収束条件のみを考える。R において滑降シンプレックス法を行う「optim」という関数は、プログラマが設定できる項目が十数個存在するが、本研究においては、初期値と評価関数を指定する箇所以外は全てデフォルトの設定としている。

本研究においては、収束条件もデフォルトのままとして解析している。デフォルトでは誤差の二乗和が 1×10^{-8} 以上変化しない場合を収束としており、本研究ではそのときのパラメータの組み合わせを最適解としている。なお、optim 関数のその他のデフォルトは[16]を参照すると良い。

なお、optim 関数は一度の命令で 500 回の最適化を実行するのがデフォルトとなっている。しかし、評価関数の出力（本研究における誤差の二乗和）が 1×10^{-8} 以上変化しない場合は、500 回を待たずして最適化が終了する。つまり、試行回数が 500 回で終わった場合の最適化はまだ続きがあり、試行回数が 500 回未満で終えた場合の最適化は、それ以上の最適化が行えず限界である、と言える。プログラムはこの考えによって作成している。

5.4. 最適化プログラム

```

library(genalg)           #ライブラリ「genalg」の呼び出し
data1<-scan("NoGND_x-smooth.txt") #data1にx軸のデータをベクトル形式で読み込み
data2<-scan("NoGND_y-smooth.txt") #data2にy軸のデータをベクトル形式で読み込み
dBV<-matrix(data1, ,1)   #data1をn行1列の行列dBVとして変換
PDF<-matrix(data2, ,1)  #data2をn行1列の行列PDFとして変換
n<-nrow(dBV)            #データ数をnに記憶
norm1 <- 0              #近似に用いる正規分布1
norm2 <- 0              #近似に用いる正規分布2
norm3 <- 0              #近似に用いる正規分布3
diff <- 0               #誤差
sqsum <- 0              #誤差の二乗和

#評価関数の定義
f <- function(x){       #ベクトルxを引数に持つ関数f(x)の作成
  for(i in 1:n){
    sd1 <- x[1]; sd2 <- x[2]; sd3 <- x[3];      #ベクトルxの要素の定義
    w1 <- x[4]; w2 <- x[5];                    #x=(sd1,sd2,sd3,w1,w2,w3,u1,u2,u3)
    u1 <- x[6]; u2 <- x[7]; u3 <- x[8];        #と設定している

    #正規分布の式を定義。wが負にならないように絶対値absとする。
    norm1 <- abs(w1)/(sqrt(2*pi*(sd1^2)))*exp(((dBV[i]-u1)^2)/(-2*(sd1^2)))
    norm2 <- abs(w2)/(sqrt(2*pi*(sd2^2)))*exp(((dBV[i]-u2)^2)/(-2*(sd2^2)))
    norm3 <- abs(1-abs(w1)-abs(w2))/(sqrt(2*pi*(sd3^2)))*exp(((dBV[i]-u3)^2)/(-2*(sd3^2)))

    diff <- PDF[i]-(norm1+norm2+norm3)          #誤差の計算
    sqsum <- sqsum+diff^2                       #誤差の二乗和の計算
  }
  return(sqsum)                                #出力として誤差の二乗和を返す
}

```

図 5.4.1 最適化プログラム（初期設定、データ読み込み、評価関数の定義）

ファイル(I)	ファイル(F)
-89.20	0.001377
-89.10	0.002753
-89.00	0.001377
-88.90	0.004130
-88.80	0.001377
⋮	⋮
-52.90	0.000000
-52.80	0.000000
-52.70	0.000000
-52.60	0.000000
-52.50	0.001377

NoGND_x-smooth.txt NoGND_y-smooth.txt

図 5.4.2 測定結果ファイル

```

#遺伝的アルゴリズムによって滑降シンプレックス法の初期値パラメータを決める
popu <- 500 #最適化の試行回数の設定
gaout <- rbga(c(0,0,0,0,0,-100,-100,-100), #各パラメータの最小値の決定
             c(100,100,100,1,1,0,0,0), #各パラメータの最大値の決定
             popSize=popu, #最適化の試行回数の設定
             evalFunc=f, #評価関数の設定
             mutationChance=0.01) #突然変異確率の設定

gamin <- 10 #最適化を行った中で、一番sqsumの値が小さかったものを記憶する変数
gacount <- 0 #そのときの配列番号を記憶する変数

for(i in 1:popu){
  if(gaout$evaluations[i] < gamin){ #全試行結果から最小の出力を探る
    gamin <- gaout$evaluations[i]
    gacount <- i #そのときの配列番号を記憶する
  }
}

gaout$population[gacount, ] #最もsqsumの値が小さかったときのパラメータを画面に出力
par <- gaout$population[gacount, ] #それらを滑降シンプレックス法の初期値としてparに代入
f(par) #そのときの評価関数の値を出力

#滑降シンプレックス法を限界まで行う(sqsumが1e-8変わらなくなるまで)
print <- 0 #最適化の試行回数履歴
sum <- 0 #滑降シンプレックス法を用いた最適化の全試行回数
i <- 0 #カウント
j <- 0 #whileループを抜けるフラグ
while(j < 1){
  i = i+1
  sim <- optim(par, f) #1回の最適化結果を変数simに代入(あえて出力はしない)
  par <- sim$par #上の最適化結果の中からpar部分のみを抽出し、次の最適化の初期値に用いる(パラメータの上書き)
  print[i] <- sim$counts[1] #配列printに最適化の試行回数を格納(配列を用いないと数字のみを抽出できない)
  if(print[i] < 501){ #最適化の試行回数が500以下であるなら、jを+1する(するとwhileループを抜ける)
    j = j+1
  }
  sum = sum + print[i] #全部を通して行われた最適化の試行回数を計算する
}
print #最適化の試行回数履歴を表示
sum #滑降シンプレックス法を用いた最適化の全試行回数を表示
sim #sim(最適化関数の出力)を表示
out <- c("parameter", sim$par, " ", "value", sim$value, " ", "counts", sum) #解析結果からparとvalueとsumを抽出
write(out, "C:/R_data/NoGND-3Norm_parameter.txt", append=F, ncolumns=1) #解析結果をテキストファイルに出力

```

図 5.4.3 最適化プログラム (最適化、ファイル出力)

図 5.4.1 および図 5.4.3 は本研究で用いた R における最適化プログラムである。解析する対象は、「No GND」における測定結果で、正規分布を 3 つ用いた最適化である。なお、読み込む測定結果ファイルの体裁は、図 5.4.2 のようになっている。このプログラムによって得られたパラメータを基に、グラフを描画する。

6. 解析結果

6.1. 解析対象データの平滑化

まず、5章で説明した解析を行う前に、解析対象のデータの平滑化を行う。この平滑化は、解析する際、対象のデータがノイズ状であると適切な解析が行えなくなってしまう、という現象を取り除くために行う。5章における図 5.4.2 はすでに平滑化後の測定結果である。

具体的には、4章における図 4.2.1 の各接地状態における確率密度関数に対し、移動平均を施す。本研究における移動平均は、あるデータの値を、その前後 5 つのデータ（計 11 データ）の平均値とするように施す。その結果を以下に示す。

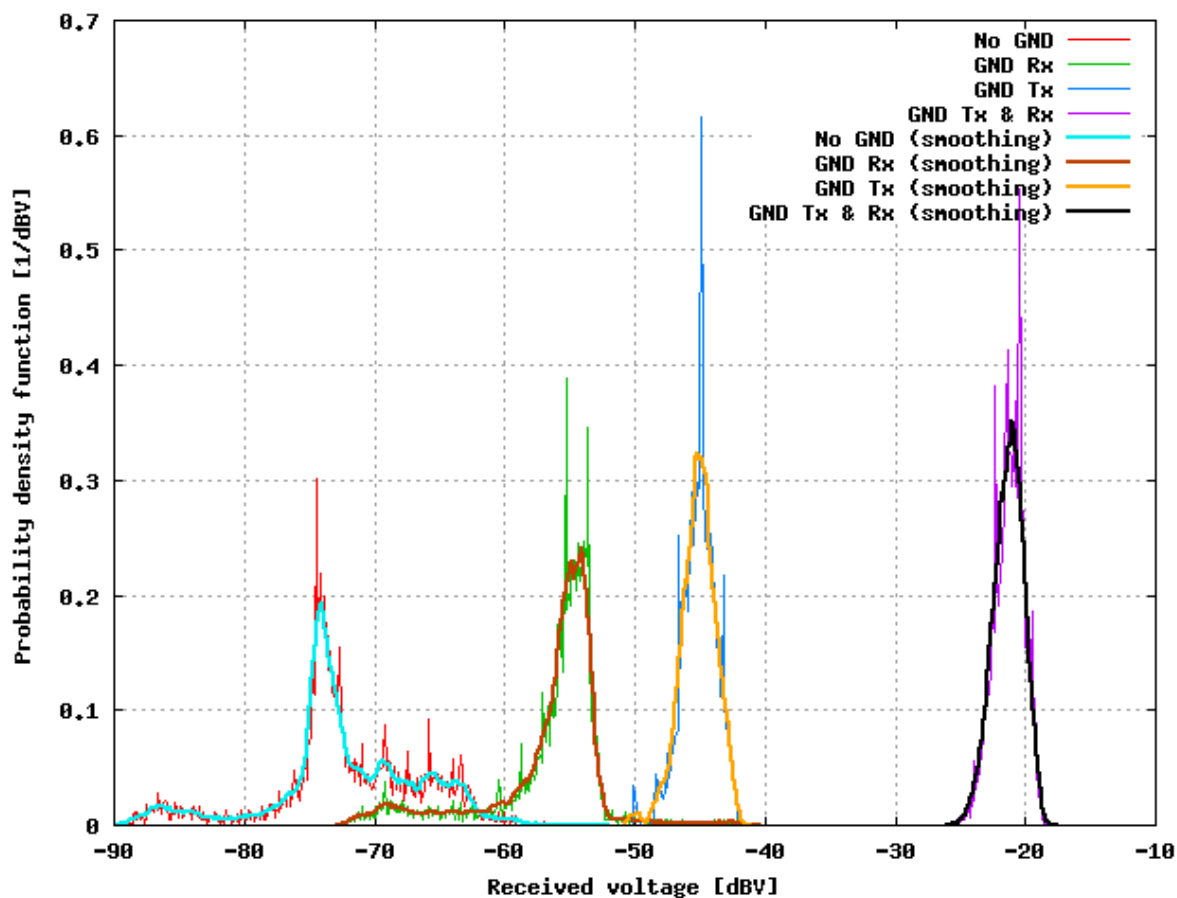


図 6.1.1 解析データの平滑化

次節以降の解析結果は、この平滑化したデータに対して行う。

6.2. 最適化結果

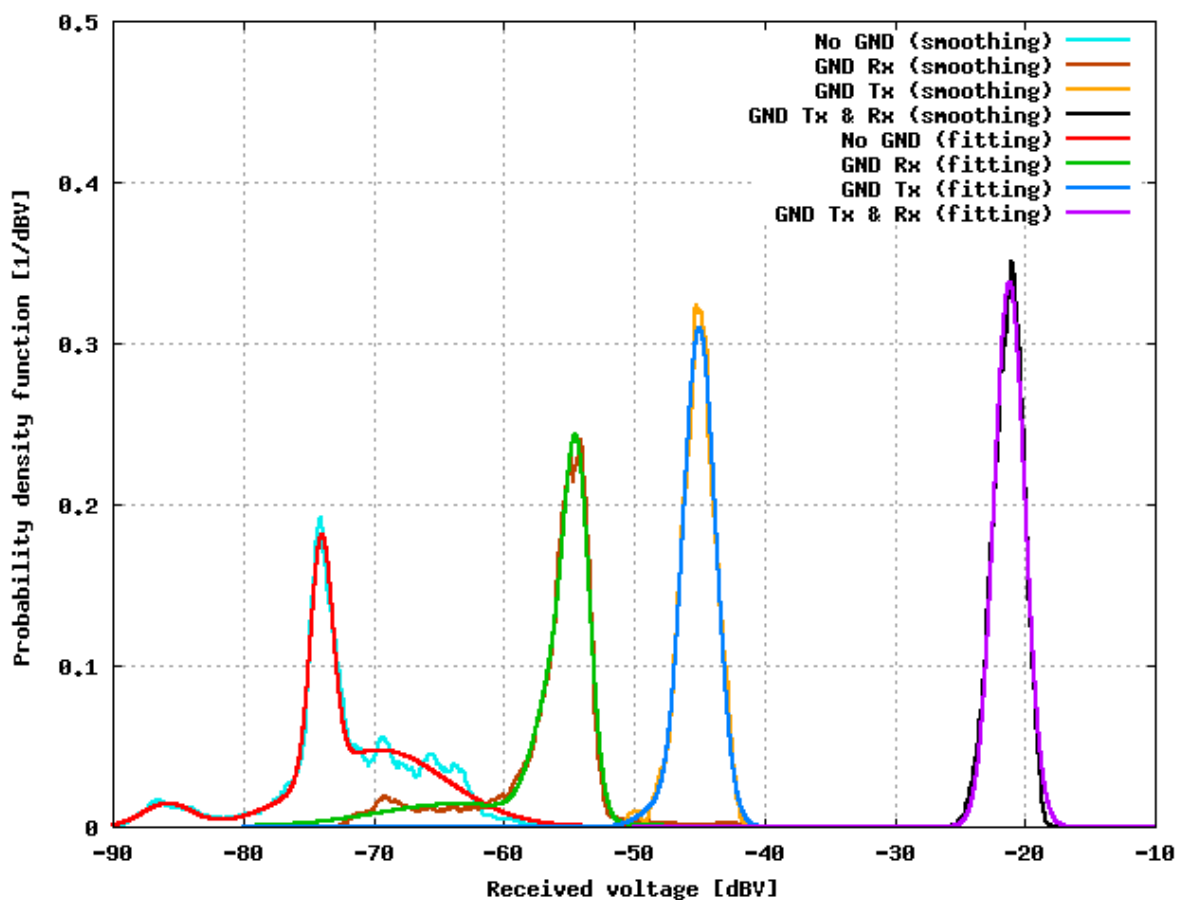


図 6.2.1 最適化結果

最適化結果は図 6.2.1 のようである。このとき各接地状態の近似曲線は、式(5.2.1)における各パラメータを次のようにすると描画できる。

表 6.2.1 近似曲線のパラメータまとめ

接地状態	n	パラメータ	
No GND	3	σ_1 [dBV]	5.206
		σ_2 [dBV]	0.8495
		σ_3 [dBV]	1.854
		w_1 [-]	0.6191
		w_2 [-]	0.3157
		w_3 [-]	0.06519
		μ_1 [dBV]	-69.55
		μ_2 [dBV]	-73.95
		μ_3 [dBV]	-85.80

GND Rx	3	σ_1 [dBV]	1.531
		σ_2 [dBV]	0.9748
		σ_3 [dBV]	6.188
		w_1 [-]	0.3164
		w_2 [-]	0.4662
		w_3 [-]	0.2174
		μ_1 [dBV]	-56.02
		μ_2 [dBV]	-54.33
		μ_3 [dBV]	-63.42
GND Tx	2	σ_1 [dBV]	1.251
		σ_2 [dBV]	1.297
		w_1 [-]	0.9687
		w_2 [-]	0.03129
		μ_1 [dBV]	-44.96
		μ_2 [dBV]	-48.46
GND Tx & Rx	1	σ_1 [dBV]	1.182
		w_1 [-]	1.000
		μ_1 [dBV]	-21.16

図6.2.1を見ると、おおよその近似曲線を求めることができていることがわかる。つまり、本研究においては、このような最適化手順によって確率密度関数モデルを推定することができると言え、パラメータも解析的に求めることができる。

6.3. 正規分布の数 n に関する依存性

前節において、近似曲線を求めることができた。そしてこの近似曲線の精度をより高めることや、もっと少ない数の正規分布によって簡単に近似する場合を考える。つまり、各接地状態の測定結果に対して、何個の正規分布を用いて近似すれば妥当な結果になるのか、ということを考える。

今、各接地状態の測定結果に対して、用いる正規分布の数 n を 1~4 に変化させ、そのときの近似精度を考える。近似の精度は、誤差の二乗和 (sqsum とする) によって相対的に評価する。なお、誤差の二乗和は、確率密度関数[1/dBV]の差を二乗し足し合わせているため、単位は[1/dBV²]である。

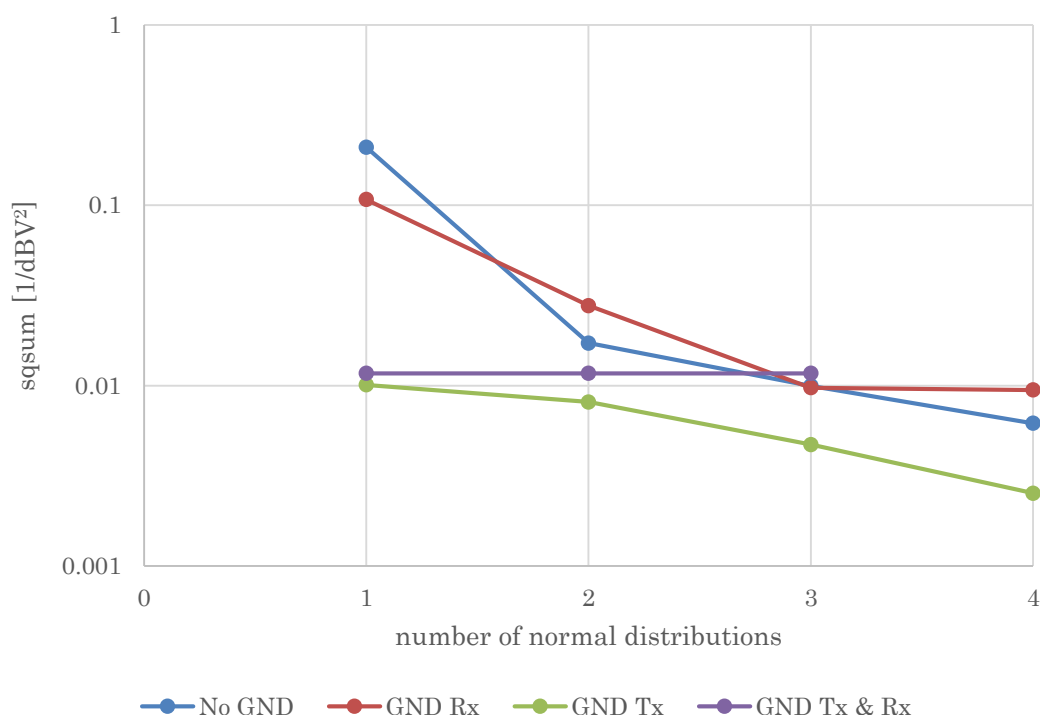


図 6.3.1 正規分布の数と近似精度との関係

図 6.3.1 は、近似に用いる正規分布の数 n と二乗誤差 sqsum との関係を表している。縦軸は対数表示となっている。この結果から、「No GND」や「GND Rx」のように、受信電圧の姿勢変化にたいする依存性の大きい条件では、精度の良い近似に必要な正規分布の数が多くなることが読み取れる。また、この結果からだけでは一概に言うことは難しいが、sqsum がおよそ 0.01 付近からは、あまり精度は向上しない。つまり sqsum \approx 0.01 となるときの n を設定すれば、十分な近似が行えていると言える。

また、「GND Tx & Rx」のような、ほぼ単一の正規分布のみで近似できる測定結果に対して、あえて n を増やしたりすると、値が収束しなかったり、数学的にはあり得ても物理的にあり得ないパラメータとなる。そのため、「GND Tx & Rx」の $n = 4$ においては、正当な結

果を取得できていない。なお、 $n = 5$ 以上に関しては、計算時間が膨大に掛かる傾向にあるため、本研究においては言及していない。

6.4. 最適化結果からラジオ体操第一の動作を考える

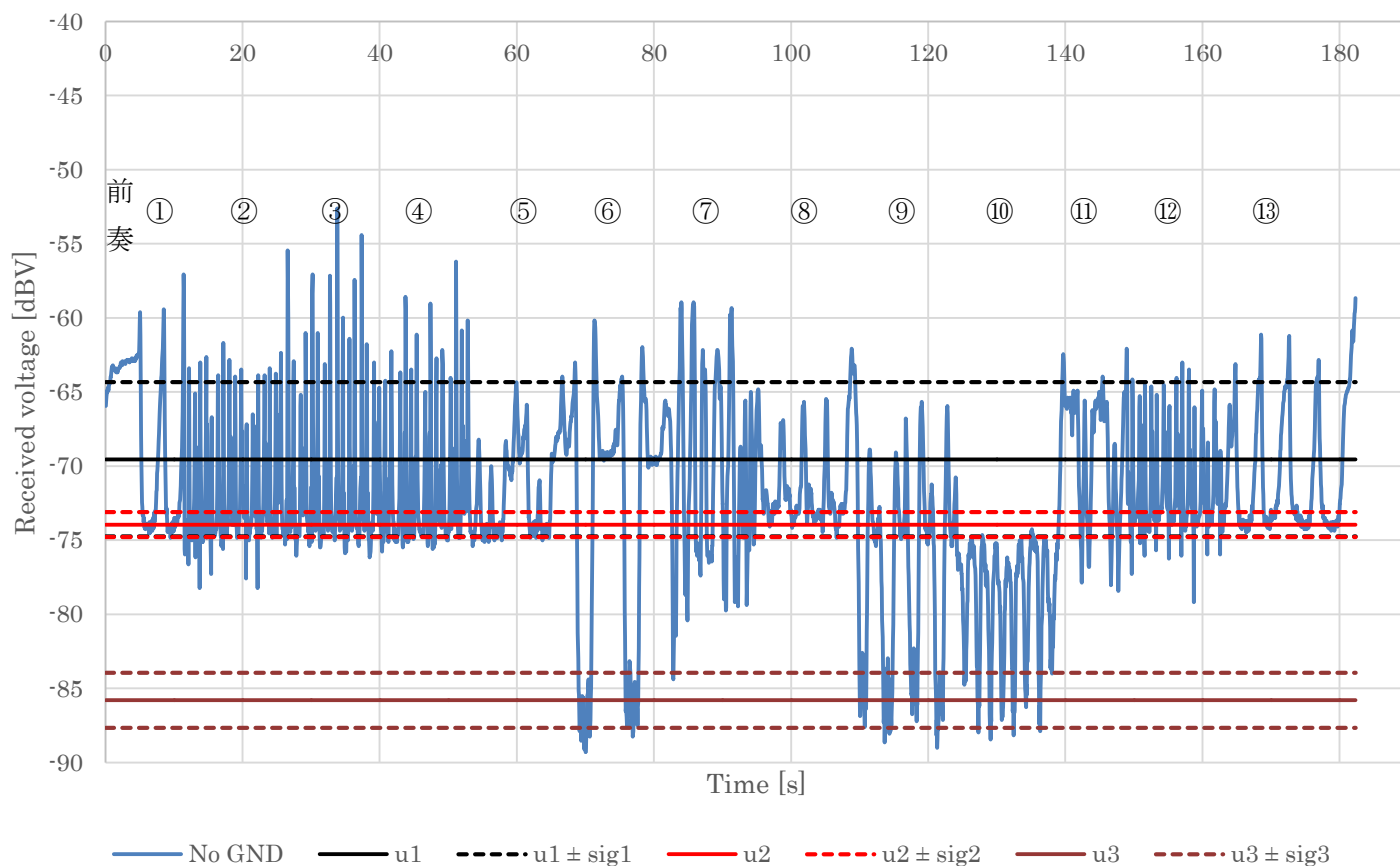


図 6.4.1 No GND における測定結果と $\mu_1 \sim \mu_3$, $\pm\sigma_1 \sim \sigma_3$

図 6.4.1 は「No GND」におけるラジオ体操第一の測定結果に、表 6.2.1 の最適化結果から平均値 $\mu_1 \sim \mu_3$ (実線) およびその平均値 $\pm\sigma_1 \sim \sigma_3$ (破線) を描画したものである。なお、 μ_2 と $\mu_1 - \sigma_1$ はほとんど値が同じであり、図中では線が被り μ_2 しか表示されていない。動作番号は 3 章の表 3.1 に対応している。

この $\mu_1 \sim \mu_3$ を持つ正規分布 $\text{Norm}_1 \sim \text{Norm}_3$ は、表 6.2.1 の最適化結果から求めた重み付け係数 $w_1 \sim w_3$ より、

$$\text{Norm}_1 : \text{Norm}_2 : \text{Norm}_3 \cong 6 : 3 : 1$$

という比率である。このことを念頭に置き図 6.4.1 を見てみると、ラジオ体操第一における動作を関連付けて考えることができる。

このような観点で最適化結果と測定結果を見比べると、他の接地状態におけるラジオ体操第一における支配的な動作が分かる。これは、「No GND」や「GND Rx」のように、受信電圧のばらつきが多い接地状態において、より有効な考え方と言える。また、本研究における近似手法によって得られた近似曲線式が、おおよそ妥当な結果であることの裏づけにもなる。

7. 結論

ボディアエリアネットワークの振る舞いを測定することに即した測定機器を開発し、それを用いた測定をすることで、測定機器の接地状態や人体の動作によってボディアエリアネットワークの伝送特性が様々に変化することを実験によって確かめることができた。これは、シミュレーションによる解析だけでは得がたい最大の成果であると考えられる。

また、最適化手法によって、複数の正規分布を用いた近似を行うことにより、先行研究よりも更に厳密な統計的チャネルモデルを定義することができたと考えられる。そのため、数十メガヘルツ帯における人体通信チャネルを議論する際は、本研究で提案する手法によって、数式モデルを定義することが有効であると考えられる。

8. 今後の課題および方針

本研究では、ボディエリアネットワークの振る舞いを測定するための動作として、ラジオ体操第一を採用した。ラジオ体操第一は、様々な動作が入り乱れ、誰にでも再現可能な動作であるため、測定のための動作としては大いに適しているが、おおよそ人間が行う日常の動作とは一致しない。そのため、本当の意味で様々な動作が入り乱れている際のボディエリアネットワークの振る舞いを測定しモデル化することが必要である。これには、日常の様々な動作における測定および解析と、また膨大な数の測定結果が必要である。そして、それらの測定結果を踏まえ、より一般的なチャネルモデルを検討、定義することが、今後の課題として挙げられる。

9. 謝辞

本研究を進めるにあたり本島邦行教授、羽賀望助教には大変お世話になりました。厚くお礼申し上げます。また、主査、副査を引き受けてくださった山越芳樹教授並びに三輪空司准教授に深くお礼申し上げます。

そして最後に、研究室の皆様におかれましても、日頃からの支援に感謝申し上げます。

10. 参考文献

- [1] 羽賀望, 齊藤一幸, 高橋応明, 伊藤公一, “準静電界を用いた人体通信チャネルにおける姿勢及び大地の影響”, 電子情報通信学会論文誌(B), vol. J95-B, no. 2, pp. 257–264, Feb. 2012.
- [2] N. Haga, K. Saito, M. Takahashi, and K. Ito, “Proper derivation of equivalent-circuit expressions of intra-body communication channels using quasi-static field,” IEICE Transactions on Communications, vol. E95-B, no. 1, pp. 51–59, Jan. 2012.
- [3] T. G. Zimmerman, Personal Area Networks (PAN): Near-Field Intra-Body Communication, M.S. thesis, MIT Media Laboratory, Cambridge, MA, 1995.
- [4] T. G. Zimmerman, “Personal area networks: Near-field intra-body communication,” IBM Syst. J., vol. 35, no. 3/4, pp. 609-617, 1996
- [5] N. Zedong, M. Jingjing, K. Ivanov, W. Lei, “An investigation on Dynamic Human Body Communication Channel Characteristics at 45 MHz in Different Surrounding Environments,” IEEE ANTENNAS AND WIRELESS PROPAGATION LETTERS, VOL. 13, 2014, pp. 309–312
- [6] 笠原佑作, 卒業論文 “ボディアエリアネットワークにおける動的通信チャネルの測定機器開発”
- [7] EAGLE によるプリントパターン自動作成
http://www.piclist.com/images/www/hobby_elec/eagle.htm
- [8] 後閑哲也, “C 言語ではじめる PIC24F 活用ガイドブック”, 技術評論社
- [9] Microchip 社ホームページ, <http://www.microchip.com/>
- [10] AD8307 データシート,
http://www.analog.com/static/imported-files/jp/data_sheets/AD8307_JP.pdf
- [11] LT1490 データシート,
http://cds.linear.com/docs/jp/datasheet/j14901_6.pdf
- [12] SD カードスロット DM3AT-SF-PELM5 データシート
<http://akizukidenshi.com/download/ds/hirose/DM3AT-SF-PEJM5.pdf>
- [13] 伊庭斉志, 遺伝的アルゴリズムの基礎—GA の謎を解く—, オーム社
- [14] 丹慶勝市他訳, ニューメリカルレシピ・イン・シー 日本語版—C 言語による数値計算のレシピ, 10.4 多次元の滑降シンプレックス法, pp.295-299, 技術評論社
- [15] CRAN, Package ‘genalg’ ,
<https://cran.r-project.org/web/packages/genalg/genalg.pdf>
- [16] 間瀬茂, R 基本統計関数マニュアル, p.308~313,
<https://cran.r-project.org/doc/contrib/manuals-jp/Mase-Rstatman.pdf>

11. 付録

以下に、受信器のマイクロコントローラ PIC24FJ64GA002 に実装したプログラムの main ソースおよび delay 関数を使用するためのライブラリとヘッダーファイル内容を掲載する。

```
8 #include <p24fj64ga002.h>
9 #include <stdio.h>
10 #include <stdlib.h>
11 #include "FSIO.h"
12 #include "delay_C30.h"
13
14 _CONFIG1(JTAGEN_OFF & GCP_OFF & GWRP_OFF & BKBUG_OFF
15         & COE_OFF & ICS_PGx1 & FWDTEN_OFF)
16 _CONFIG2(IESO_ON & FNOSC_FRCDIV & FCKSM_CSDCMD
17         & OSCIOFNC_ON & IOL1WAY_OFF & I2C1SEL_PRI & POSCMOD_NONE)
18
19     FSFILE *pointer;
20     char crlf[]="r\n";
21     int bytesWritten,i=1,k=0;
22     char fileopen=0,swl=0;
23     double VLT=0;
24     char LogSTRT[] = "Log Startr\n";
25     char LogEND[] = "Log End";
26     char newFile[] = "newfile.txt";
27     char writeArg[] = "w";
28     char fname[32];
29
30 #define STARTSW PORTBbits.RB4
31 #define STOPSW PORTAbits.RA4
32 #define LED PORTBbits.RB0
```

図 11.1 main ソース (冒頭)

```

34  int main(void) {
35
36     CLKDIV=0x0200;           //原振8MHzを4分周 (2MHz)
37     LATA=0x0000;           //Aポート出力一旦クリア
38     LATB=0x0000;           //Bポート出力一旦クリア
39     PORTA=0x0000;          //Aポート入力一旦クリア
40     PORTB=0x0000;          //Bポート入力一旦クリア
41     TRISA=0x0011;          //RA0,4を入力指定
42     TRISB=0x0070;          //RB4,5,6を入力指定
43     CNPU1=0x78EB;          //CNO,1,3,5,6,7,11,12,13,14をプルアップ
44     CNPU2=0x0801;          //CN16,27をプルアップ
45
46     RPINR20bits.SDI1R = 6;  //SDIをRP6 (RB6)ピン
47     RPOR3bits.RP7R = 8;    //SCKをRP7 (RB7)ピン
48     RPOR4bits.RP8R = 7;    //SDOをRP8 (RB8)ピン
49
50     PR3 = 0x1869;           //50msインターバル
51     T3CON = 0b1000000000010000; //プリスケラ1:8
52
53     IEC0bits.ADIIE = 1;    //AD割り込み許可
54     AD1CON1 = 0x8044;       //ADオン,整数,タイマ3トリガ,サンプル自動
55     AD1CON2 = 0x0400;       //Vr+をVdd,Vr-をVss,自動スキャン,変換完了ごとに割り込み
56     AD1CON3 = 0x0C08;       //システムクロック利用,Taq=12Tad,Tad=5Tcy
57     AD1CHS = 0x0000;        //MUXAのVr-をVssに
58     AD1PCFG = 0xFFFE;       //ANO (RA0)をアナログ指定
59     AD1CSSL = 0x0001;       //ANO (RA0)を自動スキャン対象に
60
61     LED = 1;                //初期設定終了後LED点灯

```

図 11.2 main ソース (main 関数冒頭)

```

63     while(1){ //SDカード挿入時チャタリング対策
64         if(PORTBbits.RB5 == 0){ //SDカード挿入されてたら
65             k += 1; //kを1増やし
66             delay_ms(10); //10ms待つ
67             if(k==30){ //kが30を超えたら (300msを超えたら)
68                 break; //whileループを抜けて次の処理へ
69             }
70         }
71     }
72
73     while(PORTBbits.RB5 != 0){ //SDカード挿入待ち (チャタリング対策)
74         delay_ms(50); //挿入されたら50ms待つ (チャタリング対策)
75         LED = 0;
76         while(!FSInit()){ //SD初期化
77             while(1){
78                 sprintf(fname, "data_%d.txt", i); //data_1,2,3...txtというファイル名の元をfnameに記憶
79                 pointer = FSfopen(fname, "r"); //fnameに記憶したファイル名でファイルを読み取りモードで開く
80                 if(pointer != NULL){ //ファイルが開けたら
81                     FSfclose(pointer); //一旦閉じて
82                     i += 1; //fnameに記憶してあるファイル名の数字を1増やす
83                 }else{ //存在しない数字を伴うファイル名になったら
84                     break; //whileループを抜け出す
85                 }
86             }
87             pointer = FSfopen(fname, writeArg); //上の処理でfnameに記憶したファイル名でファイルを開く
88
89             LED = 1; //開けたら赤LEDを消灯

```

図 11.3 main ソース (SD 挿入、測定値書き込みファイルの準備)


```

91 bytesWritten = FSfwrite((const void *)LogSTRT, 11, 1, pointer); //まずLog StartとSDのファイルに書き込む
92
93 while(1){ //stopスイッチがONになるまで無限ループ
94     if(STARTSW == 0){ //startスイッチオンになったら
95         sw1 = 1; //SDへ書き込むためのフラグを立てる
96     }
97     if(STOPSW == 0){ //stopスイッチオンになったら
98         sw1 = 0; //SDへ書き込むフラグをクリアし
99         break; //無限ループを抜け出す
100    }
101 }
102
103 bytesWritten = FSfwrite((const void *)LogEND, 7, 1, pointer); //最後にLog Endと書き込む
104 FSfclose(pointer); //SDのファイルポインタを閉じる
105
106 while(1){ //測定が正常に終了すればLEDは点滅を繰り返す
107     LED = 1;
108     delay_ms(50);
109     LED = 0;
110     delay_ms(100);
111     LED = 1;
112     delay_ms(50);
113     LED = 0;
114     delay_ms(500);
115 }
116 } //end of main()

```

図 11.4 main ソース (測定の開始と終了、main 関数の終了)

```

118 /****AD変換割り込みとSDへ書き込み処理****/
119 void __attribute__((interrupt, auto_psv)) _ADC1Interrupt(void){
120     IFS0bits.AD1IF = 0; //割り込みフラグクリア
121     if(sw1 == 1){ //startスイッチがONならばSDへの書き込み(測定)を開始
122         LED = 0; //測定中はLED消灯
123         FSfprintf(pointer, "%d", ADC1BUF0); //ADバッファ内のint型整数をSDへ書き込み
124         bytesWritten = FSfwrite((void *)crlf, 2, 1, pointer); //改行
125     }
126 }

```

図 11.5 main ソース (測定値書き込みのための割り込み処理関数)

```

2  #include "delay_C30.h"
3
4  int j;
5
6  void delay_us(int usec){
7      usec=(int)(CLOCK * usec)/40;
8      while(usec){
9          asm("nop");
10         asm("nop");
11         asm("nop");
12         asm("nop");
13         asm("nop");
14         asm("nop");
15         asm("nop");
16         asm("nop");
17         asm("nop");
18         asm("nop");
19         asm("nop");
20         asm("nop");
21         asm("nop");
22         asm("nop");
23         asm("nop");
24         asm("nop");
25         usec--;
26     }
27 }
28
29 void delay_ms(int msec){
30
31     for(j=0;j<msec;j++){
32         delay_us(1000);
33     }
34 }

```

図 11.6 delay 関数を使用するためのソースファイル「delay.c」

```

8  #ifndef DELAY_C30_H
9      #define DELAY_C30_H
10
11     #define CLOCK 2
12
13     void delay_us(int usec);
14     void delay_ms(int msec);
15
16 #endif /* DELAY_C30_H */

```

図 11.7 delay 関数を使用するためのヘッダファイル「delay.h」

12. 研究業績

[1].N. Haga, Y. Kasahara, and K. Motojima, “Dynamic measurements of intrabody communication channels and their dependence on grounding conditions,” *IEICE Trans. Commun.* (accepted).